

## **SCRUMIN TOIMIVUUS KÄYTÄNNÖSSÄ**

Juho Laukka  
Opinnäytetyö  
Syksy 2024  
Tieto- ja viestintätekniikka  
Oulun ammattikorkeakoulu

# TIIVISTELMÄ

Oulun ammattikorkeakoulu  
Tieto- ja viestintäteknikka  
Ohjelmistokehitys

Tekijä: Juho Laukka

Opinnäytetyön otsikko: Scrumin toimivuus käytännössä

Työn ohjaaja: Pekka Alaluukas

Työn valmistumislukukausi ja -vuosi: syksy 2024

Sivumäärä: 20

Opinnäytetyön tarkoituksena ei ollut verrata Scrumia suoraan muihin työmalleihin, tai etsiä eroavaisuuksia niiden välillä, vaan tutkia, miten Scrum toimii käytännössä teoreettiseen Scrumiin verrattuna.

Scrumin teoriaa tutkimiseen käytin materiaalia useilta ketteriin malleihin keskittyviltä sivustoilta. Käytännön kokemuksia keräsin haastattelemalla eri taustaisia sovelluskehityksen ammattilaisia.

Lopputuloksena kävin läpi haastatteluista keräämiäni kokemuksia, joista tunnistin yleisiä ongelmia ja riskejä, joita scrumtiimit ovat kokeneet. Niille etsin mahdollisia syitä ja kuinka ongelmia voitaisiin välttää.

## **ABSTRACT**

Oulu University of Applied Sciences  
Degree Program in Information Technology  
Option of Software Development

Author: Juho Laukka

Title of thesis: Scrum's effectiveness in practice

Supervisor: Pekka Alaluukas

Term and year when the thesis was submitted: autumn 2024

Number of pages: 20

The purpose of the thesis was not to directly compare Scrum with other work models or to identify differences between them but to examine how Scrum functions in practice compared to a more theoretical view of Scrum.

To study Scrum in theory, I used material from several websites focused on agile models. Practical insights were gathered by interviewing professionals in software development with varying backgrounds. Based on this, I identified common problems and risks experienced by Scrum teams.

# SISÄLLYS

1	JOHDANTO .....	5
2	KETTERÄT MALLIT .....	6
3	SCRUM TEORIASSA .....	7
3.1	Arvot .....	8
3.2	Scrum-tiimi .....	8
3.3	Sprintit .....	9
3.4	Taskit .....	9
3.5	Vauhti .....	10
3.6	Scrumin tapahtumat .....	10
3.6.1	Sprintin suunnittelu .....	11
3.6.2	Kehitysjonon työstö .....	11
3.6.3	Daily .....	11
3.6.4	Katselmointi .....	12
3.6.5	Retrospektiivi .....	12
4	SCRUMIN SKAALAAMINEN .....	13
4.1	Scrum of Scrums .....	13
4.2	Scrum of Scrums of Scrums .....	14
5	SCRUM KÄYTÄNNÖSSÄ .....	15
5.1	Inkrementtien toteutus .....	15
5.2	Scrum seremoniat ja palaverit .....	16
5.3	Prosessien noudattaminen ja ketteryys .....	16
5.4	Skaalautuvuuden hankaluudet .....	17
5.5	Jaetut roolit .....	18
6	POHDINTA .....	19
7	LÄHTEET .....	20

# 1 JOHDANTO

Tämän opinnäytetyön tarkoituksena on toimia pohjana, jonka päälle voin rakentaa tietämystä ja heijastaa omia kokemuksiani. Olen kerännyt tietoa Scrumin perusteista ja filosofiasta, sekä lisäksi käytännön näkökulmia eri taustaisten ammattilaisten kokemuksista.

Ketterät työmallit ovat yleisessä käytössä ohjelmistokehityksessä. Omat kokemukseni ovat herättäneet kiinnostusta ja tahtoa oppia enemmän, jotta voin kasvattaa omaa hyödyllisyyttä osana ketterää tiimiä. Scrum on yksi yleisimmistä ketteristä malleista, jonka ideana on jakaa projekti pienempiin osiin, joita työtetään yksi inkrementti kerrallaan. Näen hyödylliseksi oppia enemmän Scrumista sekä teoriassa, että käytännössä.

Työssä keskityn Scrumin teoriaan sekä hieman sen historiaan, jonka perusteet ja keskeiset periaatteet muodostavat pohjan käytännön tapojen tutkimiselle. Scrumin käytäntöjen tarkastelu pohjautuu omiin kokemuksiini, joita olen ehtinyt kerätä scrummasterina toimiessani, sekä haastatteluihin alan ammattilaisten kanssa, joilla on vaihtelevat taustat ja näkemykset Scrumista. Näiden tietojen avulla tunnistan yleisimpiä ongelmia ja riskejä, joita Scrumin soveltamisessa voi esiintyä. Lopussa esitän myös mahdollisia keinoja parantaa näitä ongelmia.

