

Johanna Hämäläinen

**SCRUMIN HYÖDYT JA HAASTEET KEHITYSTIIMIN
NÄKÖKULMASTA: TAPAUSTUTKIMUS IT-ALAN
PALVELUYRITYKSESSÄ**



JYVÄSKYLÄN YLIOPISTO
TIETOJENKÄSITTELYTIEDEIDEN LAITOS
2013

TIIVISTELMÄ

Hämäläinen, Johanna

Scrumin hyödyt ja haasteet kehitystiimin näkökulmasta: tapaustutkimus IT-alan palveluyrityksessä

Jyväskylä: Jyväskylän yliopisto, 2013, 83 s.

Tietojärjestelmätiede, pro gradu -tutkielma

Ohjaaja: Pirhonen, Maritta

Scrum on ohjelmistokehityksen viitekehys, joka on kehitetty helpottamaan erityisesti monimutkaisten ohjelmistokehitysprojektien hallintaa ja toteuttamista. Scrum on yksi maailman suosituimmista ja tunnetuimmista ohjelmistokehityksen viitekehyksistä ja sitä käytetään sekä pienissä, keskisuurissa että suurissa IT-alan yrityksissä ympäri maailmaa. Tämän pro gradu -tutkielman tarkoitus on luoda käsitys scrumin hyödyistä ja haasteista tieteellisten julkaisujen tarjoaman tutkimustiedon ja eräässä IT-alan palveluyrityksessä toteutettavan tapaustutkimuksen tutkimustulosten pohjalta. Scrumin hyötyjä ja haasteita tarkastellaan ohjelmistotuotteen kehityksestä vastuussa olevan kehitystiimin näkökulmasta.

Avainsanat: ketterät menetelmät, scrum, kehitystiimi

ABSTRACT

Hämäläinen, Johanna

Advantages and Disadvantages of Scrum from Development Team's Point of View: IT Service Company Case Study

Jyväskylä: University of Jyväskylä, 2013, 83 p.

Information Systems Science, Master's Thesis

Supervisor: Pirhonen, Maritta

Scrum is a software development framework, which is developed to ease especially the management of complex software development projects. Scrum is one of the most popular and well-known software development framework and various small, mid-sized and large companies around the world use Scrum. Purpose of this master's thesis is to introduce advantages and disadvantages of Scrum based on the knowledge gained from scientific sources and case study carried out in a certain IT service company. Advantages and disadvantages of Scrum are examined from the development team's point of view, which is responsible of creating a software product.

Keywords: agile methods, Scrum, development team

KUVIOT

KUVIO 1 Scrumin tapahtumat ja tuotokset.....	16
KUVIO 2 Esimerkki tuotteen kehitysjonosta.....	17
KUVIO 3 Esimerkki sprintin tehtävällystasta	18
KUVIO 4 Esimerkki sprintin edistymiskäyrästä.....	19

TAULUKOT

TAULUKKO 1 Kirjallisuuskatsauksessa esille tulleet scrumin hyödyt ja haasteet kehitystiimin näkökulmasta.....	28
TAULUKKO 2 Haastateltavien tausta.....	37
TAULUKKO 3 Tapaustutkimuksessa esille tulleet scrumin hyödyt ja haasteet kehitystiimin näkökulmasta.....	60

SISÄLLYS

TIIVISTELMÄ	2
ABSTRACT	3
KUVIOT	4
TAULUKOT	4
SISÄLLYS.....	5
1 JOHDANTO.....	7
1.1 Tutkielman tarkoitus.....	8
1.2 Tutkielman rakenne	9
2 KETTERÄT MENETELMÄT	10
2.1 Ketterien menetelmien tausta ja määritelmät.....	10
2.2 Ketterien menetelmien arvot.....	11
2.3 Ketterien menetelmien periaatteet	12
2.4 Yhteenveto	13
3 SCRUM	14
3.1 Scrumin tausta ja määritelmät	14
3.2 Scrumin roolit.....	15
3.3 Scrumin tapahtumat ja tuotokset	16
3.4 Yhteenveto	20
4 SCRUMIN HYÖDYT JA HAASTEET KEHITYSTIIMIN NÄKÖKULMASTA.....	21
4.1 Scrumin käyttöönotto.....	21
4.2 Kehitystiimin koko	22
4.3 Kehitystiimin työskentely-ympäristö	22
4.4 Kehitystiimin monitaitoisuus.....	23
4.5 Kehitystiimin itseohjautuvuus.....	23
4.6 Kehitystiimin yhteistyö.....	24
4.7 Kehitystiimin työn läpinäkyvyys	24
4.8 Kehitystiimin työmäärä ja työskentelytahti.....	25
4.9 Kehitystiimin tuottaman dokumentaation määrä	26
4.10 Kehitystiimin yhteistyö tuoteomistajan kanssa.....	27
4.11 Vaatimusten muuttuminen scrum-projektin aikana	27
4.12 Yhteenveto	28

5	TAPAUSTUTKIMUS	33
5.1	Tutkimusmenetelmä	33
5.2	Tiedonkeruumenetelmä.....	34
5.3	Aineiston analysointimenetelmä	34
5.4	Tutkimuksen rajaukset.....	35
5.5	Tutkimuskohde	36
5.6	Haastateltavien tausta	37
5.7	Tutkimuksen kulku	38
5.8	Tutkimuksen luotettavuus	39
5.9	Yhteenveto	40
6	TAPAUSTUTKIMUKSEN TUTKIMUSTULOKSET	41
6.1	Scrumin käyttöönotto.....	41
6.2	Kehitystiimin koko	43
6.3	Kehitystiimin työskentely-ympäristö	45
6.4	Kehitystiimin monitaitoisuus.....	46
6.5	Kehitystiimin itseohjautuvuus.....	48
6.6	Kehitystiimin yhteistyö.....	49
6.7	Kehitystiimin työn läpinäkyvyys	51
6.8	Kehitystiimin työmäärä ja työskentelytahti.....	53
6.9	Kehitystiimin tuottaman dokumentaation määrä	55
6.10	Kehitystiimin yhteistyö tuoteomistajan kanssa.....	57
6.11	Vaatimusten muuttuminen scrum-projektin aikana	58
6.12	Yhteenveto	60
7	TULOKSET JA JOHTOPÄÄTÖKSET.....	64
7.1	Tutkimustulosten vertailu	64
7.2	Scrumin hyödyt ja haasteet kehitystiimin näkökulmasta.....	68
7.3	Scrumin haasteiden ratkaiseminen ja hyötyjen saavuttaminen.....	69
7.4	Yhteenveto	73
8	YHTEENVETO	74
	LÄHTEET	76
	LIITE 1 HAASTATTELURUNKO.....	80

1 JOHDANTO

Ohjelmistokehitys on yhä useamman yrityksen tärkein kilpailuetu. Monet yritykset käyttävät kuitenkin vanhanaikaisia ohjelmistokehitysmenetelmiä. (Rally Software Development Corporation, Schwaber & Scrum Alliance, 2005.) Tämän päivän ohjelmistokehitysprojekteille on tyypillistä, että kehitettävää ohjelmistotuotetta koskevat vaatimukset muuttuvat ohjelmistokehitysprojektin edetessä. Myös teknologia- ja liiketoimintaympäristö voivat muuttua nopeasti ohjelmistokehitysprojektin aikana. (Williams & Cockburn, 2003.) Perinteiset ohjelmistokehitysmenetelmät soveltuvat ohjelmistokehitysprojektien toteuttamiseen silloin, kun ohjelmistotuotetta koskevien vaatimusten sekä teknologia- ja liiketoimintaympäristön voidaan ennustaa pysyvän muuttumattomina koko ohjelmistokehitysprojektin ajan (Boehm, 2002). Toisin kuin perinteiset ohjelmistokehitysmenetelmät, ketterät menetelmät on suunniteltu vastaamaan ohjelmistokehitysprojektin aikana ilmeneviin muutostarpeisiin (Williams & Cockburn, 2003).

Abrahamsson, Salo, Ronkainen ja Warsta (2002) tuovat esille, että on tärkeää tunnistaa ohjelmistokehitysprojektin luonne ja valita ohjelmistokehitysprojektin toteuttamiseen parhaiten soveltuva ohjelmistokehitysmenetelmä. Rehman ja Hussain (2007) korostavat, että valittu ohjelmistokehitysmenetelmä vaikuttaa merkittävästi ohjelmistokehitysprojektin onnistumiseen. Bradbury (2007) puolestaan huomauttaa, että on tärkeää, että yritykset ovat valmiita ottamaan riskejä ja kokeilemaan uusia ohjelmistokehitysmenetelmiä, sillä ohjelmistokehitysmenetelmän valinnalla voidaan vaikuttaa positiivisesti ohjelmistokehitysprojekteihin.

Scrum on ketteriin menetelmiin kuuluva ohjelmistokehityksen viitekehys, joka on kehitetty helpottamaan erityisesti monimutkaisten ohjelmistokehitysprojektien hallintaa ja toteuttamista (Schwaber & Sutherland, 2011; Sutherland, 2010). Se on ketteristä menetelmistä suosituin ja sen suosio on kasvanut erityisesti viime vuosien aikana (Overhage, Schlauderer, Birkmeier & Miller, 2011; Pries-Heje & Pries-Heje, 2011a). Scrumia käytetään sekä pienissä, keskisuurissa että suurissa IT-alan yrityksissä ympäri maailmaa. Esimerkiksi Yahoo!, Microsoft ja Google käyttävät scrumia ohjelmistokehitysprojektiansa toteuttamisen viitekehysenä. (Sutherland, 2010.) Scrumin viitekehityksen mukaisista ohjelmis-

tokehitysprojekteista käytetään tässä tutkielmassa nimitystä *scrum-projekti*. Ohjelmistotuotteesta puolestaan käytetään nimitystä *tuote*.

1.1 Tutkielman tarkoitus

Tämä pro gradu -tutkielma on jatkoa tutkijan syksyllä 2012 valmistuneelle kandidaatintutkielmalle (Hämäläinen, 2012), jossa tutkittiin scrumin hyötyjä ja haasteita asiakasta edustavan *tuoteomistajan* ja tuotteen kehittämisen vastuussa olevan *kehitystiimin* näkökulmasta. Tässä tutkielmassa näkökulma on rajattu koskemaan vain kehitystiimiä. Tutkielman tutkimuskysymys on **mitä hyötyjä ja haasteita scrumiin liittyy kehitystiimin näkökulmasta?**

Tutkielman yhteydessä toteutetaan kirjallisuuskatsaus, jonka tarkoitus on luoda käsitys scrumin hyödyistä ja haasteista kehitystiimin näkökulmasta yhdistelemällä aiheeseen liittyvien tieteellisten julkaisujen tarjoamaa tutkimustietoa. Kirjallisuuskatsauksen lähdemateriaalina käytetään scrumia ja ketteriä menetelmiä käsitteleviä artikkeleita, konferenssijulkaisuja, Internet-sivuja ja oppaita, joita on etsitty muun muassa Jyväskylän yliopiston informaatioteknologian tiedonhankintaportaalien tietokannoista sekä Google Scholar -hakukonetta apuna käyttäen. Lähdemateriaaliksi on valittu mahdollisimman tuoreita tieteellisiä julkaisuja.

Kirjallisuuskatsauksen lisäksi tutkielman yhteydessä toteutetaan tutkimus, jonka tarkoitus on selvittää, mitä hyötyjä ja haasteita erään IT-alan palveluyrityksen yhteen, käynnissä olevaan scrum-projektiin liittyy kehitystiimin näkökulmasta. Kyseessä on kvalitatiivinen eli laadullinen tutkimus. Kvalitatiivisena tutkimusmenetelmänä käytetään tapaustutkimusta. Tapaustutkimuksen avulla on mahdollista tuottaa tutkittavaan aiheeseen liittyvää yksityiskohtaista ja täsmällistä tietoa tapauksia, eli rajattuja tutkimuskohteita tutkimalla (Eriksson & Koistinen, 2005, 43). Tapaustutkimuksen tiedonkeruumenetelmänä käytetään teemahaastattelua. Teemahaastattelussa tutkimus etenee yksityiskohtaisten haastattelukysymysten sijaan tutkimuksen keskeisten teemojen, eli aihepiirien varassa (Hirsjärvi & Hurme, 2011, 48). Teemahaastattelujen teemat ja haastattelukysymykset muodostetaan tutkielman kirjallisuuskatsauksen aihepiirien ja tutkimustulosten pohjalta.

Scrumin hyödyillä tarkoitetaan lisäarvoa, jonka scrum tuo ohjelmistokehitysprojekteihin. Scrumin haasteilla puolestaan tarkoitetaan scrum-projekteihin liittyviä ongelmia. Scrumin hyötyihin tutustuminen on tärkeää, sillä hyödyt tiedostamalla yritykset voivat arvioida, kannattaako käytössä olevista ohjelmistokehitysmenetelmistä siirtyä scrumiin ja tarkastella, toteutuvatko scrumin hyödyt yrityksen omissa scrum-projekteissa. Myös scrumin haasteisiin tutustuminen on tärkeää, sillä scrumin haasteet tiedostamalla yritykset voivat ennustaa ja varautua scrumin käyttöönoton yhteydessä mahdollisesti ilmeneviin haasteisiin sekä tutkia, liittyykö yrityksen omiin scrum-projekteihin haasteita, jotka ovat mahdollisesti jääneet huomaamatta. Scrumin haasteiden tiedostami-

nen ja ratkaiseminen on tärkeää, jotta scrumin hyödyt ja hyötyjen tuoma lisäarvo voidaan saavuttaa.

Dybå ja Dingsøy (2008) tuovat tutkimuksessaan esille, että ketteriä menetelmiä ja erityisesti scrumia on tutkittu suosionsa nähden kohtuullisen vähän. Tämän lisäksi tutkimusten tutkimuskohteet, näkökulmat, tieteellinen taso ja laajuus eroavat hyvin paljon toisistaan (Dybå & Dingsøy, 2008). Wang, Lane, Conboy ja Pikkarainen (2009) kuitenkin huomauttavat, että ketteriä menetelmiä käsittelevien tutkimusten määrä kasvaa jatkuvasti. Ketteriä menetelmiä koskevia tutkimuksia ilmaantuukin säännöllisesti sekä ammatinharjoittajien että tutkijoiden toimesta (Wang ym., 2009). Ketterien menetelmien hyötyjä ja haasteita käsitteleviä tutkimuksia on tehty jonkin verran, mutta suurimmassa osassa tutkimuksista näkökulma on rajattu koskemaan vain tiettyä ketterää menetelmää ja tiettyyn aihepiiriin liittyviä hyötyjä ja haasteita (Dybå & Dingsøy, 2008). Tutkielma, jonka tarkoitus on luoda kokonaisvaltainen käsitys scrumin hyödyistä ja haasteista kehitystiimin näkökulmasta yhdistelemällä aikaisempien tutkimusten ja tieteellisten julkaisujen tarjoamaa tutkimustietoa voidaankin nähdä tarpeelliseksi ja ajankohtaiseksi.

1.2 Tutkielman rakenne

Tutkielman toinen, kolmas ja neljäs luku muodostavat tutkielman kirjallisuuskatsauksen. Toisessa luvussa esitellään ketterien menetelmien tausta ja määritelmät sekä kuvaillaan eri ketteriä menetelmiä yhdistävät arvot ja periaatteet. Kolmannessa luvussa esitellään ketteriin menetelmiin kuuluvan scrumin tausta ja määritelmät sekä kuvaillaan scrum-projekteille tyypilliset roolit, tapahtumat ja tuotokset. Neljännessä luvussa muodostetaan käsitys scrumin hyödyistä ja haasteista kehitystiimin näkökulmasta yhdistelemällä scrumia ja ketteriä menetelmiä käsittelevien tieteellisten julkaisujen tarjoamaa tutkimustietoa. Tutkielman viides ja kuudes luku käsittelevät tutkielman empiiristä osuutta, eli kohdeorganisaatiossa toteutettavaa tapaustutkimusta. Viidennessä luvussa kerrotaan valitusta tutkimusmenetelmästä, tiedonkeruumenetelmästä, aineiston analysointimenetelmästä, tutkimuksen rajauksista, tutkimuskohteesta, haastateltavien taustasta sekä tutkimuksen kulusta ja luotettavuudesta. Kuudennessä luvussa esitetään ja analysoidaan tapaustutkimuksen tutkimustulokset. Tutkielman seitsemännessä luvussa verrataan tutkielman kirjallisuuskatsauksen ja empiirisen osuuden tutkimustuloksia toisiinsa ja esitetään tutkimustulosten pohjalta vastaus tutkielman tutkimuskysymykseen *mitä hyötyjä ja haasteita scrumiin liittyy kehitystiimin näkökulmasta?* Seitsemännessä luvussa esitetään myös johtopäätöksiä siitä, miten tutkielmassa esille tulleet scrumin haasteet voidaan ratkaista ja tukea näin tutkielmassa esille tulleiden hyötyjen saavuttamista. Tutkielman viimeinen luku on yhteenveto, jossa luodaan selkeä kuva saavutetuista tutkimustuloksista ja niiden merkityksestä sekä esitetään jatkokäsittelemättä.

2 KETTERÄT MENETELMÄT

Tässä luvussa esitellään ketterien menetelmien tausta ja määritelmät sekä kuvaillaan eri ketteriä menetelmiä yhdistävät arvot ja periaatteet.

2.1 Ketterien menetelmien tausta ja määritelmät

Perinteisten ohjelmistokehitysmenetelmien mukaisissa ohjelmistokehitysprojekteissa kehitettävää tuotetta koskevat vaatimukset määritellään ja dokumentoidaan yhdessä asiakkaan kanssa heti ohjelmistokehitysprojektin alussa, ennen suunnittelu- ja toteutusvaiheita. Moni asiakas kuitenkin kokee vaatimusten esittämisen näin aikaisessa vaiheessa vaikeaksi tai jopa mahdottomaksi. Tämän päivän ohjelmistokehitysprojekteille on tyypillistä, että asiakas muuttaa kehitettävää tuotetta koskevia vaatimuksiaan ohjelmistokehitysprojektin edetessä. Myös teknologia- ja liiketoimintaympäristö voivat muuttua nopeasti ohjelmistokehitysprojektin aikana. (Williams & Cockburn, 2003.) Teknologia- ja liiketoimintaympäristön, kuten markkinoiden ja kilpailijoiden, nopean muuttumisen vuoksi loppukäyttäjät voivat vaatia kehitettävältä tuotteelta toiminnallisuuksia, joita ei ole suunniteltu toteutettaviksi ohjelmistokehitysprojektin alussa (Mellor, 2005). Kun asiakkaan vaatimukset ovat epäselvät ohjelmistokehitysprojektin alussa ja ohjelmistokehitykseen liittyy paljon epävarmuustekijöitä, syntyy tarve vaihtoehtoisille ohjelmistokehitysmenetelmille (Fernandez & Fernandez, 2008).