## 2 KETTERÄT MALLIT

Ketterillä malleilla tarkoitetaan lyhyihin ja toistuviin inkrementteihin perustuvaa ohjelmistokehitystä. Jatkuva suunnittelu ja kehitys voivat tapahtua samanaikaisesti, jolloin projektin suuntaan ja mahdollisiin uusiin tarpeisiin voidaan vaikuttaa nopeasti. Tämä rakenne antaa asiakkaalle mahdollisuuden seurata ja antaa palautetta projektin aikana. (Guru99.com 2024.)

Ketterien mallien käytäntöjä on harrastettu jo 1990-luvun aikana, mutta vasta 2000-luvun alkupuolella ne ovat yhtenäistetty ja kiteytetty yhden termin alle. Teknologian kehityksen nopeutuessa syntyi tarve ketterämmille työtavoille. Yhden projektin aikana asiakkaan tarve oli saattanut jo muuttua, eivätkä käytetyt teknologiat välttämättä olleet enää ajan tasalla projektin valmistuttua. Korjauksia kehityksen aikana syntyneisiin ongelmiin jouduttiin myös odottamaan jopa vuosia. Näiden ongelmien myötä useat tiimit adoptoivat itsenäisesti ketterämpiä tapoja tuottaa ohjelmistoa, joiden yleiset arvot on koostettu Agile Manifestoon. (Planview.com 2018.)

Nykyään ketterät mallit noudattavat Agile Manifeston asettamia perusteita, mikä korostaa asiakkaan tyytyväisyyttä ja nopeaa reagointia muuttuviin vaatimuksiin, ohjaa motivoituneita ammattilaisia tuottamaan korkealaatuista ohjelmistoa nopeilla ja toistuvasta julkaisuilla sekä nostaa arvoa itseohjautuvan tiimin sisäisestä työskentelystä ja työtapojen jatkuvasta parantamisesta. (Adaptovate.com 2022.)

### 3 SCRUM TEORIASSA

Scrum on ketterä työmalli, joka perustuu ylläpidettyyn kehitysjonoon sekä toistuviin, noin muutaman viikon pituisiin sykleihin, joita kutsutaan sprinteiksi. (Guru99.com 2024.) Yhden sprintin aikana on tarkoitus toteuttaa inkrementti, mikä koostuu useasta pienemmästä tehtävästä, eli taskista. Taskeista muodostetaan kehitysjono, joka kertoo järkevän järjestyksen taskien kehittämiseksi. Sprintin suunnittelun aikana kehitysjonolta valitaan sprintille sopiva määrä olennaisia taskeja tiimin koosta riippuen, jolloin kaikki sprintille valitut taskit saadaan valmiiksi seuraavaan sprintin alkuun mennessä. (Agilealliance.org 2024a.)

Jokaisella Scrum-tiimin jäsenellä on valittu rooli. Suurin osa tiimistä koostuu kehittäjistä, jotka toteuttavat ja testaavat taskeja sprintin aikana. Tuoteomistaja edustaa asiakasta ja ylläpitää kehitysjonoa, josta tarjotaan kehitystiimille asiakkaan vaatimusten mukaisia taskeja tuotettavaksi. (Cprime.com 2024.) Molemmat osapuolet toimivat yhdessä taskien työstössä. Scrummaster ajaa ketterien tapojen käyttöä ja varmistaa sovittujen prosessien seuraamisen. Hän edesauttaa tiimin tehokkuutta poistamalla mahdolliset esteet tuotannossa. (Agilealliance.org 2024a.)

Scrumilla voidaan vähentää ylimääräisiä palavereita ja korvata ne lyhyillä, säännöllisesti toistuvilla tapahtumilla. Läheinen työskentely asiakkaan kanssa, sekä valmius ketteriin muutoksiin tuotannon aikana ovat yksiä Scrumin keskeisimmistä periaatteista.

Virallisesti Scrum nimistä työmallia on alettu harrastamaan 1990-luvun puolella. Vuosien aikana Scrum on käynyt läpi monia iteraatiota, ja saa modernin perustana vuonna 2001 julistetusta Agile Manifestosta. (Planview.com 2018.)

### 3.1 Arvot

(Agilemanifesto.org 2001.) Scrum noudattaa Agile Manifestossa määritettyjä arvoja:

- Yksilöt ja niiden vuorovaikutus prosessien ja työkalujen yli.
- Toimiva ohjelmisto kattavan dokumentaation yli.
- Asiakkaan välinen yhteistyö sopimusneuvottelujen yli.
- Muutokseen vastaaminen tiukan suunnitelman yli.

Scrum päällimmäisinä periaatteina on läheinen yhteistyö asiakkaan kanssa, jolloin läpinäkyvyys tuotantoon kasvaa, sekä ongelmien ja muutosten tarpeiden ilmenemiseen pystytään reagoimaan vielä myöhään tuotannossa. Jatkuva ja nopea julkaisu mahdollistetaan motivoituneella tiimillä, jota tuetaan johtotasolla, sekä yleisesti hyvällä työympäristöllä. (Agilealliance.org 2024a.)