Ketterien menetelmien kehityksen taustalla on yritysmaailman tarve joustavammille ja nopeammille ohjelmistokehitysprojekteille (Highsmith & Cockburn, 2001). Zhi-gen, Quan ja Xi (2009) määrittelevät joustavuuden tarkoittavan vallitseviin olosuhteisiin sopeutumista ja muutostarpeisiin vastaamista nopeasti ja tehokkaasti. Ketterille menetelmille ei ole yhtä yhteistä määritelmää. Rehman ja Hussain (2007) määrittelevät ketterät menetelmät moderniksi ohjelmistokehitysmenetelmien perheeksi, joiden avulla hallitaan ohjelmistokehitysprojekteja. Zhi-gen ym. (2009) määrittelevät ketterät menetelmät joukoksi projektinhallin-

takäytänteitä, joiden avulla pyritään sopeutumaan jatkuviin muutoksiin ohjelmistokehitysprojekteissa. Ketterät menetelmät eroavat toisistaan, mutta tärkein niitä yhdistävä tekijä on pyrkimys vastata ohjelmistokehitysprojektien aikana ilmeneviin muutostarpeisiin mahdollisimman nopeasti (Williams & Cockburn, 2003).

Vuonna 2001 17 keskeistä ketterien menetelmien, joita silloin kutsuttiin kevyiksi menetelmiksi, edustajaa kokoontui yhteen keskustellakseen edustamiensa menetelmien yhteneväisyyksistä. Tilaisuuteen osallistuneet henkilöt nimitivät itseään *Agile-allianssiksi*. Tilaisuuden tarkoitus oli luoda yhteiset lähtökohdat eri ketterille menetelmille ja tilaisuuden aikaansaannoksena syntyiikin *Agile-manifesti*, joka sisälsi neljä arvoa ja 12 periaatetta, joita eri ketterien menetelmien täytyi noudattaa. (Largent, 2010.) Viime vuosien aikana ketterät menetelmät ovat saavuttaneet suosiota perinteisten ohjelmistokehitysmenetelmien rinnalla (Overhage ym., 2011; Pries-Heje & Pries-Heje, 2011b). Tunnettuja ketteriä menetelmiä ovat muun muassa scrum, Extreme Programming, Lean Development, Crystal Methods ja Adaptive Software Development (Highsmith & Cockburn, 2001).

2.2 Ketterien menetelmien arvot

Ketterien menetelmien arvot kuvaavat, mitä ketterien menetelmien mukaisessa ohjelmistokehityksessä pidetään tärkeänä verrattuna perinteiseen ohjelmistokehitykseen. Ketterien menetelmien arvoihin kuuluu (Agile Alliance, 2001):

1. arvostaa ihmisiä ja vuorovaikutusta enemmän kuin prosesseja ja työkaluja
2. arvostaa toimivaa ohjelmistoa enemmän kuin kattavaa dokumentaatiota
3. arvostaa asiakasyhteistyötä enemmän kuin sopimusneuvotteluja
4. arvostaa muutokseen vastaamista enemmän kuin pitäytymistä suunnitelmassa

Ketterien menetelmien näkökulmasta ohjelmistokehitysprojektiin osallistuvilla henkilöillä on suurin vaikutus ohjelmistokehitysprojektin onnistumiseen. Myös työskentelyprosessi ja työkalut nähdään tärkeänä osana ohjelmistokehitystä. Ilman ihmisiä ja heidän välistä vuorovaikutusta ohjelmistokehitysprojekti ei kuitenkaan voi onnistua. (Abbas, Gravell & Wills, 2008.)

Ohjelmistokehityksen tarkoitus on kehittää toimivia ohjelmistotuotteita, ei luoda dokumentaatiota. Parhaimmillaan dokumentaatio auttaa ymmärtämään, miksi tuote on kehitetty ja miten tuote toimii. (Ambler, 2005.) Ketterien menetelmien näkökulmasta dokumentaatio on kuitenkin vain lupaus kehitettävästä tuotteesta (Highsmith & Cockburn, 2001).

Asiakkaan voidaan olettaa muuttavan kehitettävää tuotetta koskevia vaatimuksia ohjelmistokehitysprojektin edetessä. Asiakasyhteistyön avulla pyritäänkin huomioimaan asiakkaan muuttuvat vaatimukset mahdollisimman ai-

kaisessa vaiheessa ohjelmistokehitysprojektia. (Ambler, 2005.) Myös asiakkaan kanssa käytävät sopimusneuvottelut ovat tärkeitä, sillä ne nähdään keinona luoda ja ylläpitää asiakassuhteita (Abrahamsson ym., 2002). Asiakkaan kanssa tehty sopimus ei kuitenkaan voi korvata asiakasyhteistyötä (Ambler, 2005).

Ketterien menetelmien näkökulmasta suunnitelmat kertovat, miten ohjelmistokehitysprojektin tavoitteeseen päästään, mutta ne eivät kerro, onko tavoiteltava suunta oikea, sillä suunnitelmissa ei ole huomioitu asiakkaan vaatimusten tai teknologia- ja liiketoimintaympäristön muuttumista ohjelmistokehitysprojektin aikana. Muutostarpeisiin vastaaminen suunnitelmassa pitäytymisen sijaan on tärkeää, jotta kehitettävä tuote vastaisi sitä koskevia lopullisia vaatimuksia. (Mellor, 2005.)

2.3 Ketterien menetelmien periaatteet

Ketterien menetelmien periaatteet kertovat, mitä ketterien menetelmien mukaiselta ohjelmistokehitykseltä edellytetään. Ketterien menetelmien periaatteisiin kuuluu (Agile Alliance, 2001):

1. toimittaa ohjelmistokehitysprojektin alusta lähtien asiakkaan vaatimukset täyttäviä versioita kehitettävästä tuotteesta
2. toimittaa tuoteversioita säännöllisesti, lyhyillä aikaväleillä, parin viikon tai kuukauden välein
3. ottaa vastaan muuttuvat, kehitettävää tuotetta koskevat vaatimukset myös ohjelmistokehitysprojektin loppupuolella
4. varmistaa, että liiketoiminnan edustajat ja ohjelmistokehitystiimi työskentelevät yhdessä päivittäin, koko ohjelmistokehitysprojektin ajan
5. valita ohjelmistokehitysprojekteihin motivoituneita henkilöitä, luottaa siihen että he tekevät työnsä ja tukea heidän työntekoaan tarjoamalla heille heidän tarvitsemansa puitteet ja tuen
6. nähdä kasvokkain käytävä keskustelu tehokkaimpana ja toimivimpana tapana välittää tietoa
7. tukea tasaista työskentelytahtia
8. tukea ohjelmistokehitystiimien itseohjautuvuutta
9. kannustaa ohjelmistokehitystiimien itsetarkastelua
10. nähdä toimiva ohjelmisto ensisijaisena edistyksen mittarina
11. panostaa tuotteen tekniseen laatuun ja hyvään rakenteeseen
12. panostaa yksinkertaisuuteen ja maksimoida tekemättä jätettävän työn määrä

Ketterien menetelmien mukainen ohjelmistokehitys on iteratiivista ja inkrementaalista. Tuote kehitetään useissa lyhyissä kehitysjaksoissa, eli iteraatioissa. Jokaisessa iteraatiossa suunnitellaan, toteutetaan, testataan ja dokumentoidaan tiettyjä tuotteen toiminnallisuuksia. Tuote kasvaa inkrementaalista, kun iteraatioissa toteutetut tuotteen toiminnallisuudet integroidaan osaksi tuotetta.

Jokaisen iteraation päätteeksi julkaistaan toimiva tuoteversio asiakkaan katselmoitavaksi. (Abbas ym., 2008.) Toimivan tuoteversion nähdessään asiakas näkee heti, mitä ohjelmistokehitystiimi todellisuudessa on kehittämässä ja vastaa-ko kehitettävä tuote sille asetettuja vaatimuksia (Highsmith & Cockburn, 2001). Iteratiivisen ja inkrementaalisen ohjelmistokehityksen ansiosta asiakas voi muuttaa tuotetta koskevia vaatimuksia ohjelmistokehitysprojektin aikana. Mitä aikaisemmin muutostarpeet huomioidaan, sitä helpompaa, nopeampaa ja edullisempaa niiden toteuttaminen on. Ohjelmistokehitysprojektin onnistumisen kannalta on tärkeää, että asiakas osallistuu ohjelmistokehitysprojektiin yhdessä ohjelmistokehitystiimin kanssa. (Coram & Bohner, 2005.)

Ketterien menetelmien mukaisissa ohjelmistokehitysprojekteissa pyritään järjestämään työskentely-ympäristö ja muut työskentelyyn liittyvät tekijät siten, että ne tukevat ohjelmistokehitysprojektiin osallistuvien henkilöiden välistä vuorovaikutusta ja yhteistyötä (Abrahamsson ym., 2002). Kasvokkain käytävän keskustelun ansiosta tiedon jakaminen ohjelmistokehitysprojektiin osallistuvien henkilöiden välillä on nopeaa (Coram & Bohner, 2005). Tasaisen työskentelytahtin ansiosta jokainen ohjelmistokehitysprojektiin osallistuva henkilö voi säilyttää työskentelytahtinsa pitkällä aikavälillä ja tuntea työnteon miellyttäväksi (Agile Alliance, 2001). Ohjelmistokehitystiimien itseohjautuvuus tarkoittaa sitä, ettei kukaan johda ohjelmistokehitystiimien toimintaa, vaan ohjelmistokehitystiimit päättävät itse työnjaosta ja työskentelytavoistaan (Schwaber & Sutherland, 2011; Sutherland, 2010). Ohjelmistokehitystiimien itsetarkastelu puolestaan tarkoittaa sitä, että ohjelmistokehitystiimit tarkastelevat säännöllisesti, kuinka parantaa tehokkuuttaan ja tuottavuuttaan ja sopeuttavat toimintaansa sen mukaisesti (Agile Alliance, 2001).

Ketterien menetelmien mukaisissa ohjelmistokehitysprojekteissa kehitettävää tuotetta testataan jatkuvasti, jotta tuotteeseen liittyvät virheet huomattaisiin mahdollisimman aikaisin. Tämä helpottaa virheiden korjaamista ja vähentää korjaamisen aiheuttamia viivästyksiä ja kustannuksia. Ohjelmistokehitystiimiä kannustetaan myös pitämään kehitettävä tuote mahdollisimman yksinkertaisena. Kun kehitettävä tuote on yksinkertainen, siihen liittyvien muutostarpeiden toteuttaminen on helppoa ja nopeaa. (Highsmith & Cockburn, 2001.)

2.4 Yhteenveto

Tämän luvun tarkoitus oli esitellä lyhyesti ketterien menetelmien tausta ja määritelmät sekä kuvailla eri ketteriä menetelmiä yhdistävät arvot ja periaatteet. Scrum kuuluu ketteriin menetelmiin ja kuten muiden ketterien menetelmien, myös scrumin mukainen ohjelmistokehitys noudattaa ketterien menetelmien arvoja ja periaatteita. Ketterien menetelmien taustaan, määritelmiin, arvoihin ja periaatteisiin tutustuminen onkin tärkeää, jotta ymmärrettäisiin, mistä scrum on lähtöisin.

3 SCRUM

Tässä luvussa esitellään ketteriin menetelmiin kuuluvan scrumin tausta ja määritelmät sekä kuvaillaan scrum-projekteille tyypilliset roolit, tapahtumat ja tuotokset. Tärkeimmät tässä luvussa lähteenä käytetyt tieteelliset julkaisut ovat Schwaberin ja Sutherlandin (2011) julkaisema scrum-opas, joka kuvaa scrumin viitekehyksen sekä Sutherlandin (2010) Scrum Training Instituten koulutusmateriaalien pohjalta kokoama scrum-ohjekirja, joka kuvaa scrumin roolit, tapahtumat ja tuotokset tarkemmalla tasolla. Scrumiin liittyvät käsitteet on suomennettu Eskelisen ym. (2012) laatiman suomenkielisen scrum-sanaston mukaisesti.

3.1 Scrumin tausta ja määritelmät

Termi scrum on alunperin lähtöisin rugbyista. Rugbyn näkökulmasta pieni joukkue, jolla on yhteinen tavoite ja joka tekee yhteistyötä tavoitteen saavuttamiseksi, saa aikaan parempia tuloksia. Tuhannet ihmiset ovat osallistuneet scrumin kehittämiseen, mutta suurimman työn ovat tehneet Ken Schwaber ja Jeff Sutherland, jotka esittelivät ja julkaisivat scrumin viitekehyksen vuonna 1995. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Scrum on ohjelmistokehityksen viitekehys, joka on kehitetty helpottamaan erityisesti monimutkaisten ohjelmistokehitysprojektien hallintaa ja toteuttamista. Se kuvaa pienen joukon rooleja, tapahtumia ja tuotoksia, jotka scrum-projekteissa täytyy esiintyä. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Scrumin viitekehystä noudattamalla ohjelmistokehitystiimin on mahdollista toteuttaa ohjelmistokehitysprojekti jatkuvasti muuttuvassa ympäristössä (Coram & Bohner, 2005). Kuten kaikki ketterät menetelmät, myös scrum soveltuu ohjelmistokehitysprojektien toteuttamiseen silloin, kun kehitettävää tuotetta koskevien vaatimusten tai teknologia- ja liiketoimintaympäristön voidaan ennustaa muuttuvan nopeasti ohjelmistokehitysprojektin aikana (Rally Software Development Corporation ym., 2005).

Scrumin lähtökohtana on ajatus siitä, että ohjelmistokehityksen pitäisi olla empiirinen prosessi. Empiiriselle prosessille on ominaista läpinäkyvyys, tarkastelu ja sopeuttaminen. Läpinäkyvyys tarkoittaa sitä, että kaikki osatekijät, jotka vaikuttavat ohjelmistokehitysprojektin lopputulokseen, ovat ohjelmistokehitysprojektiin osallistuvien henkilöiden tiedossa. Ohjelmistokehitysprojektiin osallistuvat henkilöt tarkastelevat näitä osatekijöitä jatkuvasti ohjelmistokehitysprojektin edetessä ja sopeuttavat omaa toimintaansa tarkastelun pohjalta saavuttaakseen toivotun lopputuloksen. (Cho, 2008.) Ohjelmistokehitysprojektit ovat haastavia ja kehitettävää tuotetta koskevat vaatimukset sekä teknologia- ja liiketoimintaympäristö muuttuvat nopeasti ohjelmistokehitysprojektin aikana. Onkin mahdollista ennustaa ohjelmistokehitysprojektin alussa, minkälainen ohjelmistokehitysprojektin lopuksi valmistuva tuote on, kuinka nopeasti tuote voidaan toteuttaa ja mitkä ovat tuotteen toteuttamisen lopulliset kustannukset. Scrum-projekteissa kehitettävää tuotetta ja sitä koskevia vaatimuksia sekä scrum-projektin aikataulua ja kustannuksia tarkastellaan jatkuvasti ja scrum-projektiin osallistuvat henkilöt sopeuttavat omaa toimintaansa tarkastelun pohjalta. (Rally Software Development Corporation ym., 2005.)

Kuten kaikkien ketterien menetelmien, myös scrumin mukaisissa ohjelmistokehitysprojekteissa tuote kehitetään iteratiivisesti ja inkrementaalisesti. Tuote suunnitellaan, toteutetaan, testataan ja dokumentoidaan lyhyissä kehitysjaksoissa, joita kutsutaan sprinteiksi. Jokaisen sprintin päätteeksi valmistuu tiettyjä toiminnallisuuksia sisältävä toimiva versio kehitettävästä tuotteesta. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Toimivan tuoteversion nähdessään asiakkaan edustaja näkee heti, vastaako kehitettävä tuote sille asetettuja vaatimuksia ja voi ohjata kehitystiimin työskentelyä nopeasti oikeaan suuntaan puuttamalla kehitystiimin mahdollisiin väärinymmärryksiin ja muuttamalla tarvittaessa tuotetta koskevia vaatimuksia (Mann & Maurer, 2005).

Scrum on yksi maailman suosituimmista ja tunnetuimmista ohjelmistokehityksen viitekehyksistä. Scrumia käytetään sekä pienissä, keskisuurissa että suurissa IT-alan yrityksissä ympäri maailmaa. Esimerkiksi Yahoo!, Microsoft ja Google käyttävät scrumia ohjelmistokehitysprojektiensa toteuttamisen viitekehystenä. (Sutherland, 2010.)

3.2 Scrumin roolit

Scrum-projekteissa esiintyy kolme eri roolia: tuoteomistaja, kehitystiimi ja scrummaster. Näissä eri rooleissa työskentelevät henkilöt muodostavat yhdessä *scrumtiimin*. Kehitettävän tuotteen parissa voi työskennellä useita rinnakkaisia scrumtiimejä. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

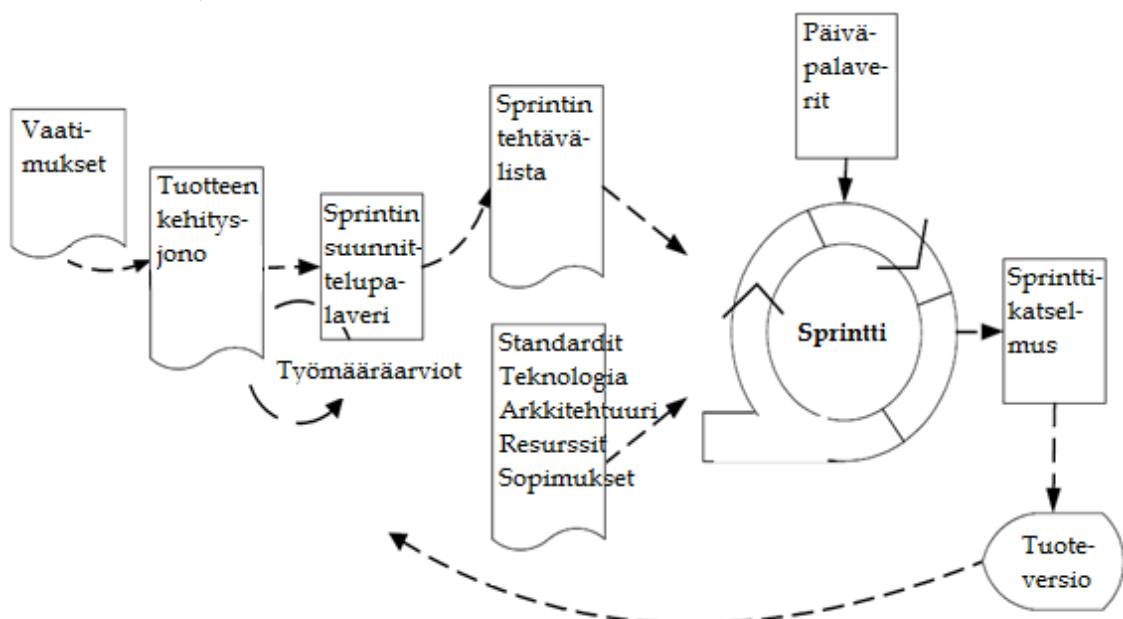
Tuoteomistaja on henkilö, joka tekee kehitettävää tuotetta koskevat päätökset. Tuoteomistaja edustaa asiakkaan näkemystä. Hän listaa ja priorisoi asiakkaan näkökulmasta tärkeimmät kehitettävää tuotetta koskevat vaatimukset, jotka kehitystiimin täytyy toteuttaa scrum-projektin aikana. Tuoteomistajana toimii aina vain yksi henkilö. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Kehitystiimi työskentelee tuoteomistajan asettamien vaatimusten parissa. Kehitystiimi muodostuu pienestä joukosta ammattilaisia, joilla on tarvittava osaaminen luoda tuoteomistajan asettamia vaatimuksia vastaava valmis tuote ilman ulkopuolista apua. Kukaan ei johda kehitystiimin toimintaa, vaan kehitystiimi päättää itse työnjaosta ja työskentelytavoistaan. Kehitystiimien voidaan sanoa olevan monitaitoisia ja itseohjautuvia. Kehitystiimiin kuuluu tyypillisesti 3-9 samassa paikassa työskentelevää jäsentä. Jokaisen kehitystiimin jäsenen titteli on kehittäjä työnkuvasta riippumatta ja vastuu tuotteen kehittämisestä kuuluu koko kehitystiimille. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Scrummasterin tärkein tehtävä on tarjota scrumtiimille scrumin viitekehykseen liittyvää koulutusta ja ohjausta ja varmistaa, että scrumtiimi noudattaa viitekehystä. Tämän lisäksi scrummaster huolehtii siitä, että kehitystiimi voi tehdä työtään optimaalisella tavalla. Kehittäjät raportoivat scrummasterille päivittäin ongelmista, jotka hidastavat heidän työntekoaan. Scrummasterin tehtävä on ratkaista nämä ongelmat ja varmistaa, että jokainen kehittäjä edistyy työtehtävissään. Scrummaster huolehtii myös siitä, etteivät ulkopuoliset tahot häiritse kehitystiimin työskentelyä scrum-projektin aikana. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

3.3 Scrumin tapahtumat ja tuotokset

Roolien lisäksi scrum-projekteissa esiintyy myös erilaisia tapahtumia ja tuotoksia (kuvio 1). Jokainen tapahtuma ja tuotos tarjoaa mahdollisuuden tarkastella ja kontrolloida scrum-projektin etenemistä. (Schwaber & Sutherland, 2011; Sutherland, 2010.)



KUVIO 1 Scrumin tapahtumat ja tuotokset (mukaillen Abrahamsson ym., 2002, 35)

Scrum-projekti lähtee liikkeelle kehitettävää tuotetta koskevien alustavien vaatimusten määrittelystä. Vaatimusten määrittelyyn voivat osallistua useat eri sidosryhmät, kuten asiakas, loppukäyttäjät, tuoteomistaja, kehitystiimi, asiantuntijat sekä markkinoinnin ja myynnin edustajat. Vaatimukset esitetään usein *käyttäjätarinoina*, jotka ovat lyhyitä ja selkeitä kuvauksia vaadituista tuotteen toiminnallisuuksista. (Sutherland, 2010.)

Scrum-projektin ensimmäinen tuotos on *tuotteen kehitysjo* (kuvio 2). Tuotteen kehitysjo on priorisoitu lista kehitettävän tuotteen toiminnallisuuksista, jotka kehitystiimin täytyy toteuttaa scrum-projektin aikana. Tuoteomistaja listaa ja priorisoi eri sidosryhmien esittämät, tuotteen toiminnallisuuksia koskevat vaatimukset tuotteen kehitysjoon. Toiminnallisuudet priorisoidaan sitä korkeammalle tuotteen kehitysjoon, mitä suurempi arvo niillä on asiakkaan tai liiketoiminnan näkökulmasta. Tuotteen kehitysjoon voidaan merkitä myös parannukset ja korjaukset, jotka tuotteeseen täytyy tehdä. Tuoteomistaja päivittää tuotteen kehitysjoa jatkuvasti scrum-projektin edetessä. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Toiminnallisuus	Lisätiedot	Prioriteetti	Arvo	Työmäärä-arvio (hpt)	Sprintti ja jäljellä oleva työmäärä (hpt)					
					1	2	3	4	5	6
Ostajan roolissa haluan voida lisätä kirjan ostoskärryyn	...	1	7	5						
Ostajan roolissa haluan voida poistaa kirjan ostoskärrystä	...	2	6	2						
Tapahtumien käsittelyn suorituskykyä täytyy parantaa	...	3	6	13						
Luottokorttien validointia täytyy nopeuttaa	...	4	6	20						
...										

KUVIO 2 Esimerkki tuotteen kehitysjonosta (mukaillen Sutherland, 2010, 18)

Scrum-projektin ensimmäinen tapahtuma on *sprintin suunnittelupalaveri*. Se on scrummasterin järjestämä, korkeintaan kahdeksan tuntia kestävä, kaksivaiheinen palaveri, johon osallistuu koko scrumtiimi. Sprintin suunnittelupalaverissa suunnitellaan alkavan sprintin sisältö. Suunnittelupalaverin ensimmäisellä puoliskolla tuoteomistaja esittelee scrumtiimille priorisoidun tuotteen kehitysjonon, jonka jälkeen kehitystiimi arvioi, mitä tuotteen kehitysjoon korkeimmalle priorisoituja tuotteen toiminnallisuuksia se ehtii toteuttaa alkavan sprintin aikana. Arvion pohjalta scrumtiimi valitsee yhdessä ne toiminnallisuudet, jotka sprintin päätteeksi julkaistavan tuoteversion täytyy sisältää. Suunnittelupalaverin toisella puoliskolla kehitystiimi suunnittelee tarvittavat työtehtävät valittujen toiminnallisuuksien toteuttamiseksi ja tekee työtehtävien toteuttamisen työmääräarviot. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Sprintin suunnittelupalaverin tuotoksena syntyy *sprintin tehtäväl* (kuvio 3). Sprintin tehtäväl on lista sprintin aikana toteutettavista tuotteen toi-

minnallisuuksista sekä suoritettavista työtehtävistä, työtehtävien toteuttajista, työmääräarvioista ja jäljellä olevasta työmäärästä työtunteina. Sprintin aikana toteutettaviksi valittuja tuotteen toiminnallisuuksia ei voida muuttaa kesken sprintin, mutta kehittäjät päivittävät muita sprintin tehtävälisan kohtia jatkuvasti sprintin edetessä. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

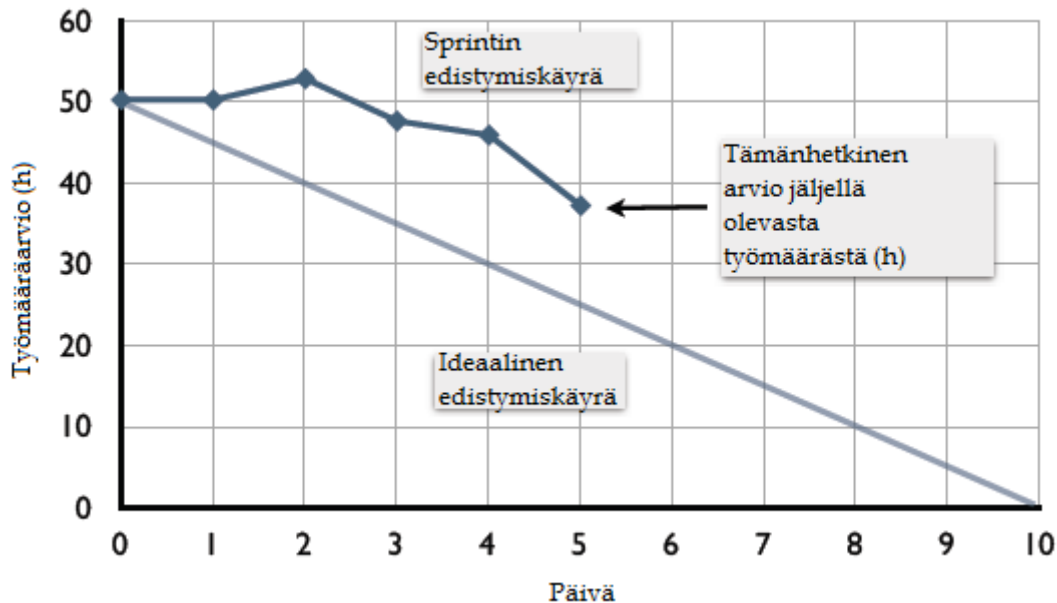
Tuotteen kehitysjonon toiminnallisuus	Työtehtävä	Toteuttaja	Työmäärä-arvio (h)	Päivä ja jäljellä oleva työmäärä (h)					
				1	2	3	4	5	6
Ostajan roolissa haluan voida lisätä kirjan ostoskärryyn	Muokkaa tietokantaa	Sarjay	5	4	3	0	0	0	
	Luo Internet-sivu	Jing	3	3	3	2	0	0	
	Kirjoita automaattiset testit	Sarah	5	5	5	5	5	0	
...	...								
...									

KUVIO 3 Esimerkki sprintin tehtävälisistä (mukaillen Sutherland, 2010, 25)

Sprintin tehtävälisan laatimisen jälkeen käynnistyy scrum-projektin toinen tapahtuma, *sprintti*. Se on lyhyt kehitysjakso, jonka aikana kehitystiimi suunnittelee, toteuttaa, testaa ja dokumentoi sprintin aikana toteutettaviksi valitut tuotteen toiminnallisuudet. Sprintin pituus on 1-4 viikkoa ja pituus säilyy samana koko scrum-projektin ajan. Sprintin kesto ei voida pidentää, vaan sprintti päättyy sovittuna päivänä huolimatta siitä, onko kehitystiimi ehtinyt toteuttaa valitut tuotteen toiminnallisuudet. Uusi sprintti alkaa välittömästi edellisen päätyttyä. Tuoteomistaja voi keskeyttää sprintin ja aloittaa uuden sprintin suunnittelupalaverin, jos sprintin toteuttaminen loppuun asti on tarpeetonta. Näin saatetaan käydä esimerkiksi teknologia- ja liiketoimintaympäristön muuttuessa sprintin aikana. Sprinttien keskeyttäminen on kuitenkin hyvin harvinaista. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Sprintin aikana scrummaster, kehitystiimi ja tarvittaessa myös tuoteomistaja kokoontuvat päivittäin lyhyeen, korkeintaan 15 minuuttia kestäväan *päiväpalaveriin*. Päiväpalaveri on scrum-projektin kolmas tapahtuma. Se pidetään scrummasterin johdolla samaan aikaan, samassa paikassa, jokaisena työpäivänä. Päiväpalaverissa jokainen kehittäjä kertoo vuorollaan, mitä on saanut aikaan edellisen päiväpalaverin jälkeen, mitä suunnittelee tekevänsä seuraavaan päiväpalaveriin mennessä ja mitkä tekijät hidastavat työntekoa. Päiväpalavereiden tarkoitus on tarkastella kehitystiimin työn ja sprintin edistymistä sekä sprintin tehtävälisan toteutumista. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Päiväpalavereiden lisäksi sprintin edistymistä voidaan seurata *sprintin edistymiskäyrän* avulla (kuviokuva 4). Sprintin edistymiskäyrä on kuvaaja, joka kuvaa sprintin jäljellä olevan kokonaistyömäärän työtunteina. Se muodostetaan sprintin tehtävälisan pohjalta ja sitä päivitetään jatkuvasti sprintin edetessä. Scrum-projektin edistymistä puolestaan voidaan seurata *tuotteen edistymiskäyrän* avulla. Sprintin ja tuotteen edistymiskäyrät ovat jokaisen scrum-projektiin osallistuvan henkilön nähtävissä. (Sutherland, 2010.)



KUVIO 4 Esimerkki sprintin edistymiskäyrästä (mukaillen Sutherland, 2010, 25)

Jokaisen sprintin tuotoksena valmistuu käyttö- ja julkaisukelpoinen, testattu ja dokumentoitu versio kehitettävästä tuotteesta. *Tuoteversiot* rakentuvat aikaisempien tuoteversioiden päälle. Scrumtiimi määrittelee tuoteversioille *valmiin määritelmän*, joka tarkoittaa scrumtiimin yhteistä sopimusta siitä, milloin tuoteversio on valmis. Valmiin määritelmään voi sisältyä esimerkiksi tuoteversion testaus ja dokumentointi. Jokaisen sprintin tavoitteena on toteuttaa tuoteversio, joka noudattaa valmiin määritelmää. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Sprintin lopussa scrumtiimi järjestää *sprinttikatselmuksen*, joka on korkeintaan neljä tuntia kestävä palaveri, jossa kehitystiimi esittelee toteuttamansa tuoteversion tuoteomistajalle ja muille sidosryhmille. Sprinttikatselmus on scrumprojektin neljäs tapahtuma. Sen tarkoitus on saada palautetta kehitetyistä tuoteversiosta sekä tietoa muuttuneista tuotteen toiminnallisuuksista koskevista vaatimuksista ja teknologia- ja liiketoimintaympäristön muutoksista ja luoda näin pohja seuraavalle sprintin suunnittelupalaverille. Sprinttikatselmuksen pohjalta scrumtiimi päättää, mitä tuotteen kehitysjonon toiminnallisuuksia lähdetään toteuttamaan seuraavassa sprintissä. Tuoteomistaja voi myös muuttaa tuotteen kehitysjonoa tai todeta tuotteen valmiiksi sprinttikatselmuksen pohjalta. Jos tuoteomistaja katsoo tuoteversion vastaavan kaikkia sille asetettuja vaatimuksia ja olevan valmis käyttöönotettavaksi, uutta sprinttiä ei käynnistetä, vaan tuote julkaistaan välittömästi (Schwaber & Sutherland, 2011; Sutherland, 2010.)

Sprinttikatselmuksen jälkeen scrumtiimi järjestää *sprintin retrospektiivin*, joka on korkeintaan kolme tuntia kestävä palaveri, jonka tarkoitus on tarkastella, mitkä asiat päättyneessä sprintissä sujuivat hyvin ja mitä asioita tulisi parantaa. Sprintin retrospektiivi on scrumprojektin viides tapahtuma. Tarkastelun kohteena voi olla esimerkiksi scrumtiimin yhteistyö, työskentelyprosessi tai

käytetyt työkalut. Sprintin retrospektiivin lopuksi scrumtiimi luo suunnitelman parannusten toteuttamiseksi seuraavassa sprintissä. (Schwaber & Sutherland, 2011; Sutherland, 2010.)

3.4 Yhteenveto

Tämän luvun tarkoitus oli esitellä lyhyesti scrumin tausta ja määritelmät sekä kuvailla scrumin viitekehysten mukaiset roolit, tapahtumat ja tuotokset. Scrumin taustaan, määritelmiin ja viitekehykseen tutustuminen on tärkeää, jotta ymmärrettäisiin, minkälaisen ohjelmistokehitysprojektien toteuttamiseen scrum soveltuu ja mitä scrum-projekteihin osallistuvilta henkilöiltä edellytetään.

4 SCRUMIN HYÖDYT JA HAASTEET KEHITYSTIIMIN NÄKÖKULMASTA

Tämän luvun tarkoitus on muodostaa käsitys scrumin hyödyistä ja haasteista kehitystiimin näkökulmasta yhdistelemällä scrumia ja ketteriä menetelmiä käsittelevien tieteellisten julkaisujen tarjoamaa tutkimustietoa. Kirjallisuuskatsauksessa tuodaan esille vain sellaisia scrumiin liittyviä hyötyjä ja haasteita, jotka voivat koskea eri scrum-projekteja ja kehitystiimejä muuttuvista tekijöistä huolimatta. Kirjallisuuskatsauksessa ei tutkitakaan esimerkiksi scrumin tapahtumiin ja tuotoksiin liittyviä hyötyjä ja haasteita, sillä niiden toteutustapa voi erota hyvin paljon eri scrum-projektien ja scrumtiimien välillä. Dybå ja Dingsøyr (2008) tuovat esille, että ketterien menetelmien hyötyjä ja haasteita käsittelevissä tutkimuksissa näkökulma on usein rajattu koskemaan vain tiettyyn aihepiiriin, kuten ketterän menetelmän käyttöönottoon tai ohjelmistokehitystiimin itseohjautuvuuteen, liittyviä hyötyjä ja haasteita. Tämän kirjallisuuskatsauksen aihepiirit, joihin liittyviä hyötyjä ja haasteita kehitystiimin näkökulmasta tutkitaan, on koottu lähdemateriaalissa esiintyneistä eri aihepiireistä. Kirjallisuuskatsauksen tutkimustulokset esitetään aihepiireittäin ja tiivistetään taulukkomuotoon luvun yhteenvedossa.

4.1 Scrumin käyttöönotto

Scrum on yksinkertainen ja helposti opittava viitekehys. Se kuvaa pienen joukon rooleja, tapahtumia ja tuotoksia, jotka scrum-projekteissa täytyy esiintyä. Kehitystiimi voi käyttää vapaasti valitsemiaan prosesseja, menetelmiä, tekniikoita ja työkaluja scrumin viitekehysten sisällä. (Schwaber & Sutherland, 2011.) Scrum voidaan ottaa käyttöön joko ohjelmistokehitysprojektin alussa tai kesken ohjelmistokehitysprojektin. Oikein noudatettuna scrum tuo useita eri hyötyjä ohjelmistokehitysprojekteihin. (Rally Software Development Corporation ym., 2005.) Kehittäjät myös pitävät työskentelystä scrum-projekteissa (Overhage ym., 2011; Zhi-gen ym., 2009).

Scrum eroaa huomattavasti perinteisistä ohjelmistokehitysmenetelmistä ja sen käyttöönotto edellyttää usein suuria muutoksia. Scrumin käyttöönotto voi olla työlästä, aikaavievää ja haastavaa, mikä puolestaan voi aiheuttaa muutostavastarintaa kehitystiimin keskuudessa. (Rally Software Development Corporation ym., 2005; Schatz & Abdelshafi, 2005.) Kehittäjät voivat vastustaa scrumin käyttöönottoa erityisesti silloin, kun kehittäjillä ei ole aikaisempaa kokemusta scrumista tai kun yritys ei tarjoa kehitystiimille scrumiin liittyvää koulutusta, tuo esille scrumiin liittyviä positiivisia kokemuksia ja kannusta kehitystiimiä scrumin käyttöönotossa. Se, haluaako kehitystiimi ottaa scrumin käyttöön, vaikuttaa merkittävästi scrum-projektien onnistumiseen. (Conboy, Coyle, Wang & Pikkarainen, 2011; Overhage ym., 2011.) Suuri virhe, jonka scrumitiimi voi tehdä scrumin käyttöönoton yhteydessä on muuttaa scrumin viitekehystä sen sijaan, että muuttaisi omaa toimintaansa viitekehysten edellyttämällä tavalla. Jos scrumitiimi muuttaa scrumin viitekehystä tai käyttää vain osaa viitekehysten rooleista, tapahtumista ja tuotoksista, scrumin hyödyt voivat jäädä saavuttamatta. (Sutherland, 2010.) Schwaber ja Sutherland (2011) huomauttavat, ettei myös voida puhua scrum-projektista, jos ohjelmistokehitysprojekti ei noudata scrumin viitekehystä.

4.2 Kehitystiimin koko

Kehitystiimi on optimaalisen kokoinen silloin, kun se on riittävän pieni jakaakseen tietoa nopeasti, mutta myös riittävän suuri tehdäkseen vaaditun määrän työtä. Scrum soveltuu parhaiten pienten kehitystiimien käyttöön. Kehitystiimiin kuuluukin tyypillisesti 3-9 kehittäjää. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Kehitystiimin pieni koko mahdollistaa suoran ja jatkuvan vuorovaikutuksen sekä nopean tiedon jakamisen kaikkien kehittäjien välillä (Ceschi, Sillitti, Succi & Panfilis, 2005; Overhage ym., 2011).