### 3.2 Scrum-tiimi

Scrum-tiimi on itseohjautuva, eli ulkoisen ohjauksen sijaan tiimi itse päättää työskentelytavoistaan. (Guru99.com 2024.) Tämän edellytyksenä on tuoteomistajan ylläpitämä kehitysiono, jota työstetään jatkuvasti. Tuoteomistajan tulee pitää kehitysiono mahdollisimman selkeänä ja järjestää se siten, että jonolla olevat tehtävät valmistuvat järkevässä järjestyksessä. (Cprime.com 2024.)

Scrummaster auttaa tuoteomistajaa ymmärtämään kehitysionon järjestämistä ja ylläpitää ketteryuden harjoittamista. Hän myös fasilitoi Scrumin tapahtumia aina tarvittaessa. (Cprime.com 2024.) Scrummaster on perillä tiimin tarpeista ja ongelmista ja ajaa niitä eteenpäin. Tiimistä riippuen Scrummaster voi olla osa kehitystiimiä, jolloin hänen vastuunsa on jaettu kehittämisen ja Scrumin ylläpitämisen välillä.

Kehitystiimi koostuu monitaitoisista ammattilaisista, jotka voivat itseohjautuvasti tuottaa sprintille suunnitelluista tehtävistä valmiita inkrementtejä tuotetta varten. (Cprime.com 2024.) Vastuu kehityksestä kuuluu yksinomaan kehitystiimille. Sopiva koko yhdelle Scrum-tiimille vaihtelee tuotteen mukaan, mutta on yleensä

noin 8-10 henkilöä. Tiimin tulee olla tarpeeksi pieni, jotta voidaan hyödyntää ketteriä toimintatapoja, mutta myös tarpeeksi suuri, että sprinttien aikana saadaan valmiiksi merkittävä määrä työtä.

### **3.3 Sprintit**

Sprintit ovat toistuvia ajanjaksoja, jolloin Scrum-tiimi tekee tuoteomistajan kanssa suunniteltuja tehtäviä. Tehtävät otetaan kehitysjonolta niille oleellisessa järjestyksessä. Yleensä yksi sprintti kestää muutamaan viikon, jonka lopussa pidetään retrospektiivi ja sprintin katselmointi. Tämän jälkeen suunnitellaan ja aloitetaan uusi sprintti. (Cprime.com 2024.) Yhden sprintin aikana kehitystiimi saa toteutettua yhden inkrementin, joka voi olla vielä puutteellinen, mutta toimiva osa tuotetta. Tarvittaessa tuoteomistaja voi keskeyttää sprintin ennen sen päättymistä. Näin voidaan tehdä, jos sprintin tavoitteet osoittautuvat tarpeettomiksi. Sprintin keskeyttäminen on yleensä raskas prosessi tiimin jäsenille, joten sitä ei harrasteta kovin herkästi.

### **3.4 Taskit**

Kehitysjonolla oleva työ jaetaan tehtäviksi, eli taskeiksi. (Cprime.com 2024.) Yleensä taski on osa suurempaa kokonaisuutta, mutta yhdestä taskista tulee kuitenkin syntyä mahdollisimman itsenäinen toteutus. Taskissa tulee olla kaikki tarvittava tieto sen toteutusta varten, sekä määrittäminen sen valmiudesta, jotta itse kehittäminen sujuu sulavasti ja inkrementti toteutuu ongelmitta. Valmiuden määrittämisellä voidaan todeta, onko taski toteutettu oikein.

Valmiilla taskilla on kehitystiimin arvioima pisteytys, joka kertoo taskin toteuttamisen hankaluuden. Pistemäärä perustuu oletukseen, että yksi kehittäjä pystyy toteuttamaan taskin yhden sprintin aikana. Liian suuren pistemäärän omaavat taskit tulee koittaa pilkkoa useaksi pienemmäksi taskiksi. Pisteiden skaalana voidaan käyttää suoraan taskin toteuttamiseen kuluvien päivien määrää, tai mitä tahansa satunnaista skaalaa, kunhan se on yhtenäinen ja Scrum-tiimille selkeä. Sprintin päättyessä voidaan valmistuneiden taskien