Yrityksillä ei välttämättä ole tarpeeksi resursseja muodostaa optimaalisen kokoisia kehitystiimejä (Pikkarainen & Wang, 2011). Jos kehitystiimiin kuuluu vähemmän kuin kolme kehittäjää, vuorovaikutus ja osaaminen sekä tämän seurauksena myös kehitystiimin tehokkuus ja tuottavuus vähenevät. Jos kehitystiimiin kuuluu enemmän kuin yhdeksän kehittäjää, kehitystiimin toiminta vaatii liikaa koordinoitua. (Schwaber & Sutherland, 2011.) Jos kehittäjät puolestaan toimivat samaan aikaan useissa eri projekteissa, heidän voi olla vaikea osallistua ja sitoutua scrum-projektiin sekä säilyttää tasainen työskentelytahti (Moe, Dingsøyr & Dyba, 2009; Pikkarainen & Wang, 2011).

4.3 Kehitystiimin työskentely-ympäristö

Kehitystiimejä kannustetaan työskentelemään avoimessa työtilassa. Avoimessa työtilassa kehittäjät eivät ole eristyksissä toisistaan, vaan he näkevät ja kuulevat

toisensa kokoajan. Työskentely avoimessa työtilassa mahdollistaa kasvokkain käytävän keskustelun ja tukee kehittäjien välistä vuorovaikutusta ja yhteistyötä. (Cho, 2008.)

Avoimessa työtilassa on kuitenkin paljon häiriötekijöitä, jotka voivat vaikeuttaa kehittäjien työntekoa. Esimerkiksi ympärillä käytävän keskustelun vuoksi kehittäjien voi olla vaikea keskittyä työntekoon. Tämä puolestaan saattaa vaikuttaa negatiivisesti kehitystiimin tehokkuuteen ja tuottavuuteen. (Cho, 2008.)

4.4 Kehitystiimin monitaitoisuus

Kehitystiimin monitaitoisuus tarkoittaa sitä, että kehitystiimi muodostuu pienestä joukosta ammattilaisia, joilla on tarvittava osaaminen toteuttaa scrum-projekti ilman ulkopuolista apua. Kehittäjät voivat toteuttaa sekä työtehtäviä, joiden toteuttamiseen heillä on tarvittava osaaminen että työtehtäviä, joiden toteuttamisen he haluavat oppia esimerkiksi työskentelemällä yhdessä muiden kehittäjien kanssa. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Kehittäjiä kannustetaan osallistumaan erilaisten työtehtävien toteuttamiseen. Työskentely monitaitoisessa kehitystiimissä tukee kehittäjien välistä tiedon jakamista ja parantaa kehittäjien osaamista. Osaamisen kehittyessä scrum-projektin eteneminen ei ole kiinni ainoastaan yhden, tietyn työtehtävän asiantuntijaksi erikoistuneen kehittäjän työpanoksesta. (Monteiro ym., 2011; Pikkarainen & Wang, 2011.)

Koska kehitystiimit ovat pieniä, kehittäjiltä vaaditaan usein laajaa, esimerkiksi tuotteen suunnitteluun, toteutukseen ja testaamiseen liittyvää osaamista (Conboy ym., 2011; Coram & Bohner, 2005; Moe ym., 2009; Pikkarainen & Wang, 2011). Scrum-projekteihin voi kuitenkin olla vaikea löytää kehittäjiä, joilla on tarpeeksi laaja osaaminen ja kokeneemmat kehittäjät voivat joutua neuvomaan kokemattomampia kehittäjiä, mikä vie aikaa varsinaiselta työnteolta. Laaja osaaminen sumentaa myös kehittäjien rooleja. Jos kehittäjille ei tarjota mahdollisuutta erikoistua tietyn työtehtävän asiantuntijoiksi, heidän asemansa urakehitys- ja rekrytointitilanteissa voi kärsiä. (Conboy ym., 2011.) Jos yrityksissä puolestaan arvostetaan kehittäjien erityisosaamista tai kehittäjät haluavat säilyttää oman erityisosaamisensa, kehittäjien välinen tiedon jakaminen voi olla vähäistä (Moe ym., 2009).

4.5 Kehitystiimin itseohjautuvuus

Kehitystiimin itseohjautuvuus tarkoittaa sitä, ettei kukaan johda kehitystiimin toimintaa, vaan kehitystiimi päättää itse työnjaosta ja työskentelytavoistaan (Schwaber & Sutherland, 2011; Sutherland, 2010). Työskentely itseohjautuvassa kehitystiimissä lisää kehittäjien mahdollisuuksia vaikuttaa omiin työtehtäviinsä ja parantaa kehitystiimin toiminnan koordinoitua. Kehitystiimin itseohjautu-

vuuden ansiosta myös kehittäjien motivaatio, sitoutuminen scrum-projektiin, tehokkuus ja tuottavuus kasvaa. (Monteiro ym., 2011; Overhage ym., 2011.)

Scrummasterin ja muiden scrum-projektin sidosryhmien voi olla vaikea antaa työnjakoon ja työskentelytapoihin liittyvä päätöksentekovalta kehitystiimille ja he voivat pyrkiä tekemään päätöksiä aivan kuten perinteisten ohjelmistokehitysmenetelmien mukaisissa ohjelmistokehitysprojekteissa. Kehittäjien voi puolestaan olla vaikea ottaa vastuu työnjakoon ja työskentelytapoihinsa liittyvästä päätöksenteosta, jos scrumista ei ole aikaisempaa kokemusta. Tällöin kehitystiimin itseohjautuvuus ei toteudu. (Conboy ym., 2011; Coram & Bohner, 2005; Marchenko & Abrahamsson, 2008; Moe ym., 2009; Pikkarainen & Wang, 2011.)

4.6 Kehitystiimin yhteistyö

Scrum-projekteissa vastuu tuotteen kehittamisestä kuuluu koko kehitystiimille. Kehittäjät tekevätkin paljon yhteistyötä tuotteen kehitysjonossa määriteltyjen tuotteen toiminnallisuuksien toteuttamisen parissa. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Kehittäjät voivat myös halutessaan toteuttaa työtehtäviä yhdessä muiden kehittäjien kanssa (Sutherland, 2010). Kehittäjien tekemä yhteistyö parantaa kehittäjien välistä vuorovaikutusta ja tukee ryhmähengen muodostumista (Ceschi ym., 2005; Conboy ym., 2011; Figueiredo, 2009; Overhage ym., 2011; Pikkarainen & Wang, 2011; Rising & Janof, 2000). Highsmith ja Cockburn (2001) korostavat, että tekemällä yhteistyötä kehittäjät pystyvät myös tuottamaan parempia tuloksia kuin työskentelemällä yksin. Lisäksi keskittymisen yksilösuoritusten sijaan kehitystiimin yhteiseen suoritukseen vähentää kehittäjiin kohdistuvia paineita (Overhage ym., 2011).

Kehittäjät voivat kuitenkin haluta työskennellä itsenäisesti, jolloin vuorovaikutus ja yhteistyö kehittäjien välillä voi olla vähäistä, mikä vaikuttaa negatiivisesti myös ryhmähengen muodostumiseen (Moe ym., 2009). Kehittäjillä voi olla myös huonot vuorovaikutustaidot, minkä vuoksi vuorovaikutus ja yhteistyö kehittäjien välillä voi olla vaikeaa, mikä puolestaan voi estää myös ryhmähengen muodostumisen. (Conboy ym., 2011; Coram & Bohner, 2005.) Jos kehittäjät eivät tee yhteistyötä, heidän työnsä tulokset voi olla vaikea yhdistää. Tämä puolestaan vaikuttaa negatiivisesti kehitystiimin tehokkuuteen ja tuottavuuteen. (Marchenko & Abrahamsson, 2008.)

4.7 Kehitystiimin työn läpinäkyvyys

Kehitystiimin työn läpinäkyvyys tarkoittaa sitä, että jokainen kehittäjä tietää, mitä muut kehittäjät tekevät. Päiväpalavereiden, sprintin tehtävälistan ja sprintin edistymiskäyrän ansiosta kehittäjät ovat jatkuvasti tietoisia toistensa tekemästä työstä, sprintin etenemisestä ja etenemistä hidastavista tekijöistä.

(Schwaber & Sutherland, 2011; Sutherland, 2010.) Koska kehitystiimin työtä ja sprinttien edistymistä seurataan jatkuvasti erilaisten scrumin tapahtumien ja tuotosten avulla, kehittäjät ovat motivoituneita aikaansaamaan jotain joka päivä (Felker, Slamova & Davis, 2012). Päiväpalavereissa kehittäjät voivat raportoida kohtaamistaan ongelmista heti niiden ilmaannuttua ja pyytää apua muilta kehittäjiltä tai scrummasterilta, jos he eivät pysty etenemään työtehtävissään. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Ongelmien raportointi ja avun pyytäminen heti ongelmien ilmaannuttua vähentää huomattavasti tuotteeseen liittyviä virheitä, helpottaa ja nopeuttaa virheiden korjaamista ja auttaa kehitystiimiä pysymään aikataulussa. Työn läpinäkyvyys mahdollistaakin laadukkaamman tuotteen nopean kehittämisen. (Coram & Bohner, 2005; Li, Moe & Dybå, 2010; Sutherland, 2010; Zhi-gen ym., 2009.) Schatz ja Abdelshafi (2005) tuovat esille, että scrum-projekteissa asiakkaan raportointien, tuotteeseen liittyvien virheiden määrä vähenee huomattavasti, minkä ansiosta kehitystiimi voi keskittyä seuraavassa sprintissä uusien tuotteen toiminnallisuuden toteuttamiseen virheiden korjaamisen sijaan. Työn läpinäkyvyys kasvattaa myös kehittäjien luottamusta toisiinsa ja scrum-projektin onnistumiseen (Pries-Heje & Pries-Heje, 2011b; Rising & Janof, 2000).

Kehittäjät voivat kuitenkin pelätä, että työn läpinäkyvyys paljastaa heidän heikkoutensa. Työn läpinäkyvyyden vuoksi heikon osaamisen ja huonon työn jäljen peittely onkin vaikeaa. Toisinaan työn läpinäkyvyys voi johtaa mahtailuun ja kiusaamiseen kehittäjien keskuudessa, minkä seurauksena työskentelyilmapiiri voi muuttua negatiiviseksi tai jopa vihamieliseksi. Negatiivinen työskentelyilmapiiri puolestaan johtaa pahimmillaan siihen, että kehittäjät lähtevät kehitystiimistä tai irtisanoutuvat yrityksestä. Jos kehittäjät lähtevät kehitystiimistä kesken scrum-projektin, kehitystiimin tehokkuus ja tuottavuus heikkenee. (Conboy ym., 2011; Sutherland, 2010.) Työn läpinäkyvyys voi myös kasvattaa kehittäjiin kohdistuvia paineita, sillä päiväpalavereiden vuoksi jokainen poikkeaminen sovitusta työtehtävistä tai aikataulusta on kaikkien nähtävillä (Li ym., 2010; Rising & Janof, 2000). Joskus kehittäjien voi olla vaikea ymmärtää, mitä muut kehittäjät tekevät. Tietyn työtehtävän asiantuntijoiksi erikoistuneet kehittäjät voivatkin tuntea scrumin tapahtumiin osallistumisen ja tekemänsä työn esittelyn muille kehittäjille epämiellyttäväksi, jos he eivät usko muiden kehittäjien ymmärtävän tekemäänsä työtä, eivät tarvitse apua työtehtäviensä kanssa, eivätkä osaa auttaa muita kehittäjiä heidän työtehtävissään. (Marchenko ja Abrahamsson, 2008.)

4.8 Kehitystiimin työ määrä ja työskentelytahti

Sprintin suunnittelupalavereissa kehitystiimi suunnittelee alkavan sprintin aikana toteutettavat työtehtävät ja tekee työtehtävien toteuttamisen työmääräarviot. Sprintin suunnittelupalaverin tuotoksena syntyy sprintin tehtävälista. Päiväpalavereissa varmistetaan, että kehitystiimi työskentelee ainoastaan sprintin tehtävälissä määriteltyjen työtehtävien parissa. (Schwaber & Sutherland,

2011; Sutherland, 2010.) Työskentely sprinteissä auttaa pitämään kehitystiimin työmäärän kohtuullisena koko scrum-projektin ajan. Tämä takaa kehitystiimille tasaisen työskentelytahdin. (Felker ym., 2012; Schatz & Abdelshafi, 2005.) Tasaisen työskentelytahdin ansiosta kehittäjät kokevat työnannon miellyttäväksi (Pikkarainen & Wang, 2011). Myös ylitöiden määrä vähenee huomattavasti scrumin ansiosta. (Mann & Maurer, 2005; Schatz & Abdelshafi, 2005).

Kehitystiimin voi kuitenkin olla vaikea arvioida työtehtävien viemää aikaa ja tehdä realistisia työmääräarvioita, jos työtehtävät ovat haastavia, niistä ei ole aikaisempaa kokemusta tai ne vaativat uusien taitojen opettelua (Felker ym., 2012). Kehitystiimi voi myös sitoutua tekemään liian paljon työtä yhden sprintin aikana, jolloin kehittäjät saattavat joutua tekemään ylitöitä (Marchenko & Abrahamsson, 2008; Moe ym., 2009). Kehittäjiltä edellytetään osallistumista scrumin eri tapahtumiin. Jos kehittäjät, scrummaster tai tuoteomistaja ovat valmistautuneet tapahtumiin huonosti, tai tapahtumat eivät noudata scrumin viitekehystä, ne voivat pitkittyä ja viedä aikaa varsinaiselta työnteolta. (Mann & Maurer, 2005; Marchenko & Abrahamsson, 2008; Pikkarainen & Wang, 2011.) Scrumin tapahtumat ovat aikarajattuja, eli ne kestävät sovitun ajan tapahtuman sisällöstä riippumatta. Scrumin tapahtumille voikin olla varattu liikaa tai liian vähän aikaa käsiteltävien asioiden määrään ja aikaavievyyteen nähden. Joskus myös sopivan ajankohdan löytäminen scrumin tapahtumille voi olla vaikeaa. (Cho, 2008.)

4.9 Kehitystiimin tuottaman dokumentaation määrä

Scrum ei ota kantaa siihen, minkälaista dokumentaatiota kehitystiimin pitäisi tuottaa (Coram & Bohner, 2005; Overhage ym., 2011). Scrum-projekteissa kehitystiimi ei yleensä tuota dokumentaatiota, jota jouduttaisiin päivittämään scrum-projektin edetessä (Selic, 2009). Kehittäjien ei tarvitse tehdä esimerkiksi vaatimusmäärittelyjä tai projektisuunnitelmia, joissa ei muuttuvien vaatimusten ja teknologia- ja liiketoimintaympäristön muutosten vuoksi olisi mahdollista pitäytyä (Zhi-gen ym., 2009). Scrum-projekteissa tuotetun dokumentaation määrä onkin vähäinen, mikä säästää kehittäjien aikaa ja parantaa näin kehitystiimin tehokkuutta ja tuottavuutta. Dokumentaation sijaan kehitystiimin keskuudessa syntyy hiljaista tietoa kehitystiimin yhteistyön myötä. (Coram & Bohner, 2005; Overhage ym., 2011.) Dokumentaation vähäisyys kannustaa kehittäjiä jakamaan scrum-projektiin liittyvää tietoa toisillensa (Cao & Ramesh, 2008; Cho, 2008).

Dokumentaation vähäisyys aiheuttaa kuitenkin ongelmia, jos kehittäjät vaihtuvat tai lähtevät kehitystiimistä kesken scrum-projektin. Tällöin oleellista tietoa saattaa kadota. Ilman dokumentaatiota myös uusien tai kokemattomampien kehittäjien perehdyttäminen voi olla vaikeaa. (Cao & Ramesh, 2008; Cho, 2008; Conboy ym., 2011; Coram & Bohner, 2005; Overhage ym., 2011.)

4.10 Kehitystiimin yhteistyö tuoteomistajan kanssa

Scrum-projekteissa tuoteomistaja osallistuu ohjelmistokehitykseen yhdessä kehitystiimin kanssa (Cao & Ramesh, 2008). Tuoteomistajan osallistuminen ohjelmistokehitykseen on tärkeää, jotta kehitystiimi ymmärtäisi, mitä asiakas haluaa (Ceschi ym., 2005). Kehitystiimi tekeekin säännöllistä yhteistyötä tuoteomistajan kanssa ja saa tuoteomistajalta jatkuvasti palautetta tekemästään työstä. Yhteistyön ansiosta sekä kehitystiimi että tuoteomistaja ovat varmoja siitä, että kehitettävä tuote vastaa asiakkaan vaatimuksia ja on käyttökelpoinen. Näin sekä kehitystiimin että tuoteomistajan tyytyväisyys kehitettävää tuotetta kohtaan kasvaa. (Cao & Ramesh, 2008; Mann & Maurer, 2005.) Säännöllisen yhteistyön ansiosta kehitystiimi voi myös kysyä nopeasti selvennystä tai neuvoa tuoteomistajalta esimerkiksi tuotetta koskeviin vaatimuksiin liittyen (Cao & Ramesh, 2008; Cho, 2008).

Kehittäjiltä vaaditaan teknisen osaamisen lisäksi asiakkaan toimialaan ja liiketoimintaan liittyvää tietoa. Jos kehittäjät eivät tunne asiakkaan toimialaa tai liiketoimintaa, tuoteomistaja ei välttämättä usko kehitystiimin pystyvän toteuttamaan tuotetta, joka tuottaa arvoa asiakkaalle. (Conboy ym., 2011; Pries-Heje & Pries-Heje, 2011b.) Kehittäjillä pitäisi olla myös hyvät vuorovaikutus- ja esiintymistaidot. Jos kehittäjien vuorovaikutus- ja esiintymistaidot ovat huonot, yhteistyö tuoteomistajan kanssa voi olla vaikeaa. (Conboy ym., 2011.) Jos tuoteomistajalla on muita päivittäisiä työtehtäviä tai hän on kiireinen, yhteistyö kehitystiimin kanssa voi vähentyä (Cho, 2008). Jos tuoteomistaja ei ole sitoutunut tekemään yhteistyötä kehitystiimin kanssa ja osallistumaan scrumin eri tapahdumiin, scrum-projekti voi epäonnistua (Felker ym., 2012; Overhage ym., 2011).

4.11 Vaatimusten muuttuminen scrum-projektin aikana

Scrum-projekteissa kehitettävää tuotetta koskevat vaatimukset eivät ole ennalta määriteltäviä, vaan tuoteomistaja voi muuttaa kehitettävää tuotetta koskevia vaatimuksia scrum-projektin edetessä. Sprintin aikana toteutettavia tuotteen toiminnallisuuksia ei kuitenkaan voida muuttaa kesken sprintin, vaan tuoteomistaja voi tuoda esille muuttuneet vaatimukset sekä teknologia- ja liiketoimintaympäristön muutokset sprintin lopuksi järjestettävässä sprinttikatselmuksessa. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Näin kehitystiimi voi olla varma siitä, ettei sprintin sisältö muutu kesken sprintin ja keskittyä sprintin tehtävälistan toteuttamiseen sekä huomioida ja vastata muuttuneisiin vaatimuksiin ja teknologia- ja liiketoimintaympäristön muutoksiin nopeasti, heti seuraavassa sprintissä (Sutherland, 2010). Muutostarpeiden nopean huomioidamisen ansiosta kehitetty tuote vastaa paremmin asiakkaan sekä teknologia- ja liiketoimintaympäristön tarpeita ja vaatimuksia (Cao & Ramesh, 2008; Figueiredo, 2009). Mitä aikaisemmin kehitystiimi huomioi muutostarpeet, sitä hel-

pompaa, nopeampaa ja edullisempaa niiden toteuttaminen on (Cao & Ramesh, 2008; Coram & Bohner, 2005).

Rising ja Janof (2000) tuovat esille, että perinteisiä ohjelmistokehitysmenetelmiä käyttäneet kehittäjät voivat uskoa pystyvänsä selvittämään kehitettävää tuotetta koskevat vaatimukset jo scrum-projektin alussa ja suhtautua negatiivisesti vaatimusten muuttumiseen scrum-projektin aikana. Jos tuoteomistaja on määritellyt tuotetta koskevat vaatimukset huonosti, vaatimusten analysointi voi viedä paljon kehittäjien aikaa. Tällöin kehitystiimi saattaa suhtautua negatiivisesti erityisesti muuttuvien vaatimusten toteuttamiseen. (Cho, 2008; Marchenko & Abrahamsson, 2008; Pikkarainen & Wang, 2011.)

4.12 Yhteenveto

Tämän luvun tarkoitus oli muodostaa käsitys scrumin hyödyistä ja haasteista kehitystiimin näkökulmasta yhdistelemällä scrumia ja ketteriä menetelmiä käsittelevien tieteellisten julkaisujen tarjoamaa tutkimustietoa. Scrumin hyödyillä tarkoitetaan lisäarvoa, jonka scrum tuo ohjelmistokehitysprojekteihin. Scrumin haasteilla puolestaan tarkoitetaan scrum-projekteihin liittyviä ongelmia. Scrumin hyötyihin tutustuminen on tärkeää, sillä hyödyt tiedostamalla voidaan arvioida, kannattaako käytössä olevista ohjelmistokehitysmenetelmistä siirtyä scrumiin ja tarkastella, toteutuvatko hyödyt omissa scrum-projekteissa. Myös scrumin haasteisiin tutustuminen on tärkeää, sillä scrumin haasteet tiedostamalla voidaan ennustaa ja varautua scrumin käyttöönoton yhteydessä mahdollisesti ilmeneviin ongelmiin sekä tutkia, liittyykö omiin scrum-projekteihin haasteita, jotka ovat jääneet huomaamatta. Scrumin haasteiden tiedostaminen ja ratkaiseminen on tärkeää, jotta scrumin hyödyt ja hyötyjen tuoma lisäarvo voidaan saavuttaa. Kirjallisuuskatsauksessa esille tulleet scrumin hyödyt ja haasteet kehitystiimin näkökulmasta on tiivistetty seuraavaan taulukkoon (taulukko 1).

TAULUKKO 1 Kirjallisuuskatsauksessa esille tulleet scrumin hyödyt ja haasteet kehitystiimin näkökulmasta

	Hyödyt ja haasteet
Scrumin käyttöönotto	<ul style="list-style-type: none"> + Scrum on yksinkertainen ja helposti opittava viitekehys. + Kehitystiimi voi käyttää vapaasti valitsemaansa prosesseja, menetelmiä, tekniikoita ja työkaluja scrumin viitekehysten sisällä. + Scrum voidaan ottaa käyttöön ohjelmistokehitysprojektin alussa tai kesken ohjelmistokehitysprojektin. + Oikein noudatettuna scrum tuo useita hyötyjä ohjelmistokehitysprojekteihin. + Kehittäjät pitävät työskentelystä scrum-projekteissa. - Scrumin käyttöönotto voi olla työlästä, aikaavievää ja haastavaa. - Kehittäjät voivat vastustaa scrumin käyttöönottoa. - Jos scrumitiimi muuttaa scrumin viitekehystä tai käyttää vain osaa viitekehysten rooleista, tapahtumista ja tuotoksista, scrumin hyödyt voi-

	vat jäädä saavuttamatta.
Kehitystiimin koko	<ul style="list-style-type: none"> + Scrum soveltuu parhaiten pienten kehitystiimien käyttöön. + Kehitystiimin pieni koko mahdollistaa suoran ja jatkuvan vuorovaikutuksen sekä nopean tiedon jakamisen kaikkien kehittäjien välillä. - Yrityksillä ei välttämättä ole tarpeeksi resursseja muodostaa optimaalisen kokoisia kehitystiimejä. - Jos kehitystiimi on liian pieni, sen vuorovaikutus, osaaminen, tehokkuus ja tuottavuus vähenevät. - Jos kehitystiimi on liian suuri, sen toiminta vaatii liikaa koordinoitua. - Jos kehittäjät toimivat samaan aikaan useissa eri projekteissa, heidän voi olla vaikea osallistua ja sitoutua scrum-projektiin ja säilyttää tasainen työskentelytahti.
Kehitystiimin työskentelyympäristö	<ul style="list-style-type: none"> + Kehitystiimejä kannustetaan työskentelemään avoimessa työtilassa. + Työskentely avoimessa työtilassa mahdollistaa kasvokkain käytävän keskustelun ja tukee kehittäjien välistä vuorovaikutusta ja yhteistyötä. - Avoimessa työtilassa on paljon häiriötekijöitä, jotka voivat vaikeuttaa kehittäjien työntekoa ja vaikuttaa negatiivisesti kehitystiimin tehokkuuteen ja tuottavuuteen.
Kehitystiimin monitaitoisuus	<ul style="list-style-type: none"> + Kehitystiimi muodostuu pienestä joukosta ammattilaisia, joilla on tarvittava osaaminen toteuttaa scrum-projekti ilman ulkopuolista apua. + Kehittäjät voivat toteuttaa sekä työtehtäviä, joiden toteuttamiseen heillä on tarvittava osaaminen että työtehtäviä, joiden toteuttamisen he haluavat oppia. + Työskentely monitaitoisessa kehitystiimissä tukee kehittäjien välistä tiedon jakamista ja parantaa kehittäjien osaamista. + Osaamisen kehittyessä scrum-projektin eteneminen ei ole kiinni ainoastaan yhden kehittäjän työpanoksesta. - Koska kehitystiimit ovat pieniä, kehittäjiltä vaaditaan usein laajaa osaamista, mikä sumentaa kehittäjien rooleja. - Scrum-projekteihin voi olla vaikea löytää kehittäjiä, joilla on tarpeeksi laaja osaaminen. - Kokeneemmat kehittäjät voivat joutua neuvomaan kokemattomampia kehittäjiä, mikä vie aikaa varsinaiselta työnteolta. - Jos kehittäjille ei tarjota mahdollisuutta erikoistua tietyn työtehtävän asiantuntijoiksi, heidän asemansa urakehitys- ja rekrytointitilanteissa voi kärsiä. - Jos yrityksissä arvostetaan kehittäjien erityisosaamista tai kehittäjät haluavat säilyttää oman erityisosaamisensa, kehittäjien välinen tiedon jakaminen voi olla vähäistä.
Kehitystiimin itseohjautuvuus	<ul style="list-style-type: none"> + Kukaan ei johda kehitystiimin toimintaa, vaan kehitystiimi päättää itse työnsä ja työskentelytavoistaan. + Työskentely itseohjautuvassa kehitystiimissä lisää kehittäjien mahdollisuuksia vaikuttaa omiin työtehtäviinsä ja parantaa kehitystiimin toiminnan koordinoitua. + Työskentely itseohjautuvassa kehitystiimissä parantaa kehittäjien motivaatiota, sitoutumista scrum-projektiin, tehokkuutta ja tuottavuutta. - Jos scrumista ei ole aikaisempaa kokemusta, scrummaster ja muut scrum-projektin sidosryhmät voivat vastustaa päätöksentekovallan antamista kehitystiimille ja kehitystiimin voi olla vaikea ottaa vastuu päätöksenteosta.
Kehitystiimin yhteistyö	<ul style="list-style-type: none"> + Vastuu tuotteen kehittämisestä kuuluu koko kehitystiimille ja kehittäjät tekevät paljon yhteistyötä tuotteen toteuttamisen parissa.

	<ul style="list-style-type: none"> + Kehittäjät voivat toteuttaa työtehtäviä yksin tai yhdessä muiden kehittäjien kanssa. + Kehittäjien tekemä yhteistyö parantaa kehittäjien välistä vuorovaikutusta ja tukee ryhmähengen muodostumista. + Tekemällä yhteistyötä kehittäjät pystyvät tuottamaan parempia tuloksia kuin työskentelemällä yksin. + Keskittyminen kehittäjien yksilösuoritusten sijaan kehitystiimin yhteiseen suoritukseen vähentää kehittäjiin kohdistuvia paineita. - Jos kehittäjät haluavat työskennellä itsenäisesti tai heidän vuorovaikutustaitonsa ovat huonot, kehittäjien välinen vuorovaikutus ja yhteistyö voi olla vähäistä ja vaikeaa, mikä voi estää ryhmähengen muodostumisen. - Jos kehittäjät eivät tee yhteistyötä, heidän työnsä tulokset voi olla vaikea yhdistää, mikä vaikuttaa negatiivisesti kehitystiimin tehokkuuteen ja tuottavuuteen.
Kehitystiimin työn läpinäkyvyys	<ul style="list-style-type: none"> + Kehittäjät ovat jatkuvasti tietoisia toistensa tekemästä työstä, sprintin etenemisestä ja etenemistä hidastavista tekijöistä. + Kehitystiimin työn ja sprinttien edistymisen jatkuva seuranta motivoi kehittäjiä aikaansaamaan jotain joka päivä. + Kehittäjät voivat raportoida kohtaamistaan ongelmista ja pyytää apua muilta kehittäjiltä tai scrummasterilta nopeasti, mikä vähentää tuotteen liittyviä virheitä, helpottaa ja nopeuttaa virheiden korjaamista ja auttaa kehitystiimiä pysymään aikataulussa. + Työn läpinäkyvyys mahdollistaa laadukkaan tuotteen nopean kehittämisen. + Työn läpinäkyvyys kasvattaa kehittäjien luottamusta toisiinsa ja scrum-projektin onnistumiseen. - Työn läpinäkyvyyden vuoksi kehittäjien heikko osaaminen, huono työnjälki sekä sovitusta työtehtävistä ja aikatauluista poikkeaminen ilmenevät nopeasti, mikä voi kasvattaa kehittäjiin kohdistuvia paineita. - Työn läpinäkyvyys voi johtaa mahtailuun ja kiusaamiseen kehittäjien keskuudessa, jonka seurauksena työskentelyilmapiiri voi muuttua negatiiviseksi tai vihamieliseksi, mikä johtaa pahimmillaan kehittäjien lähtemiseen kehitystiimistä tai irtisanoutumiseen yrityksestä. - Kehittäjät voivat tuntea scrumin tapahtumiin osallistumisen ja tekemänsä työn esittelyn muille kehittäjille epämukavaksi, jos he eivät usko muiden kehittäjien ymmärtävän tekemäänsä työtä, eivät tarvitse apua työtehtäviensä kanssa, eivätkä osaa auttaa muita kehittäjiä heidän työtehtävissään.
Kehitystiimin työmäärä ja työskentelytahti	<ul style="list-style-type: none"> + Sprinttien ansiosta kehitystiimin työmäärä pysyy kohtuullisena koko scrum-projektin ajan, mikä takaa kehitystiimille tasaisen ja miellyttävän työskentelytahdin. + Scrumin ansiosta ylitöiden määrä vähenee huomattavasti. - Kehitystiimin voi olla vaikea tehdä realistisia työmääräarvioita, jos työtehtävät ovat haastavia, niistä ei ole aikaisempaa kokemusta tai ne vaativat uusien taitojen opettelua. - Kehitystiimi voi sitoutua tekemään liian paljon työtä yhden sprintin aikana, jolloin kehittäjät saattavat joutua tekemään ylitöitä. - Scrumin tapahtumat voivat pitkittyä, jos niihin on valmistauduttu huonosti tai ne eivät noudata scrumin viitekehystä. - Scrumin tapahtumille voi olla varattu liikaa tai liian vähän aikaa käsi-

	<p>teltäviin asioihin nähden.</p> <ul style="list-style-type: none"> - Sopivan ajankohdan löytäminen scrumin tapahtumille voi olla vaikeaa.
Kehitystiimin tuottaman dokumentaation määrä	<ul style="list-style-type: none"> + Scrum-projekteissa tuotetun dokumentaation määrä on vähäinen, mikä säästää kehittäjien aikaa ja parantaa kehitystiimin tehokkuutta ja tuotavuutta. + Dokumentaation sijaan kehitystiimin keskuudessa syntyy hiljaista tietoa kehitystiimin yhteistyön myötä. + Dokumentaation vähäisyys kannustaa kehittäjiä jakamaan scrum-projektiin liittyvää tietoa toisillensa. - Ilman dokumentaatiota oleellista tietoa voi kadota, jos kehittäjät vaihtuvat tai lähtevät kehitystiimistä kesken scrum-projektin. - Ilman dokumentaatiota uusien tai kokemattomampien kehittäjien perehdyttäminen voi olla vaikeaa.
Kehitystiimin yhteistyö tuoteomistajan kanssa	<ul style="list-style-type: none"> + Kehitystiimi tekee säännöllistä yhteistyötä tuoteomistajan kanssa ja saa tuoteomistajalta jatkuvasti palautetta tekemästään työstä. + Säännöllisen yhteistyön ansiosta kehitystiimi ja tuoteomistaja ovat varmoja siitä, että kehitettävä tuote vastaa asiakkaan vaatimuksia, jolloin myös kehitystiimin ja tuoteomistajan tyytyväisyys kehitettävää tuotetta kohtaan kasvaa. + Säännöllisen yhteistyön ansiosta kehitystiimi voi kysyä nopeasti selvennystä tai neuvoa tuoteomistajalta. - Jos kehittäjien asiakkaan toimialaan ja liiketoimintaan liittyvä tieto tai vuorovaikutus- ja esiintymistaidot ovat huonot, yhteistyö tuoteomistajan kanssa voi olla vaikeaa. - Jos tuoteomistajalla on muita päivittäisiä työtehtäviä tai hän on kiireinen, yhteistyö kehitystiimin kanssa voi vähentyä. - Jos tuoteomistaja ei ole sitoutunut tekemään yhteistyötä kehitystiimin kanssa ja osallistumaan scrumin eri tapahtumiin, scrum-projekti voi epäonnistua.
Vaatimusten muuttuminen scrum-projektin aikana	<ul style="list-style-type: none"> + Tuoteomistaja voi muuttaa tuotetta koskevia vaatimuksia scrum-projektin edetessä. + Kehitystiimi voi olla varma siitä, ettei tuotetta koskevat vaatimukset muutu kesken sprintin. + Kehitystiimin on mahdollista vastata muuttuneisiin vaatimuksiin ja teknologia- ja liiketoimintaympäristön muutoksiin nopeasti, heti seuraavassa sprintissä. + Muutostarpeiden nopean huomioimisen ansiosta kehitettävä tuote sekä asiakkaan ja teknologia- ja liiketoimintaympäristön tarpeet ja vaatimukset vastaavat paremmin toisiaan. + Muutostarpeiden nopean huomioimisen ansiosta muutosten toteuttaminen on helpompaa, nopeampaa ja edullisempaa. - Perinteisiä ohjelmistokehitysmenetelmiä käyttäneet kehittäjät voivat uskoa pystyvänsä selvittämään kehitettävää tuotetta koskevat vaatimukset jo scrum-projektin alussa ja suhtautua negatiivisesti vaatimusten muuttumiseen scrum-projektin aikana. - Jos tuoteomistaja on määritellyt tuotetta koskevat vaatimukset huonosti, vaatimusten analysointi voi viedä paljon kehittäjien aikaa, jolloin kehitystiimi saattaa suhtautua negatiivisesti erityisesti muuttuvien vaatimusten toteuttamiseen.

Kirjallisuuskatsaus luo pohjan tutkielman empiiriselle osuudelle. Empiirinen osuus muodostuu tapaustutkimuksesta, jonka yhteydessä toteutettavien teemahaastattelujen teemat ja haastattelukysymykset muodostetaan kirjallisuuskatsauksen aihepiirien ja tutkimustulosten pohjalta. Tutkielman Tulokset ja johtopäätökset -luvussa (luku 7) verrataan kirjallisuuskatsauksen ja empiirisen osuuden tutkimustuloksia toisiinsa ja esitetään tutkimustulosten pohjalta vastaus tutkielman tutkimuskysymykseen *mitä hyötyjä ja haasteita scrumiin liittyy kehitystiimin näkökulmasta?*

5 TAPAUSTUTKIMUS

Tässä luvussa kuvaillaan tutkielman empiiristä osuutta. Empiirinen osuus muodostuu tapaustutkimuksesta, jonka tarkoitus on selvittää tutkielman kirjallisuuskatsauksen aihepiirien ja tutkimustulosten pohjalta, mitä hyötyjä ja haasteita erään IT-alan palveluyrityksen yhteen, käynnissä olevaan scrum-projektiin liittyy kehitystiimin näkökulmasta. Tässä luvussa kerrotaan tarkemmin valitusta tutkimusmenetelmästä, tiedonkeruumenetelmästä, aineiston analysointimenetelmästä, tutkimuksen rajauksista, tutkimuskohteesta, haastateltavien taustasta sekä tutkimuksen kulusta ja luotettavuudesta.

5.1 Tutkimusmenetelmä

Kvalitatiivisen eli laadullisen tutkimuksen avulla pyritään löytämään tai paljastamaan tosiasioita tutkimuskohteesta (Hirsjärvi, Remes & Sajavaara, 2007, 157). Tämän tutkielman yhteydessä toteutettavan tutkimuksen tarkoitus on paljastaa kohdeorganisaation yhteen, käynnissä olevaan scrum-projektiin liittyviä hyötyjä ja haasteita kehitystiimin näkökulmasta. Tutkimus on siis kvalitatiivinen.

Kvalitatiivisena tutkimusmenetelmänä käytetään tapaustutkimusta. Tapaustutkimuksessa tarkastelun kohteena on yksi tai useampi tapaus, joka tarkoittaa rajattavissa olevaa tutkimuskohdetta. Tapaustutkimuksen tarkoitus ei ole luoda yleistyksiä tai tilastollisia säännönmukaisuuksia, vaan tuottaa tutkitavaan ilmiöön liittyvää yksityiskohtaista ja täsmentävää tietoa tapauksia tutkimalla. (Eriksson & Koistinen, 2005, 13-15, 43.) Tapaustutkimus voi olla luonteeltaan kuvailevaa, teoriaa testaavaa tai uutta teoriaa luovaa (Järvinen & Järvinen, 2011, 74). Tämän tutkielman yhteydessä toteutettavassa tapaustutkimuksessa tutkittavia tapauksia on yksi, sillä tutkimuskohteeksi on valittu vain yksi kohdeorganisaation scrum-projekti. Lisäksi tapaustutkimus on luonteeltaan kuvailevaa, sillä sen tarkoitus on kuvailla tutkimuskohteeksi valittuun scrum-projektiin liittyviä hyötyjä ja haasteita kehitystiimin näkökulmasta.

5.2 Tiedonkeruumenetelmä

Tutkimushaastattelun avulla voidaan selvittää haastateltavan tutkittavaan ilmiöön liittyviä ajatuksia, käsityksiä, kokemuksia ja tunteita (Hirsjärvi & Hurme, 2011, 41). Tutkimushaastattelun etu on se, että haastattelija voi säädellä tiedonhankintaa tilanteen edellyttämällä tavalla ja haastateltavaa myötäillen. Haastattelija voi esimerkiksi pyytää haastateltavaa perustelevaan, tarkentamaan tai selventämään vastauksiaan. (Hirsjärvi ym., 2007, 200.)