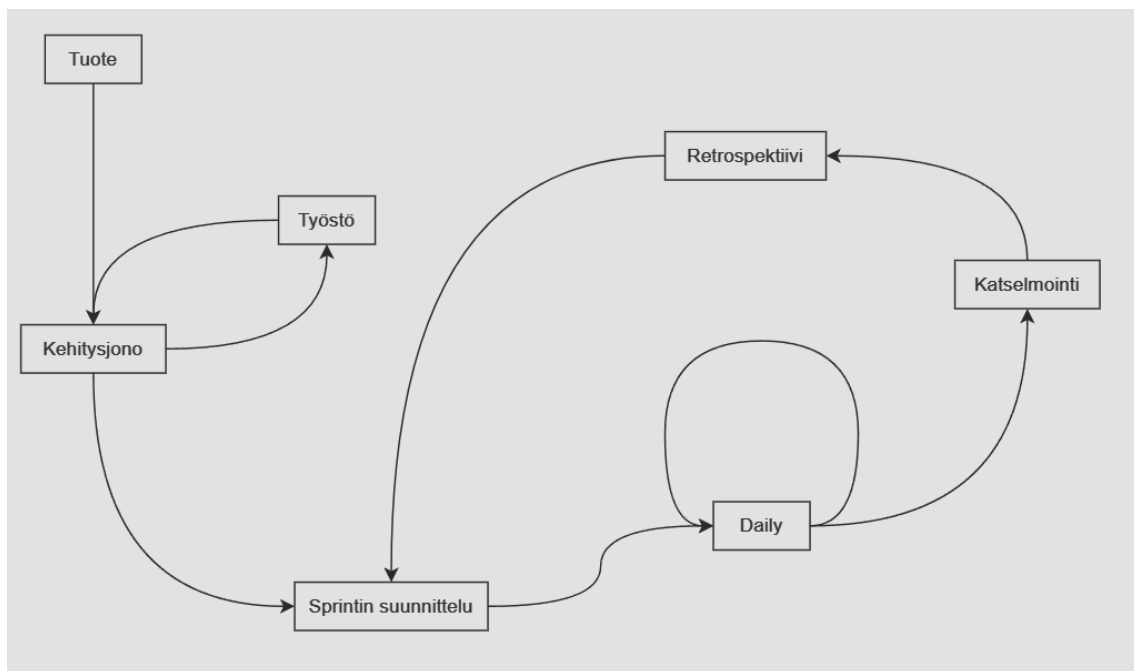
pisteistä laskea tiimin vauhti, eli arvioi tiimin tehokkuudesta yhden sprintin aikana. Sprintin pituus tulee pitää mahdollisimman tasaisena todellisen vauhdin laskemiseksi.

### 3.5 Vauhti

Tiimille laskettua vauhtia käytetään hyväksi sprintin suunnittelussa. (Cprime.com 2024.) Sen avulla voidaan tarkemmin arvioida kuinka paljon, ja mitä työtä tulevan sprintin aikana pystytään tekemään. Ideaalisesti sprintille suunnitellaan juuri sen verran työtä, mitä kehitystiimi kykenee toteuttamaan. Taskit, jota kehitystiimi ei ehdi toteuttaa yhden sprintin aikana, jatkuvat seuraavalla sprintillä. Jatkuva vuoto ei ole hyväksi sprinttien terveydelle; se estää inkrementtien valmistumista sekä vääristää tiimin vauhtia.

### 3.6 Scrumin tapahtumat

Scrumissa koitetaan minimoida tarvittavien palaverien määrää suorittamalla säännöllisiä palavereita, joiden ideana on jalostaa arvoa tiimille, pitää lyhyitä tilannekatsauksia tiimin kesken päivittäin, käydä läpi mitä tiimi on saanut aikaan sprintin aikana sekä pohtia mahdollisia parannuskeinoja tiimin tehokkuudelle.



## *KUVA 1. Scrumin tapahtumat ja sprintin rakenne*

### **3.6.1 Sprintin suunnittelu**

Ennen uuden sprintin alkua, Scrum-tiimi suunnittelee tulevalle sprintille valmistuvan inkrementin. Inkrementti muodostuu tuotteen omistajan ehdottamista taskeista, jotka saadaan kehitysjonosta tärkeysjärjestyksessä. (Agilealliance.org 2024a.) Tätä edellyttää, että kehitysjonon taskit ovat tarpeeksi selkeitä ja niiden valmius on määritetty. Scrummasterin ja tuotteenomistajan tehtävänä on varmistaa inkrementin selkeys kehitystiimille. Sprintille suunniteltuja taskeja ei määrätä kellekkään yksittäisellä henkilöllä, vaan sprintin alussa kehitystiimi aloittaa taskien toteuttamisen itsenäisesti.

### **3.6.2 Kehitysjonon työstö**

Kehitysjonon työstöllä luodaan selkeyttä tuleville inkrementeille, jonka avulla saadaan sprintin suunnittelun taakkaa kevennettyä. Sessioita kehitysjonon työstöön voidaan pitää aina tarvittaessa sprintin aikana ja työstöön osallistuu koko Scrum-tiimi. Session aikana on tarkoitus käydä läpi kehitysjonon kokonaisuuksia ja työstää niistä valmiita taskeja. Puutteita voidaan korjata palaverin aikana tai sen jälkeen, jolloin taski tulee katselmoida uudelleen seuraavassa sessiossa. Kun taski ollaan saatu jalostettua valmiiksi, voi kehitystiimi siihen arvioida kuluvan pistemäärän.

### **3.6.3 Daily**

Daily, eli päivittäispalaveri, nähdään yleensä Scrumin tärkeimpänä palaverina. Daily pidetään nimensä mukaan päivittäin, ja siihen osallistuu jokainen kehitystiimin jäsen. Dailyssä käydään lyhyesti läpi mitä tehtäviä on tehty edellisen dailyn jälkeen, sekä mihin kukin jäsen keskittyy seuraavaksi. (Agilealliance.org 2024a.) Tarkoituksena on nostaa esille tehtävien sujuvuus sekä mahdolliset ongelmat, mitkä estävät taskien toteuttamista. Dailyssä itsessään ei ole tarkoitus ratkoa ongelmia, vaan scrummaster ottaa vastuun ja järjestää tarvittavat

jatkopalaverit, joissa ongelmat voidaan käydä läpi. Tarkoitus on kutsua vain oleelliset henkilöt, jolloin vähennetään palaveriin käytettyä aikaa.

#### **3.6.4 Katselmointi**

Sprintin lopussa kehitystiimi esittelee sprintin aikana toteutetun inkrementin tuoteomistajalle. Toteutukset voidaan käydä läpi esittelemällä jokainen yksityiskohtaisesti juurta jaksuen, mutta olennaisinta on tiimin valmiiksi saadut taskit. (Agilealliance.org 2024a.) Katselmoinnissa voidaan myös käydä läpi sprintin statistiikkaa, eli analysoidaan sprintin vauhtia ja verrataan sitä aiempiin sprintteihin.

Tilaisuuteen voi osallistua henkilöitä Scrum-tiimin ulkopuolelta, jolloin saadaan palautetta myös Scrum-tiimin ulkopuolelta. Katselmointi luo läpinäkyvyyttä tiimin ulkopuolisille tahoille. (Agilealliance.org 2024a.)

#### **3.6.5 Retrospektiivi**

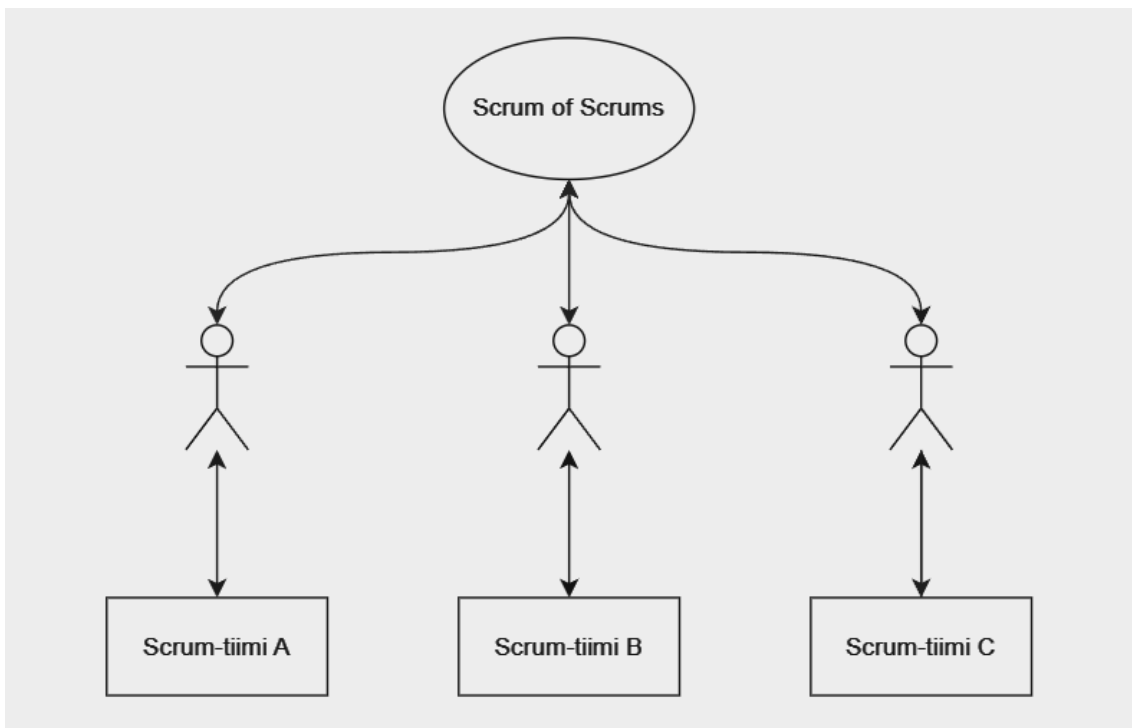
Retrospektiivi pidetään sprintin lopuksi, ennen uuden sprintin suunnittelua. Retrospektiivissä tarkastellaan tiimin kesken sprintin sujuvuutta yhteistyöhön, prosesseihin ja työkaluihin liittyen. Tarkoituksena on tunnistaa tiimissä ilmeneviä hyviä ja huonoja käytäntöjä, joiden perusteella löydetään tärkeimmät parannukset. Scrummaster varmistaa, että tapahtuma pidetään ja sen tarkoitus ymmärretään. Retrospektiivin lopputuloksena tiimin on tarkoitus löytää mahdollisia muutoksia, joilla saadaan tiimin työskentelytapoja ja tehokkuutta parannettua. (Agilealliance.org 2024a.) Parannuksia voidaan ottaa käyttöön heti seuraavalla sprintillä.

## 4 SCRUMIN SKAALAAMINEN

Suuremmissa projekteissa voi olla useampia Scrum-tiimejä, tällöin nousee myös tarve tiimien väliselle kommunikaatiolle ja yhteistyölle. Koska jokainen Scrum-tiimi toimii itsenäisesti, saattavat tiimien riippuvuudet toisistaan osoittautua hyvinkin hankalaksi.