Tutkimushaastattelut voidaan jakaa kolmeen eri ryhmään sen mukaan, miten tarkasti haastattelukysymykset on suunniteltu ennalta. Strukturoidussa haastattelussa haastattelija on määritellyt haastattelukysymyksiin valmiit vastaukset, joiden joukosta haastateltavan täytyy valita sopivin vastausvaihtoehto. Avoimessa haastattelussa haastattelija ja haastateltava käyvät avointa keskustelua ja haastateltava voi vastata haastattelukysymyksiin omin sanoin. Puolistrukturoitu haastattelu on strukturoidun ja avoimen haastattelun välimuoto. Se sisältää sekä strukturoituja haastattelukysymyksiä että avointa keskustelua. (Järvinen & Järvinen, 2011, 145.) Tässä tutkimuksessa tiedonkeruumenetelmänä käytetään puolistrukturoituihin haastatteluihin lukeutuvaa teemahaastattelua. Teemahaastattelussa tutkimus etenee yksityiskohtaisten haastattelukysymysten sijaan tutkimuksen keskeisten teemojen, eli aihepiirien varassa. Haastattelukysymysten tarkka muoto ja järjestys voivat puuttua, mutta teemat ovat kaikille haastateltaville samat. (Hirsjärvi & Hurme, 2011, 48.)

Tämän tutkimuksen yhteydessä toteutettavissa teemahaastatteluissa haastateltaville esitetään tutkimuksen teemoihin liittyviä dikotomisia haastattelukysymyksiä, joihin haastateltavat voivat vastata joko kyllä tai ei, sekä avoimia haastattelukysymyksiä, joihin haastateltavat voivat vastata omin sanoin. Haastattelukysymysten tarkoitus on ohjata teemoihin liittyvää keskustelua. Haastattelukysymysten muotoa ja järjestystä voidaan muuttaa haastattelutilanteen edellyttämällä tavalla ja haastateltavia kannustetaan haastattelukysymyksiin vastaamisen lisäksi tuomaan esille myös muita teemoihin liittyviä ajatuksia, käsityksiä, kokemuksia ja tunteita. Teemahaastatteluissa käytettävä haastattelurunko on tutkielman liitteenä (liite 1).

5.3 Aineiston analysointimenetelmä

Teemahaastatteluissa aineistoa kertyy usein runsaasti, vaikka haastateltavien lukumäärä olisi pieni. Tutkijalla on useita eri tapoja tallentaa aineisto. Hän voi esimerkiksi nauhoittaa tai videoida haastattelutilanteet. Kun aineisto on tallennettu, tutkija voi analysoida aineistoa suoraan tallenteista tai litteroida aineiston. Aineiston litterointi tarkoittaa aineiston kirjoittamista tekstiksi joko sanatarkasti tai valikoiden litteroinnin kohteeksi esimerkiksi haastateltavien puhetta tai tutkimuksen teemoihin liittyviä vastauksia. (Hirsjärvi & Hurme, 2011, 135-142.) Tämän tutkimuksen yhteydessä teemahaastattelut nauhoitetaan suoraan tieto-

koneelle mikrofonia apuna käyttäen ja aineistosta litteroidaan ainoastaan haastateltavien teemoihin ja haastattelukysymyksiin liittyvät vastaukset.

Aineiston analysoimisessa voidaan käyttää erilaisia lähestymistapoja. Yksi lähestymistapa on tulkita litteroitua aineistoa. Kyseinen lähestymistapa sisältää kolme eri vaihetta. Ensimmäisessä vaiheessa tutkija järjestää aineiston ja tuo esille sen rakentumisen. Toisessa vaiheessa tutkija selventää aineistoa, eli poistaa aineistosta asiaankuulumattomat osat. Kolmannessa vaiheessa tutkija tekee varsinaisen aineiston analysoinnin. (Hirsjärvi & Hurme, 2011, 137.) Tässä tutkimuksessa aineisto analysoidaan tulkitsemalla litteroitua aineistoa. Haastateltavien teemoihin ja haastattelukysymyksiin liittyvät vastaukset litteroidaan aluksi tallenteilta tietokoneelle, omiin tekstitiedostoihinsa, tekstinkäsittelyohjelmaa apuna käyttäen. Litteroinnin jälkeen aineistosta poistetaan ne haastateltavien vastaukset, jotka eivät ole tutkimuksen kannalta oleellisia. Viimeisenä tehdään varsinainen aineiston analysointi. Aineiston analysointimenetelminä käytetään teemoittelua ja aineiston yhteyksien tarkastelua.

Teemoittelu tarkoittaa aineiston luokittelua ja analysoimista tutkimuksen teemojen mukaan. Tutkija voi luokitella haastateltavien vastaukset joko teema-haastattelun lähtökohtana olleiden teemojen tai temahaastatteluissa esille tulleiden uusien teemojen alle. Teemoittelu mahdollistaa haastateltavien vastausten vertailun ja säännönmukaisuuksien etsimisen aineistosta. (Hirsjärvi & Hurme, 2011, 173-176.) Tässä tutkimuksessa aineisto teemoitellaan siten, että haastateltavien vastaukset tallennetaan litteroinnin jälkeen samaan tekstitiedostoon ja eri haastateltavien samaa teemaa ja haastattelukysymystä koskevat vastaukset siirretään peräkkäin. Tämä mahdollistaa vastausten vertailun keskenään sekä vastausten jakautumisen, yhteneväisyyksien, eroavaisuuksien ja säännönmukaisuuksien tarkastelun.

Aineiston yhteyksien tarkastelu tarkoittaa aineistossa esiintyvien ilmiöiden tarkastelua suhteessa toisiinsa. Aineiston yhteyksien tarkastelu on aineiston analysoinnin tärkein vaihe ja se voidaan toteuttaa eri tavoilla. Yksi tapa on tyyppittely, joka tarkoittaa sitä, että haastateltavat ryhmitellään tiettyjen yhteisten piirteidensä perusteella eri ryhmiin ja ryhmittelyjen avulla pyritään selittämään haastateltavien vastausten yhteneväisyyksiä ja eroavaisuuksia. Toinen tapa on poikkeavien tapausten analysointi, joka tarkoittaa sitä, että muista vastauksista poikkeaville vastauksille pyritään löytämään selitys. Poikkeavien tapausten analysointi pakottaa tarkastelemaan aineistoa uudesta näkökulmasta ja mahdollisesti myös kumoamaan aineiston pohjalta luotuja, tutkittavaa ilmiötä koskevia yleistyksiä. (Hirsjärvi & Hurme, 2011, 174-176.) Tässä tutkimuksessa aineiston yhteyksiä tarkastellaan sekä tyyppittelyn että poikkeavien tapausten analysoinnin avulla.

5.4 Tutkimuksen rajaukset

Tutkielman kirjallisuuskatsaus muodostaa käsityksen siitä, minkälaisia hyötyjä ja haasteita scrum-projekteihin voi liittyä kehitystiimin näkökulmasta. Tapaus-

tutkimuksen yhteydessä toteutettavien teemahaastattelujen teemat ja haastattelukysymykset muodostettiin tutkielman kirjallisuuskatsauksen aihepiirien ja tutkimustulosten pohjalta. Aivan kuten tutkielman kirjallisuuskatsauksessa, myös tapaustutkimuksessa pyrittiin tuomaan esille sellaisia scrumiin liittyviä hyötyjä ja haasteita, jotka voivat koskea eri scrum-projekteja ja kehitystiimejä muuttuvista tekijöistä huolimatta. Tapaustutkimuksessa ei tutkittukaan esimerkiksi tutkimuskohteeksi valitun scrum-projektin tapahtumiin ja tuotoksiin liittyviä hyötyjä ja haasteita, sillä niiden toteutustapa voi erota hyvin paljon eri scrum-projektien ja scrumtiimien välillä.

5.5 Tutkimuskohde

Tutkimuksen kohdeorganisaatioksi valittiin eräs IT-alan palveluyritys, joka tarjoaa konsultointipalveluja ja yksilöityjä liiketoimintatiedon hallinnan ratkaisuja asiakkailleen. Kohdeorganisaatiossa scrumia käytetään useiden asiakkaille toteutettavien projektien viitekehystenä. Kaikilla kohdeorganisaation scrum-projekteissa toimivilla scrummastereilla ja kehittäjillä on mahdollisuus osallistua Certified Scrum Master -koulutukseen sekä suorittaa koulutuksen päätyttyä Certified Scrum Master -sertifikaatti. Kyseinen koulutus antaa osallistujille valmiudet soveltaa scrumia menestyksekkäästi käytännössä.

Tapaustutkimuksen tarkoitus ei ole luoda yleistyksiä tai tilastollisia säännönmukaisuus, vaan tuottaa tutkittavaan ilmiöön liittyvää yksityiskohtaista ja täsmäntävää tietoa tapauksia tutkimalla (Eriksson & Koistinen, 2005, 43). Tässä tutkimuksessa tutkimuskohteeksi riittikin yksi scrum-projekti, jota tutkimalla saatiin yksityiskohtaista, kirjallisuuskatsauksen tutkimustuloksia täsmäntävää, scrumin hyötyihin ja haasteisiin kehitystiimin näkökulmasta liittyvää tietoa. Tutkimuskohteeksi valittiin yksi kohdeorganisaatiossa vuoden käynnissä ollut, asiakkaalle toteutettava scrum-projekti, jossa kehitettiin asiakasyrityksen intranetiä ja työtiloja. Asiakkaan puolelta scrum-projektiin osallistui asiakkaan projektipäällikkö, joka toimi tuoteomistajan roolissa ja listasi kehitettävän tuotteen toiminnallisuuksia koskevat vaatimukset tuotteen kehitysjonoon. Scrum-projektissa toimiva scrummaster puolestaan vastasi kehitettävän tuotteen toiminnallisuuksia koskevien vaatimusten priorisoinnista. Asiakkaan ja scrummasterin näkökulmasta tutkimuskohteeksi valittu scrum-projekti ei siis noudattanut täysin scrumin viitekehystä, sillä viitekehysten mukaan tuoteomistaja on vastuussa sekä tuotteen toiminnallisuuksia koskevien vaatimusten listaamisesta että priorisoinnista. Muilta osin tutkimuskohteeksi valittu scrum-projekti oli täysin scrumin viitekehysten mukainen ja siinä esiintyi kaikki scrumin viitekehyksessä kuvatut roolit, tapahtumat ja tuotokset.

5.6 Haastateltavien tausta

Yksi kvalitatiivisen tutkimuksen tunnuspiirre on, että tutkimuksen kohdejoukko valitaan tarkoituksenmukaisesti, ei siis otantamenetelmiä käyttäen. Tällöin puhutaan harkinnanvaraisesta näytteestä. (Hirsjärvi ym., 2007, 160.) Haastateltaviksi voidaan valita aiheesta parhaiten tietäviä henkilöitä, jolloin tutkimuksesta saadaan mahdollisimman kattava (Järvinen & Järvinen, 2011, 145). Tässä tutkimuksessa haastateltiin kaikkia tutkimuskohteena olevassa scrum-projektissa toimivia kehittäjiä sekä scrummasteria, joka osallistui scrum-projektiin yhdessä kehittäjien kanssa ja osasi näin vastata kehitystiimiä koskeviin haastattelukysymyksiin omasta näkökulmastaan.

Koska kvalitatiivisen tutkimuksen tarkoitus ei ole luoda yleistä tai tilastollisia säännönmukaisuuksia vaan ymmärtää tutkimuskohdetta, kohdejoukon koko ei ole rajattu. Aineisto voidaan muodostaa esimerkiksi yhden tai useamman yksilöhaastattelun pohjalta. Kvalitatiivisessa tutkimuksessa haastateltavien määrä on silloin riittävä, kun samat asiat alkavat toistua eri haastatteluisissa. (Hirsjärvi ym., 2007, 176-177.) Tässä tutkimuksessa haastateltavia oli yhteensä neljä, sillä scrummasterin lisäksi tutkimuskohteeksi valitussa scrum-projektissa toimi kolme kehittäjää. Jos haastateltavia olisi ollut enemmän, samaa scrum-projektia koskevissa vastauksissa olisi todennäköisesti ollut turhaa toistoa ja päällekkäisyyksiä.

Haastateltavien taustatiedot on koottu seuraavaan taulukkoon (taulukko 2). Haastateltavat on nimetty lyhenteitä käyttäen. Samaa nimeämiskäytäntöä käytetään myös tutkielman Tapaustutkimuksen tutkimustulokset -luvussa (luku 6), jossa esitetään ja analysoidaan haastateltavien antamia vastauksia tutkimuksen teemahaastatteluisissa heille esitettyihin haastattelukysymyksiin.

TAULUKKO 2 Haastateltavien tausta

	H1	H2	H3	H4
IT-alan työkokemus	Yli 10 vuotta	Noin 13 vuotta	Noin 8 vuotta	Noin 5 vuotta
Käytetyt ohjelmistokehitysmenetelmät	Vesiputousmalli, scrum	Vesiputousmalli, scrum	Scrum	Vesiputousmalli, scrum
Scrumiin liittyvä koulutus	Certified Scrum Master -koulutus ja sertifiikaatti	Certified Scrum Master -koulutus ja sertifiikaatti	Certified Scrum Master -koulutus ja Sertifiikaatti	Certified Scrum Master -koulutus ja sertifiikaatti
Scrumiin liittyvä käytännön kokemus	4 scrum-projektia, lisäksi muita scrumin viitekehystä mukailevia projekteja	2 scrum-projektia, lisäksi yksi scrumin viitekehystä mukaileva projekti	2 scrum-projektia, lisäksi kaksi scrumin viitekehystä mukailevaa projektia	1 scrum-projekti
Rooli	Scrummaster	Kehittäjä	Kehittäjä	Kehittäjä

5.7 Tutkimuksen kulku

Huhtikuussa 2013 otettiin sähköpostitse yhteyttä tutkimuksen kohdeorganisaatioissa käynnissä olevan scrum-projektin scrummasteriin ja kysyttiin, olivatko scrummaster ja scrum-projektissa toimivat kehittäjät kiinnostuneita osallistumaan scrumin hyötyjä ja haasteita kehitystiimin näkökulmasta käsittelevään tutkimukseen. Scrummasterille lähetettiin sähköpostitse tutkielman tutkimussuunnitelma ja saatekirje, jossa kerrottiin tarkemmin tutkimuksen tarkoituksesta, hyödyistä, luottamuksellisuudesta ja käytännön järjestelyistä, kuten haastattelujen mahdollisesta ajankohdasta, paikasta, arvioidusta kestosta ja käytetystä välineistöstä. Saatekirjeessä kerrottiin, että haastateltavat saivat pro gradu - tutkielman tutkimustuloksineen käyttöönsä sen valmistuttua ja pystyivät hyödyntämään tutkimustuloksia tutkimuskohteena olevan scrum-projektin tai muiden scrum-projektien tarkastelun, arvioimisen ja kehittämisen lähtökohtana. Saatekirjeessä korostettiin myös, että haastattelut olivat luottamuksellisia ja täysin anonyymeja, eikä haastateltavien henkilöllisyyttä, tutkimuskohteena olevaa scrum-projektia tai tutkimuksen kohdeorganisaatiota tuotu esille tutkielmassa. Sekä scrummaster että scrum-projektissa toimivat kehittäjät olivat kiinnostuneita osallistumaan tutkimukseen ja sähköpostikeskustelun pohjalta sovittiin sopivat ajankohdat ja paikat scrummasterin ja kehittäjien haastatteluille.

Haastattelujen teemat ja haastattelukysymykset muodostettiin aikaisemmin keväällä 2013 valmistuneen tutkielman kirjallisuuskatsauksen aihepiirien ja tutkimustulosten pohjalta. Ennen varsinaisia haastatteluja haastattelukysymyksiä muokattiin koehaastattelun perusteella. Koehaastattelu toteutettiin erään IT-alan ammattilaisen kanssa, joka tunsu työnsä puolesta scrumin viitekehyksen, mutta ei ollut itse osallistunut scrum-projekteihin.

Haastattelut toteutettiin huhtikuussa 2013 yksilöhaastatteluina haastateltavien omilla työpaikoilla, kohdeorganisaation eri toimipisteiden neuvotteluhuoneissa. Jokaisessa haastattelussa käytettiin samaa haastattelurunkoa (liite 1). Haastattelujen alussa haastateltaville kerrottiin tutkijan omasta taustasta, tutkimuksen tarkoituksesta, hyödyistä ja luottamuksellisuudesta sekä haastattelun etenemisestä. Ensimmäiset haastattelukysymykset koskivat haastateltavien taustaa ja tutkimuskohteena olevaa scrum-projektia. Haastateltavien taustaa koskevien haastattelukysymysten pohjalta pyrittiin etsimään selityksiä haastateltavien yhteneville ja eriäville vastauksille aineiston analysointivaiheessa. Loput haastattelukysymykset liittyivät tutkimuksen keskeisiin teemoihin. Haastateltaville esitettiin teemoihin liittyviä dikotomisia ja avoimia haastattelukysymyksiä. Jos haastateltavat vastasivat dikotomisiin kysymyksiin ainoastaan kyllä tai ei, heitä kannustettiin perustelemaan vastauksiaan tarkemmin. Haastattelukysymyksiin vastaamisen lisäksi haastateltavia kannustettiin tuomaan esille myös muita teemoihin liittyviä hyötyjä ja haasteita, joita he olivat huomanneet.

Jokainen haastattelu kesti noin tunnin ja haastatteluja tehtiin yhteensä neljä. Haastattelut nauhoitettiin suoraan tietokoneelle mikrofonina apuna käyttäen ja haastatteluaineistosta litteroitiin haastateltavien teemoihin ja haastatteluky-

symyksiin liittyvät vastaukset. Haastateltavilta kysyttiin haastattelujen loppuksi, haluavatko he tarkastaa vastauksensa litteroinnin jälkeen, mutta jokainen haastateltava kertoi, että oli kiireinen eikä ehtinyt perehtyä litteroituun aineistoon. Haastateltavat käyttivät scrumin tapahtumista ja tuotoksista puhuessaan englanninkielisiä käsitteitä, jotka muutettiin litteroinnin yhteydessä vastaamaan Eskelisen ym. (2002) laatimaa suomenkielistä scrum-sanastoa. Muuten haastateltavien vastauksia ei muokattu, vaan ne litteroitiin suorina lainauksina haastatteluaineistosta. Litteroinnin jälkeen aineisto analysointiin käyttäen aineiston analysointimenetelminä teemoittelua ja aineiston yhteyksien tarkastelua. Tutkielman valmistuttua valmis tutkielma lähetettiin kiitoksena haastateltaville.

5.8 Tutkimuksen luotettavuus

Tutkimuksen luotettavuutta voidaan arvioida tutkimuksen reliabiliteetin ja validiteetin pohjalta. *Reliabiliteetti* tarkoittaa tutkimuksen toistettavuutta. (Hirsjärvi & Hurme, 2011, 185-186.) Tämän tutkimuksen toistettavuutta tukee se, että tutkimuksen toteutus on kuvattu alusta loppuun tutkielman Tapaustutkimus -luvussa (luku 5) ja tutkimuksessa käytetty haastattelurunko on esitetty tutkielman liitteenä (liite 1). Se, että haastateltavien vastauksissa esiintyi hyvin paljon yhteneväisyyksiä viittaa myös tutkimuksen toistettavuuteen. Tutkimuksen reliabiliteetti oli siis hyvällä tasolla.