### 4.1 Scrum of Scrums

Scrum of Scrums (SoS) ratkaisee riippuvuuksiin ja kommunikointiin liittyvät ongelmat yhdistämällä eri Scrum-tiimit SoS -tapahtuman kautta. SoS toimii hyvinkin samalla tavalla kuin normaali dailyt yhdessä Scrum-tiimissä; jokaisen tiimin edustaja osallistuu palaveriin, jossa käydään läpi Scrum-tiimille olennaiset taskit, aikataulut, ongelmat ja muut asiat, mitkä voivat olla hyödyllistä jakaa muille tiimeille. (Agilealliance.org 2024b.)



KUVA 2. Kommunikaatio eri tiimien välillä Scrumin of Scrumsissa

Kuten Scrumin dailyssä, SoS:ssa nostettuja ongelmia ei käydä läpi kokonaisuudessaan sillä hetkellä, vaan Scrum of Scrums -master on apuna niiden ratkaisussa ja löytää tarvittavat vastuuhenkilöt. (Business.adobe.com 2023.) Scrum of Scrums on tarkoitus olla nopea ja tehokas tapahtuma, joissa konteksti saattaa muuttua paljon Scrum-tiimien välillä.

## **4.2 Scrum of Scrums of Scrums**

Scrum of Scrums of Scrums on seuraava askel Scrumin skaalauksessa. Hyvin samankaltainen kuin SoS, mutta tässä eri SoS -tiimien välinen katsaus käydään läpi vielä korkeammalla tasolla. (Agilest.org 2024.) Tällä tasolla esitetyt ongelmat kiteytyvät yksinkertaisempaan muotoon, sillä keskustelu ei ole enää suoraan yhteydessä kehitykseen.

## 5 SCRUM KÄYTÄNNÖSSÄ

Tämä osio perustuu haastattelemieni ammattilaisten perusteella kerättyihin kokemuksiin ja mielipiteisiin. Henkilöiden kokemus Scrumista vaihtelee alle vuoden ja useamman kymmenen vuoden välillä. Olen kerännyt tähän sekä hyväksi todettuja käytäntöjä, että yleisiä ongelmia, mitä useassa Scrum-tiimissä on noussut esiin.

### 5.1 Inkrementtien toteutus

Suurin osa haastattelemistani ihmisistä kokee sprinttien sopivaksi pituudeksi noin 2-3 viikkoa. Kahdessa viikossa suuremmalla Scrum-tiimillä voi kulua liikaa aikaa Scrumin tapahtumissa, kun taas neljä viikkoa ei nähdä enää tarpeeksi ketteränä. Optimaalinen pituus riippuu tiimin koosta ja tuotteesta, joka löytyy ajan kanssa ja eri pituuksia kokeilemalla. Joillain ihmisillä on myös kokemuksia yhden viikon pituisia sprinteistä, mutta silloin Scrumin tapahtumat ovat olleet paljon vapaamuotoisempia, tai niitä kaikkia ei ole aina järjestetty lainkaan.

Suuremmat sprintit on todettu helpommaksi toteuttaa pienempiin verrattuna, koska pidemmällä ajalla on enemmän liikkumavaraa ja inkrementti suunnitellaan helpommin kevyemmäksi.

Lyhyiden sprinttien ja suurten tiimien kanssa kehitysjonon ylläpidon tärkeys on tullut hyvin esille. Kun kehitysjonoa ei ole työstetty tarpeeksi, se on johtanut pidempiin suunnitteluihin ja epäselvyyksiin kehityksen aikana. Taskien epäselvyys on ollut yleinen ongelma monissa tiimeissä.

Kehittäjät saattavat jättää vähemmän mielekkäät taskit myöhemmäksi sprinttiä, jolloin ne jäävät herkemmin valmistumatta, eikä mahdollisia ongelmia tunnisteta tarpeeksi nopeasti. Joissain tiimeissä tämä on ratkaistu valitsemalla taskeille kehittäjät jo suunnittelun aikana, jolloin hankalimmat tiketit eivät aina jäisi sprintin loppupäähän.

## 5.2 Scrum seremoniat ja palaverit

Dailyjen pituudessa ja sen annissa on nähty ongelmia aina. Suurissa tiimeissä daily venyy usein jopa 45 minuutin mittaiseksi, eivätkä kaikki jäsenet saa tasa-arvoista puheenvuoroa. Usein muutama tiimiläinen saattaa keskustella jostain asiasta liian yksityiskohtaisesti ja liian kauan, jonka johdosta muiden puheenvuoro dailyssä mitätöityy. Sopivan mittaiseksi dailyksi nähdään näin 30 minuuttia, joista ensimmäinen puolisko käytetään itse dailyyn, ja toinen on varattu mahdolliselle jälkipuinnille dailyssä nousseista aiheista.

Retrospektiivi nähdään Scrum-tiimin kannalta yhtenä mielekkäimmistä Scrumin tapahtumista. Retrospektiivissa päästään jakamaan mielipiteitä ja kokemuksia, joka auttavat niiden prosessoinnissa. Tämä koetaan positiivisena vaihteluna normaaliin työelämään, koska yleensä kehitystiimi kokee hankalaksi tulla kuulluksi tiimin ulkopuolella. Retrospektiivi voidaan myös kokea hyvinkin turhauttavana, jos sen perusteella ei koskaan synny parannuksia. Pieniä muutoksia tiimin käytännöissä on helppo kokeilla, eikä silloin vähemmän toimivista muutoksista ole enää hankala luopua jälkeinpäin. Yleinen mielipide on, että näitä pieniä muutoksia pitäisi uskaltaa kokeilla enemmän.