Tutkimuksen validiteettia voidaan arvioida sisäisen ja ulkoisen validiteetin näkökulmasta. *Sisäinen validiteetti* tarkoittaa, että tutkimuksessa tutkitaan sitä, mitä on tarkoitus tutkia. (Metsämuuronen, 2006.) Tämän tutkimuksen tarkoitus oli tutkia, mitä hyötyjä ja haasteita erään IT-alan palveluyrityksen yhteen, käynnissä olevaan scrum-projektiin liittyy kehitystiimin näkökulmasta. Tutkimus perustui ketteriä menetelmiä ja scrumia käsittelevään kirjallisuuteen. Tutkimuksen yhteydessä toteutettujen teemahaastattelujen teemat ja haastattelukysymykset muodostettiin tutkielman kirjallisuuskatsauksen aihepiirien ja tutkimustulosten pohjalta ja tutkimuksessa tutkittiin kaikkiin tutkielman kirjallisuuskatsauksessa esille tulleisiin aihepiireihin liittyviä hyötyjä ja haasteita kehitystiimin näkökulmasta. Sisäinen validiteetti siis saavutettiin tutkimuksessa. *Ulkoinen validiteetti* tarkoittaa tutkimustulosten yleistettävyyttä (Metsämuuronen, 2006). Koska tämän tutkimuksen tutkimuskohteena oli vain yksi scrum-projekti ja haastateltavien lukumäärä oli pieni, tutkimuksen tutkimustuloksia ei voida yleistää. Tutkimuksen tarkoitus ei kuitenkaan ollut luoda scrumin hyötyjä ja haasteita kehitystiimin näkökulmasta koskevia yleistyksiä, vaan tuottaa yksityiskohtaista, kirjallisuuskatsauksen tutkimustuloksia täsmentävää tietoa tutkimuskohteeksi valittua scrum-projektia tutkimalla. Ulkoista validiteettia ei siis saavutettu tutkimuksessa.

5.9 Yhteenveto

Tämän luvun tarkoitus oli kuvailla tutkielman empiiristä osuutta ja tapaustutkimuksen toteuttamista alusta loppuun kertomalla tarkemmin valitusta tutkimusmenetelmästä, tiedonkeruumenetelmästä, aineiston analysointimenetelmästä, tutkimuksen rajauksista, tutkimuskohteesta, haastateltavien taustasta sekä tutkimuksen kulusta ja luotettavuudesta. Tapaustutkimuksen toteuttamisen kuvaaminen alusta loppuun auttaa ymmärtämään, miten tutkija on päätenyt tapaustutkimuksen tutkimustuloksiin ja mahdollistaa tutkimuksen toistamisen.

6 TAPAUSTUTKIMUKSEN TUTKIMUSTULOKSET

Tässä luvussa esitetään ja analysoidaan tutkielman empiirisen osuuden, eli kohdeorganisaatiossa teemahaastattelujen avulla toteutetun tapaustutkimuksen tutkimustulokset. Tapaustutkimuksen tarkoitus oli tuottaa yksityiskohtaista, kirjallisuuskatsauksen tutkimustuloksia täsmentävää, scrumin hyötyihin ja haasteisiin kehitystiimin näkökulmasta liittyvää tietoa tutkimuskohteeksi valittua scrum-projektia tutkimalla. Tapaustutkimuksen yhteydessä toteutettavien teemahaastattelujen teemat ja haastattelukysymykset muodostettiin kirjallisuuskatsauksen aihepiirien ja tutkimustulosten pohjalta. Tapaustutkimuksen tutkimustulokset esitetään teemoittain, samassa järjestyksessä kirjallisuuskatsauksen aihepiirien ja tutkimustulosten kanssa ja tiivistetään luvun yhteenvedossa samanlaiseen taulukkoon, kuin kirjallisuuskatsauksen tutkimustulokset. Tämä mahdollistaa kirjallisuuskatsauksen ja tapaustutkimuksen tutkimustulosten vertailun toisiinsa.

6.1 Scrumin käyttöönotto

Scrum on ohjelmistokehityksen viitekehys, joka on kehitetty helpottamaan ohjelmistokehitysprojektien hallintaa ja toteuttamista. Se kuvaa pienen joukon rooleja, tapahtumia ja tuotoksia, jotka scrum-projekteissa täytyy esiintyä. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Ensimmäinen teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *scrumin käyttöönotto*. Scrumin käyttöönotto oli ainoa teema, johon liittyvä tutkimus ei kohdistunut vain tutkimuskohteena olevaan scrum-projektiin, vaan haastateltavia pyydettiin haastattelukysymyksiin vastatessaan muistelemaan myös aikaisempia scrum-projekteja, joihin he olivat osallistuneet.

Ensimmäinen teemaan liittyvä haastattelukysymys oli *onko scrumin viitekehys yksinkertainen ja helposti opittava?* H1, H2 ja H3 olivat sitä mieltä, että scrumin viitekehys oli yksinkertainen ja helposti opittava. "Kehittäjän näkökulmasta kyllä" (H2). H4 oli kuitenkin eri mieltä muiden haastateltavien kanssa. "Mun

mielestä se on kaikkea muuta kuin yksinkertainen, pääpiirteet mitä scrumissa on toteutuu hyvin, mutta sitten ne hienot nyanssit on asia erikseen” (H4). Kuten muilla haastateltavilla, myös H4:llä oli sekä koulutuksen että käytännön tuomaa kokemusta scrumista. Haastateltavien eriävien vastausten pohjalta voidaan todeta, että oli yksilöllistä, kuinka yksinkertaisena ja helposti opittavana kehittäjät pitivät scrumin viitekehystä ja oli mahdollista, että kehittäjien mielestä scrumin viitekehys oli jopa monimutkainen ja vaikeasti opittava.

Kun haastateltavilta kysyttiin *onko scrumin käyttöönotto työlästä, aikaavievää tai haastavaa*, heidän vastauksensa jakaantuivat huomattavasti. H2 ja H3 olivat sitä mieltä, ettei scrumin käyttöönotto ollut työlästä, aikaavievää tai haastavaa. ”Ei sen työläämpää kuin minkään muunkaan prosessin käyttöönotto” (H3). H1 kertoi scrumin käyttöönoton vievän aikaa. ”Kyllä se aikaa vie, ja varsinkin sen täysmittainen hyödyntäminen ja tiettyjen asioiden käytännön toteutus, joita ei välttämättä kuvata scrumin viitekehyksessä” (H1). H4 puolestaan oli sitä mieltä, että scrumin käyttöönotto oli haastavaa. ”Haastavaa se kyllä on” (H4). Haastateltavien eriävien vastausten pohjalta voidaan olettaa, että oli mahdollista, että scrumin käyttöönotto oli työlästä, aikaavievää tai haastavaa. H1, H3 ja H4 korostivat, että scrum-projekteihin osallistuvien kehittäjien ja scrummasteiden kannatti ehdottomasti osallistua scrumia käsittelevään koulutukseen ennen scrumin käyttöönottoa.

Ehdottomasti kannattaa käydä koulutuksessa. Tietenkin jos toimii asiakkaan roolissa, riittää ehkä että toimittaja käy läpi scrumin viitekehysten ja sen, miten heidän kanssa toimitaan. Mutta kehittäjien ja scrummasteiden pitäisi sitten tuntea koko prosessi. (H4.)

Haastateltavilta kysyttiin myös *voivatko kehittäjät vastustaa scrumin käyttöönottoa?* Kukaan haastateltavista ei ollut kohdannut tilannetta, jossa kehittäjät olisivat vastustaneet scrumin käyttöönottoa. ”Ei mun mielestä, yleensä kehittäjät tykkäävät scrumista” (H2). ”Ei ole tullut vastaan ketään, kuka vastustaisi sitä” (H4). H1 ja H3 kuitenkin huomauttivat, että oli mahdollista, että yksittäiset kehittäjät vastustivat scrumin käyttöönottoa. ”Ei ole tullut vastaan, en usko että yleisellä tasolla olisi mitään laajaa vastustusta, mutta kyllähän se yksilötasolla on mahdollista” (H1). ”Niiden henkilöiden, jotka eivät ole tottuneet scrumiin, voi olla haastavaa lähteä mukaan, mutta se ei välttämättä ole ihan scrum mitä ne vastustaa, vaan sen määrättyjä piirteitä” (H3).

Neljäs teemaan liittyvä haastattelukysymys oli *voiko scrumtiimi muuttaa scrumin viitekehystä tai käyttää viitekehystä vain osittain?* Jokainen haastateltava kertoi, että oli mahdollista, että scrumtiimi muutti scrumin viitekehystä tai käytti vain osaa viitekehysten rooleista, tapahtumista ja tuotoksista. ”Voi ja sitä myös tapahtuu” (H4). H1, H2 ja H3 olivat myös aikaisemmin osallistuneet scrum-projekteihin, joissa näin oli tehty. H1 kuitenkin huomautti, ettei scrumin viitekehystä mukailevia ohjelmistokehitysprojekteja yleensä kutsuttu scrum-projekteiksi. ”Se on varmaan hyvin yleistä, mutta se on sitten eri asia sanoaanko niitä scrum-projekteiksi” (H1). Haastateltavilta ei kysytty, kuinka scrumin viitekehystä mukailevat ohjelmistokehitysprojektit olivat onnistuneet. H3 ja H4 kuitenkin korostivat, että oli tärkeää, että scrumtiimi ja scrum-projektit noudat-

tivat scrumin viitekehystä. ”Kyllä se sitten alkaisi vähän hiertämään, jos joku kehittäjä ei noudattaisi viitekehystä tai sovittua prosessia” (H4). ”Voi, mutta kyllä se pitäisi ottaa pääpiirteittäin käyttöön, vähintään kaikki roolit ja vaiheet, jotta se toimisi ja siitä tulisi tehokas ja hyvä” (H3). H3:n ja H4:n vastausten pohjalta voidaankin olettaa, että oli mahdollista, että scrumin hyödyt jäivät saavuttamatta, jos scrumtiimi muutti scrumin viitekehystä tai käytti vain osaa viitekehysten rooleista, tapahtumista ja tuotoksista. Voidaan myös olettaa, että oikein noudatettuna scrum toi useita hyötyjä ohjelmistokehitysprojekteihin, koska haastateltavat korostivat, kuinka tärkeää scrumin viitekehysten noudattaminen oli.

Viimeinen teemaan liittyvä haastattelukysymys oli *pitävätkö kehittäjät työskentelystä scrum-projekteissa?* Jokainen haastateltava oli ehdottomasti sitä mieltä, että kehittäjät pitivät työskentelystä scrum-projekteissa. ”Yleisesti ottaen kehittäjät pitävät siitä toimintatavasta” (H1). ”Olen tykännyt scrumista enemmän kuin perinteisistä menetelmistä” (H2). ”Olen tykännyt scrumista ja voisin ihan kokoajan työskennellä scrumin mukaan, ei ole kertaakaan harmittanut tulla töihin tämän projektin aikana” (H4). H3 ja H4 toivat esille, että he käyttivät scrumia myös muissa projekteissaan.

Olen tykännyt scrumista enkä vaihtaisi sitä muihin ohjelmistokehitysmenetelmiin. Ja on hyvä, että on tällainen yhteinen viitekehys, jota ei ole kehittänyt joku talon sisäinen johtaja, vaan se on tullut ulkopuolelta ja ihmiset ovat itse sitoutuneet siihen. Meillä ei ole oikeastaan ollut mitään selkeää prosessia, jota olisi noudatettu, niin scrum määrittelee pähkinänkuoressa, miten ohjelmistokehitysprojektit tulisi toteuttaa. Ja pienemmätkin projektit toteutan ihan samalla tavalla, koska niissä on ihan samat elementit. Scrum selkeyttää asioita huomattavasti. (H3.)

6.2 Kehitystiimin koko

Toinen teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin koko*. Kehitystiimiin kuuluu tyypillisesti 3-9 kehittäjää (Schwaber & Sutherland, 2011; Sutherland, 2010). Tutkimuskohteena olevassa scrum-projektissa kehitystiimiin kuului kolme kehittäjää. Ensimmäinen teemaan liittyvä haastattelukysymys oli *onko kehitystiimi sopivan kokoinen?* Jokaisen haastateltavan mielestä kehitystiimi oli sopivan kokoinen ja sai aikaan vaaditun määrän työtä. ”Meitä on kolme kehittäjää tällä hetkellä ja se on hyvä määrä tähän projektiin nähden” (H2). ”Projektin kokoon ja kokonaisuusmäärän tarpeeseen nähden kyllä” (H1). H3 ja H4 toivat esille, että olisi ollut myös mielenkiintoista nähdä, miten scrum-projektit, joissa olisi enemmän kehittäjiä, onnistuisivat. ”Olisi jännä kokeilla scrumia myös isommalla kehitystiimillä, koska silloin tarvittaisiin niitä sääntöjä enemmän” (H3). Haastateltavien vastausten pohjalta voidaan olettaa, että scrum soveltui hyvin pienen kehitystiimin käyttöön, mutta haastateltavat eivät olleet varmoja, olisiko scrum-projekti onnistunut, jos kehittäjiä olisi ollut enemmän. Haastateltavilta kysyttiin myös, *kuinka usein kehittäjät ovat vuorovaikutuksessa toistensa kanssa?* Jokainen

haastateltava kertoi, että kehittäjät keskustelivat toistensa kanssa jatkuvasti. ”Päivittäin tulee puhuttua” (H2). Tämä viittasi siihen, että kehitystiimin pieni koko mahdollisti suoran ja jatkuvan vuorovaikutuksen sekä nopean tiedon jakamisen kaikkien kehittäjien välillä.

Kolmas teemaan liittyvä haastattelukysymys oli *onko yrityksellä tarpeeksi resursseja muodostaa optimaalisen kokoisia kehitystiimejä?* H4 oli sitä mieltä, että kohdeorganisaatiolla oli tarpeeksi resursseja muodostaa optimaalisen kokoisia kehitystiimejä. H1, H2 ja H3 olivat kuitenkin eri mieltä H4:n kanssa. H1 kertoi haasteena olevan, että samoja resursseja jouduttiin osittain lainaamaan joko vanhoihin projekteihin liittyviin ylläpitotehtäviin tai muihin projekteihin. H2 toi esille, että työtä oli enemmän kuin tekijöitä. ”Tällä hetkellä on se tilanne, että kysyntää on enemmän kuin tekijöitä” (H2). Haastateltavien vastausten pohjalta voidaan todeta, että haasteena ei ollut varsinaisesti kehitystiimin koko, vaan kehittäjien muut samanaikaiset projektit, mikä saattoi selittää haastateltavien eriävät vastaukset. Haastattelut kuitenkin osoittivat, ettei kohdeorganisaatiolla ollut aina tarpeeksi resursseja muodostaa optimaalisen kokoisia kehitystiimejä.

Neljäs teemaan liittyvä haastattelukysymys oli *toimivatko kehittäjät samaan aikaan useissa eri projekteissa?* Jokainen haastateltava kertoi, että kehittäjät olivat toimineet scrum-projektin aikana myös muissa projekteissa. ”Kyllä, ainakin tällä hetkellä on päällekkäisiä hommia ja joutuu välillä hyppimään eri projektien välillä” (H4). H3 kuitenkin huomautti, että koska kyseessä oli IT-alan palveluyritys, joka toteutti projekteja asiakkailleen, resurssipula voitiin ottaa huomioon sovittaessa asiakkaan kanssa projektin aikataulusta.

Kyllä ja se on vähän ongelmallistakin. Mutta pitää muistaa, että me ei olla tuotekehitysyritys, vaan me toteutetaan projekteja ja asiakas ostaa projektin tietyn mittaisena. Jos projekti pitäisi toteuttaa kahdessa kuukaudessa, sitten tarvittaisiin isompi kehitystiimi ja enemmän osallistumista sekä meidän että myös asiakkaan puolelta. (H3.)

H2, H3 ja H4 pitivät muista projekteista aiheutuvia töitä valitettavana, sillä ne vaikeuttivat scrum-projektiin osallistumista. H4 kertoi, että kehittäjät toteuttivat scrum-projektiin liittyviä työtehtäviä välillä yksin muiden kehittäjien osallistuksessa muihin projekteihin. ”Muista töistä johtuen saa välillä tehdä töitä yksin” (H4). H3 kuitenkin huomautti, että kehittäjät pyrkivät järjestämään aikataulunsa niin, että he pystyivät tekemään scrum-projektiin liittyvää työtä yhtä aikaa. ”Mutta jos me tiedetään että joudumme hyppimään eri projekteissa kesken sprintin, me pyritään kuitenkin tekemään työtä samaan aikaan, esimerkiksi sprintin ensimmäisen tai toisen viikon ajan, jolloin informaatio kulkee paljon paremmin” (H3). H3 ja H4 huomauttivat myös, että vaikka kehittäjät osallistuiivat muihin projekteihin, he osallistuivat päiväpalavereihin ja seurasivat sprintin tehtävälistaa, jolloin he pysyivät ajantasalla muiden kehittäjien tekemästä työstä ja sprintin etenemisestä ja pystyivät kommentoimaan tarvittaessa.

6.3 Kehitystiimin työskentely-ympäristö

Kolmas teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin työskentely-ympäristö*. Kehitystiimejä kannustetaan työskentelemään avoimessa työtilassa, jossa kehittäjät eivät ole eristyksissä toisistaan, vaan he näkevät ja kuulevat toisensa kokoajan (Cho, 2008). Ensimmäinen teemaan liittyvä haastattelukysymys oli *työskentelevätkö kehittäjät samalla työpaikalla?* Haastatteluissa tuli esille, että tutkimuskohteenä olevassa scrum-projektissa kaikki kehittäjät eivät työskennelleet samalla työpaikalla, vaan scrummaster ja yksi kehittäjä työskentelivät eri paikkakunnalla sijaitsevalla työpaikalla kuin kaksi muuta kehittäjää. Kehittäjät olivat kuitenkin aikaisemmin työskennelleet samalla työpaikalla ja tunsivat toisensa entuudestaan. Kehittäjät käyttivät päivittäin yrityskäyttöön suunniteltua pikaviestinsovellusta, jonka kautta he pystyivät muun muassa keskustelemaan toistensa kanssa ja jakamaan tietokoneensa ruudun toisillensa. ”Keskustelu tapahtuu (Microsoft) Lyncin kautta ja ääni ja kuvaruutu välitetään muille kehittäjille” (H2). Jokainen haastateltava toi esille, että olisi ollut parempi, jos kehittäjät olisivat työskennelleet samalla työpaikalla, mutta korostivat, että käytetty pikaviestinsovellus mahdollisti vuorovaikutuksen ja yhteistyön eri paikkaan työskentelevien kehittäjien välillä. ”Tottakai olisi vielä parempi jos olisimme kaikki samassa tilassa, mutta se ei ole mahdollista” (H1). Haastattelujen pohjalta voidaan todeta, ettei kehittäjien ollut aina mahdollista työskennellä samalla työpaikalla tai samassa työtilassa, mutta viestintäteknologian avulla voitiin tukea eri paikkaan työskentelevien kehittäjien välistä vuorovaikutusta ja yhteistyötä.