Palavereita pidetään usein liian paljon, eikä niistä aina ole arvoa kaikille osallistujille. Palavereihin kutsutaan turhaan ihmisiä, joiden ei niissä tarvitsisi olla. Ajanvarauksia ei myöskään aina kunnioiteta, vaan tapahtuma venyy varauksen ulkopuolelle. Tapahtumat kuuluisi pitää lyhyinä ja tehokkaina, mutta liian niukat ajanvaraukset on nähty myös ongelmalliseksi.

## 5.3 Prosessien noudattaminen ja ketteryys

Prosesseja saatetaan herkästi seurata liian tarkkaan, tai liian löyhästi. Prosessien tulisi olla selkeitä ja Scrum-tiimiä palvelevia, ilman että ne sitovat tiimiä liikaa. Kaikki Scrumin tapahtumat tulisi pitää, koska se parantaa lyhyen aikavälin ennustettavuutta.

Taskien vaatimukset ovat usein liian epäselviä, ja niiden selvittäminen jää kehitystiimille. Tämä hankaloittaa taskien työstämistä, sekä kehitysjonon selkeyttä. Taskeista jätetään usein myös pois testauksen näkökulma ja siihen kuluva aika. Toteutusta ei aina selvennetä taskilla, jolloin testaus ja valmiuden varmistaminen hankaloituu. Taskien pisteytys saattaa myös olla liian epätarkka, jos taskin tehtävä ei ole tarpeeksi selvä tiimille.

Uusia taskeja saatetaan luoda ja ottaa heti työn alle kesken sprintin. Pienissä määrissä ja yksinkertaisten taskien kanssa tämä ei välttämättä nouse ongelmaksi, mutta saattaa silti herkästi karata käsistä. Joskus taskeja on pakko ottaa suoraan sprintille, jotta koko inkrementti saadaan valmiiksi sprintin aikana. Kesken sprintin otettujen taskien tulee silti olla laadukkaita ja tehtävän tulisi olla selvää sekä suunnittelijalle, että tekijälle. Sprintti tulisi aloittaa uusiksi jos suunnitelma muuttuu liikaa, mutta kovin moni ei sitä halua harrastaa.

Suunnittelussa tuoteomistajan ehdottama inkrementti annetaan mennä suoraan läpi, eikä kehitystiimi aina osallistu keskusteluun. Näissä tapauksissa suunnittelu nähdään ajan hukkana, eikä tiimiä kiinnosta tarpeeksi, kunhan he saavat aloittaa taskien kehittämisen. Jotkus tiimit ovat myös jättäneet kehitysjonon työstön kokonaan pois, jolloin kehitystiimi käy taskit tarkemmin läpi vain suunnittelun aikana. Kehitysjonon jatkuva työ nähdään joskus pakollisena pahana, eikä siitä tulevaa hyötyä havaita tiimin sisällä. Liian kevyillä prosesseilla päädytään lopulta olettamaan asioita taskien määrityksen ja toteutuksen suhteen.

Prosessien noudattamisessa jokaisen tiimin tulee löytää omat tapansa. Scrumin prosessit tunnistetaan ehdotuksina, eikä lakeina. Ne ohjaavat oikeaan suuntaan, mutta loppujen lopuksi jättävät ohjat tiimin käsiin. Ihmisten hyvinvointi ja viihtyvyys nähdään tärkeämpänä, kuin prosessien tarkka noudattaminen.

#### **5.4 Skaalautuvuuden hankaluudet**

Scrum-tiimin koko vaikuttaa monesti olevan aivan liian suuri. Hyväksi Scrum-tiimin kooksi on todettu noin 5-10 henkilöä, mutta usealla henkilöllä on ollut kokemusta jopa kahdenkymmenen henkilön tiimeistä. Näin suurissa tiimeissä on ilmennyt ongelmia kommunikaatiossa ja tiedonjaossa. Esimerkiksi sprintin

suunnittelu on myös hankaloitunut suurissa tiimeissä, koska tuoteomistajan on pitänyt käyttää iso osa ajastaan taskien työstöön tiimiä varten. Pienemmät tiimit on koettu laadukkaammaksi Scrumin tapahtumien ja työn jakamisen kannalta. Yleisesti tiimeissä työskentely on parantunut, kun kaikki jäsenet ovat olleet perillä tiimin sisäisistä taskeista.

Tiimien välinen retrospektiivi nähdään hyödylliseksi, mutta jokainen tiimi ei sitä ole aina harrastanut. Scrum of Scrumsin kaltainen tapahtuma on aina nähty pakolliseksi tiimien välisessä yhteistyössä, mutta joitain hankaluuksia kommunikoinnissa on silti havaittu. Scrum of Scrumsissa toistuu myös samat ongelmat mitä yksittäisissä tiimeissä. Dokumentaatio ja sen ylläpito on koettu tärkeäksi eri Scrum-tiimien välisessä yhteistyössä, mutta sitä ei aina harrasteta tarpeeksi. Myös selkeä omistajuus eri osa-alueista on tarpeellista.

Yhdeksi Scrum of Scrums suurimmaksi ongelmaksi on koettu liian monen tiimin osallistuminen, jolloin tapahtuman osallistujat kävivät vain lukemassa listalta raporttinsa, mutta kukaan ei myöskään ottanut nostetuista ongelmista vastuuta. Joissain tapauksissa myös johtoryhmän liittyminen Scrum of Scrumsiin on nähty heikentävän tiimien välistä kommunikointia

## **5.5 Jaetut roolit**

Useimmat scrummasterit ovat olleet jaetussa roolissa, yleensä joko kehittäjänä tai testaajana. Lähes kaikki ovat myös huomanneet työpanoksensa kärsivän, kun kummankaan roolin tehtäviä ei saada toteutettua tarpeeksi hyvin, tai aina ajallaan. Osalla on myös kokemusta usealla tiimille jaetusta scrummasterin roolista, jolloin omien tehtävien jakaminen ja tiimien ylläpito on koettu hankalaksi.

## 6 POHDINTA

Haastattelujen perusteella huomataan, että Scrumia käytetään pohjana, jota voidaan tarpeen mukaan muokata ja jonka päälle voidaan rakentaa jokaiselle tiimille sopiva malli. Scrumin asettamat prosessit nähdään enemmänkin ehdotuksina, joita voidaan tarkoituksellisesti rikkoa tarvittaessa, kunhan se ei oteta tavaksi.

Suurimmaksi ongelmaksi näkisin tiimien suuret koot, mikä luonnollisesti nostaa Scrumin tapahtumien pituutta huomattavasti, sekä vähentää niistä saatua hyötyä. Suuret tiimit kuuluisi jakaa pienempiin, noin 8 henkilön kokoisiin tiimeihin, joissa jokaisessa toimii oma scrummaster. Jaetut roolit aiheuttavat liian suuria työmääriä yhtä henkilöä kohtaan, mutta pienemmissä tiimeissä toimivat paremmin, edellyttäen että prosessit ovat selkeitä koko tiimille.

Taskit, joita ei saada valmiiksi sprintin aikana, on myös liian usein toistuva ongelma. Uuden sprintin suunnittelussa pitäisi antaa enemmän painoarvoa keskeneräisille taskeille, jotta sama ongelma ei toistui taas seuraavalla sprintillä. Testauksen impakti taskin valmistumisen kannalta on myös helposti unohdettu, tai nähty liian vaikeana arvioida. Inkrementti ei toteudu, ellei kaikkia sen taskeja saada valmiiksi, eikä taskia voida todeta valmiiksi ennen sen testausta.

Nämä ovat mielestäni tärkeimpiä ongelmia, mihin ihmiset ovat törmänneet käytännössä. Liian kiireelliset aikataulut ja vaikeasti tavoitettavat henkilöt hankaloittavat epäkohtien korjaamista. Tiimin pitäisi myös alustavasti itse tunnistaa ongelmia ja aktiivisesti haluta niihin muutosta. Retrospektiivi on oleellinen osa tiimin toimintatapojen parantamista, mutta sen hyödyllisyyttä edellyttää motivoitunut ja ammattimainen Scrum-tiimi.

## 7 LÄHTEET

Adaptovate.com 2022. Agile. Agile Methodology Frameworks and Models. Luettavissa: <https://www.adaptovate.com/agile/agile-methodology-frameworks-models/>. Luettu: 16.11.2024.

Business.adobe.com 2023. Scrum of Scrums — everything you need to know. Luettavissa: <https://business.adobe.com/blog/basics/scrum-of-scrums>. Luettu: 12.10.2024

Agilealliance.org 2024a. Glossary. Scrum. Luettavissa: <https://www.agilealliance.org/glossary/scrum/>. Luettu: 12.10.2024.

Agilealliance.org 2024b. Glossary. Scrum of Scrums. Luettavissa: <https://www.agilealliance.org/glossary/scrum-of-scrums/>. Luettu: 12.10.2024.

Agilemanifesto.org 2001. Manifesto for Agile Software Development. Luettavissa: <https://agilemanifesto.org/>. Luettu: 12.10.2024.

Agilest.org 2024. Scaled Agile. Scrum of Scrums. Luettavissa: <https://www.agilest.org/scaled-agile/scrum-of-scrums/>. Luettu: 12.10.2024

Cprime.com 2024. Resources. What is Agile. Luettavissa: <https://www.cprime.com/resources/what-is-agile-what-is-scrum/>. Luettu: 12.10.2024

Guru99.com 2024. Agile Model in Software Engineering. Luettavissa: <https://www.guru99.com/fi/agile-model.html>. Luettu: 17.11.2024.

Planview.com 2018. Resource. History of Agile. Luettavissa: <https://www.planview.com/resources/guide/agile-methodologies-a-beginners-guide/history-of-agile/>. Luettu: 10.12.2024.