(Microsoft) Lynciä käytetään. Se on melkein kuin vieressä olisi kun voi jakaa muille oman työpöytänsä. Saa nähdä miten scrumin määritykset muuttuvat. Onhan se tietenkin hyvä olla samassa tilassa, koska silloin sä tutustut toiseen ihmiseen ihan eri tavalla, mutta nykyään tekniikka on niin erilaisia kuin silloin kun scrum on kehitetty, niin se ei enää niinkään haittaa. Mutta olisihan se tietenkin vielä parempi jos me oltaisiin kaikki samalla työpaikalla ja samassa tilassa. (H3.)

Toinen teemaan liittyvä haastattelukysymys oli *työskentelevätkö kehittäjät avoimessa työtilassa?* Haastatteluissa tuli esille, että scrummaster ja yksi kehittäjä sekä toisella työpaikalla työskentelevät kehittäjät työskentelivät toistensa kanssa avoimissa työtiloissa, joissa he näkivät ja kuulivat toisensa kokoajan. ”Kyllä ja haaveena olisi että saisimme työtilasta vielä vähän avonaisemman, nyt siinä on pieni sermi meidän välissä” (H2). ”Kyllä, meillä on vain pöytä välissä. Ja me pyritään siihen, että ne jotka ovat samassa projektissa työskentelevät samassa huoneessa.” (H3.) Haastateltavien vastausten pohjalta voidaan todeta, että tutkimuskohteenä olevassa scrum-projektissa toimivat kehittäjät pyrkivät työskentelemään avoimessa työtilassa.

Haastateltavilta kysyttiin myös *tukeeko työskentely avoimessa työtilassa kehittäjien välistä vuorovaikutusta ja yhteistyötä?* Jokainen haastateltava oli ehdottomasti sitä mieltä, että työskentely avoimessa työtilassa tuki kehittäjien välistä

vuorovaikutusta ja yhteistyötä. ”Tukee ja siinä tulee tutuksi toisen kanssa ja mitä enemmän sä tutustut toiseen, sitä paremmin se vuorovaikutus ja yhteistyö sujuu” (H3). Viimeinen teemaan liittyvä haastattelukysymys oli *vaikeuttavatko avoimeen työtilaan liittyvät häiriötekijät kehittäjien työntekoa?* Kukaan haastateltavista ei ollut huomannut avoimeen työtilaan liittyvien häiriötekijöiden, kuten ympärillä käytävän keskustelun, vaikeuttavan kehittäjien työntekoa. ”Ei meitä häiritse, koska meillä on sen verran suljettu työhuone ja me itse pidetään kauhean kovaa meteliä ja naureskellaan” (H3). ”Ei oikeastaan, se on sen verran hiljainen tila” (H4). H1 kuitenkin huomautti, että oli yksilöllistä, häiritsivätkö avoimeen työtilaan liittyvät häiriötekijät kehittäjiä. ”Se riippuu henkilöstä, mutta meitä ei ainakaan omasta mielestäni ole häirinnyt” (H1). H1:n vastauksen pohjalta voidaan todeta, että avoimeen työtilaan liittyvät häiriötekijät saattoivat vaikeuttaa kehittäjien työntekoa, vaikka tutkimuskohteena olevassa scrum-projektissa näin ei ollut.

6.4 Kehitystiimin monitaitoisuus

Neljäs teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin monitaitoisuus*. Kehitystiimin monitaitoisuus tarkoittaa sitä, että kehitystiimi muodostuu pienestä joukosta ammatillaisia, joilla on tarvittava osaaminen toteuttaa scrum-projekti ilman ulkopuolista apua (Schwaber & Sutherland, 2011; Sutherland, 2010). Ensimmäinen teemaan liittyvä haastattelukysymys olikin *onko kehitystiimillä tarvittava osaaminen toteuttaa scrum-projekti ilman ulkopuolista apua?* Jokaisen haastateltavan mielestä kehitystiimin monitaitoisuus toteutui tutkimuskohteena olevassa scrum-projektissa. ”Kyllä, ainakaan tähän mennessä ei ole tarvinnut pyytää ulkopuolista apua” (H2).

Toinen teemaan liittyvä haastattelukysymys oli *onko kehittäjillä selkeät roolit vai edellytetäänkö kehittäjiltä laajaa osaamista?* Jokainen haastateltava toi esille, ettei kehittäjillä ollut selkeitä rooleja, vaan heillä oli tarvittava osaaminen osallistua kaikkien scrum-projektiin liittyvien työtehtävien toteuttamiseen. ”Toteutus työtä ja testausta tekevät kaikki” (H1). ”Työtehtävät eivät ole henkilöityneet, vaan kaikki pystyvät tekemään samoja työtehtäviä” (H4). Kehitystiimin pieni koko ja kehittäjien muut samanaikaiset projektit olivat todennäköisiä syitä siihen, miksi kehittäjiltä edellytettiin laajaa osaamista. Haastateltavat kuitenkin huomauttivat, että oli myös muutamia työtehtäviä, jotka olivat henkilöityneet tietyn kehittäjän toteutettaviksi, sillä ne olivat luonteeltaan sellaisia, ettei muiden kehittäjien kannattanut opetella niitä.

Kyllä me pääasiassa pyritään siihen, että olisi laaja osaaminen, mutta tässäkin projektissa osa työtehtävistä henkilöityy väkisin kun on sellaisia työtehtäviä, joita muiden kehittäjien ei ole järkevää alkaa opettelemaan. Mutta me jaetaan osaamista niistä tekniikoista, joita me tarvitaan enemmän. (H3.)

Kolmas teemaan liittyvä haastattelukysymys oli *toteuttavatko kehittäjät vain oman osaamisalueensa mukaisia työtehtäviä?* Jokainen haastateltava kertoi, etteivät kehittäjät toteuttaneet ainoastaan oman osaamisalueensa mukaisia työtehtäviä, vaan he pyrkivät hankkimaan laajan, scrum-projektin edellyttämän osaamisen joko opettelemalla itse tai kysymällä neuvoa muilta kehittäjiltä. ”Me osallistutaan kaikkeen ja opetellaan paljon myös itse” (H3). ”Kyllähän niitäkin tilanteita tulee, että tulee kysyttyä muilta kehittäjiltä, mitenäs tämä (työtehtävä) nyt tehdään” (H4). Haastateltavien vastausten pohjalta voidaankin todeta, että kehittäjien oli mahdollista toteuttaa sekä työtehtäviä, joiden toteuttamiseen heillä oli tarvittava osaaminen että työtehtäviä, joiden toteuttamisen he halusivat oppia. Haastateltavilta kysyttiin myös *neuvovatko kokeneemmat kehittäjät kokemattomampia kehittäjiä?* Haastateltavat olivat yksimielisiä siitä, että kehittäjät jakoivat mielellään työtehtäviin liittyvää tietoa toisillensa ja suhtautuivat positiivisesti, jos muut kehittäjät kysyivät neuvoa. ”Kyllä ja siihen suhtaudutaan hyvin positiivisesti” (H2).

Kyllä. Se on nimenomaan tapa siirtää osaamista. Ja kun kaikilla on halu kehittyä ja oppia uutta, siihen suhtaudutaan todella hyvin. Harvemmin kokeneemmat kehittäjät tulevat neuvomaan ketään, vaan kehittäjät pyytävät itse tarvittaessa neuvoa. Tässä projektissa se on ainakin 100% positiivista. (H3.)

Seuraavat teemaan liittyvät haastattelukysymykset olivat *tukeeko työskentely monitaitoisessa kehitystiimissä kehittäjien välistä tiedon jakamista ja parantaako työskentely monitaitoisessa kehitystiimissä kehittäjien osaamista?* Jokainen haastateltava oli ehdottomasti sitä mieltä, että työskentely monitaitoisessa kehitystiimissä tuki kehittäjien välistä tiedon jakamista ja paransi kehittäjien osaamista. ”Kyllä, me opitaan kokoajan toisiltamme ja jos joku tulee mukaan projektiin me myös perehdytetään ja opastetaan häntä” (H4). Haastatteluissa tuli myös esille, että osaamisen kehittyessä scrum-projektin eteneminen ei ollut kiinni ainoastaan yhden kehittäjän työpanoksesta, vaan kehittäjät pystyivät osallistumaan kaikkien scrum-projektiin liittyvien työtehtävien toteuttamiseen ja jatkamaan toistensa työtehtäviä esimerkiksi kiireiden tai poissaolojen yhteydessä. ”Sitten jos itse juuttuu johonkin kohtaan, joku muu ottaakin sen seuraavan työtehtävän hoidakseen” (H2).

Viimeinen teemaan liittyvä haastattelukysymys oli *onko scrum-projektiin helppo löytää kehittäjiä, joilla on tarvittava osaaminen?* H1, H3 ja H4 olivat sitä mieltä, että tutkimuksen kohdeorganisaatiossa scrum-projekteihin oli helppo löytää kehittäjiä, joilla oli projektin toteuttamisen kannalta tarvittava osaaminen. ”Kyllä tuolta yläkerrasta voisi ottaa kenet tahansa tähän projektiin” (H4). H3 huomautti, että scrum-projekteihin voitiin valita myös kokemattomampia kehittäjiä, sillä kehittäjien oli mahdollista kehittää osaamistaan scrum-projektin aikana. ”Uskoisin niin ja jos tarpeeksi rahaa tulee sisään, kehittäjät voivat tulla vaikka toiselta paikkakunnalta. Ja meillä on myös aika paljon sitä, että me opetellaan itse.” (H3.) H2 toi kuitenkin esille, että scrum-projekteihin saattoi olla ylipäättään vaikea löytää kehittäjiä. H2:n eriävän vastauksen saattoi selittää se, että scrumin viitekehyksen mukaisia ohjelmistokehitysprojekteja oli toteutettu

tutkimuksen kohdeorganisaatiossa kohtuullisen vähän, eikä kaikilla kehittäjillä välttämättä ollut vielä koulutuksen tai käytännön tuomaa kokemusta scrumista.

6.5 Kehitystiimin itseohjautuvuus

Viides teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin itseohjautuvuus*. Kehitystiimin itseohjautuvuus tarkoittaa sitä, ettei kukaan johda kehitystiimin toimintaa, vaan kehitystiimi päättää itse työnjaosta ja työskentelytavoistaan (Schwaber & Sutherland, 2011; Sutherland, 2010). Ensimmäinen teemaan liittyvä haastattelukysymys olikin *päättääkö kehitystiimi itse työnjaosta ja työskentelytavoistaan?* Jokaisen haastateltavan mielestä kehitystiimin itseohjautuvuus toteutui tutkimuskohteenä olevassa scrum-projektissa. ”Kyllä, meillä on täysi päätösvalta” (H3). ”Kyllä, muilla ei ole sananvaltaa siihen, miten me työt tehdään” (H4). Haastateltavilta kysyttiin myös *antaako scrummaster ja muut scrum-projektin sidosryhmät työnjakoon ja työskentelytapoihin liittyvän päätöksentekovallan kehitystiimille?* Haastateltavat olivat yksimielisiä siitä, että scrummaster ja muut scrum-projektin sidosryhmät antoivat työnjakoon ja työskentelytapoihin liittyvän päätöksentekovallan kehitystiimille. ”Se on luontevaa että he vastaavat toteutuksesta, mutta asiakkaan ja tuoteomistajan priorisointia tietenkin tuodaan esille, eli missä järjestyksessä asioita pitäisi tehdä” (H1). ”Kyllä antaa, se on ollut ihan luontevaa, priorisointi sieltä tietenkin tulee” (H2).

Kolmas teemaan liittyvä haastattelukysymys oli *lisääkö työskentely itseohjautuvassa kehitystiimissä kehittäjien mahdollisuuksia vaikuttaa omiin työtehtäviinsä?* Jokaisen haastateltavan mielestä työskentely itseohjautuvassa kehitystiimissä lisäsi kehittäjien mahdollisuuksia vaikuttaa omiin työtehtäviinsä. ”Scrummaster tai muut henkilöt eivät osallistu siihen, kun me pilkootaan tuotteen kehitysjonon kohdat työtehtäviksi ja tehdään niille työmääräarviot, eli siinä kehittäjillä on vapaat kädet” (H4). Haastateltavilta kysyttiin myös *parantaako työskentely itseohjautuvassa kehitystiimissä kehitystiimin toiminnan koordinoointia?* Jokainen haastateltava oli huomannut, että työskentely itseohjautuvassa kehitystiimissä paransi kehitystiimin toiminnan koordinoointia. ”Kyllä, erityisesti tehtävien hallinta on ehkä perinteisiin malleihin verrattuna heille selkeämpää, koska on vain yksi priorisoitu lista, josta kehittäjät voivat katsoa aina seuraavan työtehtävän sitten kun edellinen valmistuu” (H1). H1 huomautti, että toiminnan koordinoinnin parantuminen johtui osittain myös siitä, että kehitystiimi kävi työtilanteen päivittäin läpi päiväpalaverissa. ”Ja se että kerran päivässä käydään tilanne läpi ohjaa sitä automaattisesti” (H1). H1 ja H3 toivat myös esille, että kehitystiimi tarkasteli ja kehitti toimintaansa säännöllisesti, koko scrum-projektin ajan. ”Lyyhyitä sprintin retrospektiiveja on myös pidetty ja aina on löytynyt ongelmakohtia, joita halutaan parantaa” (H1). ”Meidän projekti kehittyy jatkuvasti, kun pidetään sprintin retrospektiiveja ja kehitetään jatkuvasti meidän omaa toimintaa” (H3). Haastateltavien vastausten pohjalta voidaankin todeta, että useat scrum-

projektin tapahtumat ja tuotokset tukivat itseohjautuvan kehitystiimin toimintaa.

Viimeinen teemaan liittyvä haastattelukysymys oli *parantaako työskentely itseohjautuvassa kehitystiimissä kehittäjien motivaatiota, scrum-projektiin sitoutumista, tehokkuutta tai tuottavuutta?* H2, H3 ja H4 olivat ehdottomasti sitä mieltä, että työskentely itseohjautuvassa kehitystiimissä paransi kehittäjien motivaatiota, scrum-projektiin sitoutumista, tehokkuutta ja tuottavuutta. ”Olen huomannut oikeastaan kaikkiin näihin positiivisia vaikutuksia” (H2).

Kyllä. Kaikkia niistä tosi paljon. Se on ihan uskomatonta kuinka nopeasti se motivaatio ja sitoutuminen tulee kun huomaa, että voi vaikuttaa ja päättää. Se on ihan täysin erilaista kuin siinä tapauksessa jos siellä olisi joku tyyppi, joka ei tee sitä työtä, komentamassa että nyt tehdään näin. (H3.)

Myös H1 oli huomannut kehitystiimin motivaation, scrum-projektiin sitoutumisen, tehokkuuden ja tuottavuuden parantuneen, mutta huomautti, ettei se välttämättä johtunut vain kehitystiimin itseohjautuvuudesta. ”Siis kyllä joo, mutta sitä on vaikea sanoa kuinka paljon se johtuu siitä itseohjautuvuudesta ja kuinka paljon ylipäättään siitä kokonaistoimintatavasta, jonka scrum tuo” (H1). H1 toimi scrum-projektissa scrummasterina, mikä saattoi osittain selittää, miksi hänen vastauksensa erosi hieman muiden haastateltavien vastauksista.

6.6 Kehitystiimin yhteistyö

Kuudes teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin yhteistyö*. Scrum-projekteissa vastuu tuotteen kehittämisestä kuuluu koko kehitystiimille. Kehittäjät tekevätkin paljon yhteistyötä tuotteen kehitysjonossa määriteltyjen tuotteen toiminnallisuuden toteuttamisen parissa. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Kehittäjät voivat myös halutessaan toteuttaa työtehtäviä yhdessä muiden kehittäjien kanssa (Sutherland, 2010). Ensimmäiset teemaan liittyvät haastattelukysymykset olivat *tekevätkö kehittäjät yhteistyötä muiden kehittäjien kanssa ja toteuttavatko kehittäjät työtehtäviä yhdessä?* Haastatteluissa tuli esille, että tutkimuskohteena olevassa scrum-projektissa toimivat kehittäjät tekivät yhteistyötä tuotteen toiminnallisuuden toteuttamisen parissa, mutta eivät toteuttaneet työtehtäviä yhdessä. H2, H3 ja H4 kuitenkin korostivat, että vaikka kehittäjät toteuttivat työtehtäviä itsenäisesti, kaikki työtehtävät liittyivät toisiinsa.

Sprintin tehtävälistan työtehtäviä ei toteuteta yhdessä, vaan jos valitsee jonkun työtehtävän sen myös toteuttaa itse. Mutta tuotteen toiminnallisuuden, joihin liittyy useampia työtehtäviä, toteutetaan yhdessä. Eli tehdään jatkuvasti yhteistyötä tuotteen eri toiminnallisuuden liittyen, mutta työtehtäviä ei jaeta. (H3.)

Kolmas teemaan liittyvä haastattelukysymys oli *parantaako kehittäjien tekemä yhteistyö kehittäjien vuorovaikutustaitoja tai kehittäjien välistä vuorovaikutusta?*

Haastateltavien mielestä kehittäjien tekemä yhteistyö ei merkittävästi vaikuttanut kehittäjien vuorovaikutustaitoihin. "En ehkä sanoisi niin, luultavasti vuorovaikutustaidot kehittyvät aina kun tekee yhteistyötä, mutta en tiedä onko se varsinaisesti scrumista lähtöisin" (H2). H1, H3 ja H4 olivat kuitenkin sitä mieltä, että kehittäjien tekemä yhteistyö paransi kehittäjien välistä vuorovaikutusta. "Kommunikointi parantuu sekä scrumtiimin sisällä että asiakkaan suuntaan" (H1). "Me ollaan tehty töitä kauan yhdessä niin vuorovaikutus sujuu jo aika sutjakkaasti, ei tarvitse jännittää mihinkään suuntaan ja aika paljon heitetään myös läppää, välillä vähän liikaakin" (H4). Haastateltavilta kysyttiin myös *tukeeko kehittäjien tekemä yhteistyö ryhmähengen muodostumista?* Haastateltavat olivat yksimielisiä siitä, että kehittäjien tekemä yhteistyö tuki ryhmähengen muodostumista. "Se lisää yhteenkuuluvuuden tunnetta kun tehdään töitä saman projektin parissa ja tiedetään, miten projekti edistyy" (H2).

Viides teemaan liittyvä haastattelukysymys oli *keskitytäänkö scrum-projektissa kehittäjien yksilösuoritukseen vai kehitystiimin yhteiseen suoritukseen?* H1 toi esille, että päiväpalavereissa tarkasteltiin kehittäjien yksilösuorituksia. "Päivittäin katsotaan kehittäjä kerrallaan mitä on tehnyt viimeksi, mitä aikoo tehdä seuraavaksi ja meidän tapauksessa ehtiikö kehittäjä osallistua projektiin" (H1). H2, H3 ja H4 olivat kuitenkin sitä mieltä, että tutkimuskohteena olevassa scrum-projektissa keskityttiin kehittäjien yksilösuorituksia enemmän kehitystiimin yhteiseen suoritukseen. "Yhteiseen suoritukseen, siihen että saadaan työtehtävät valmiiksi" (H3). "Käytännössä se sprintin lopputulos, joka käydään sprinttikatselmuksessa asiakkaan kanssa läpi on se, mitä katsotaan" (H4). H2:n mielestä keskittyminen kehitystiimin yhteiseen suoritukseen vähensi kehittäjiin kohdistuvia paineita. "Aikataulupaine on tavallaan yhteinen, eikä kohdistu yksittäisiin kehittäjiin" (H2). H3 oli kuitenkin sitä mieltä, ettei kehittäjiin kohdistunut paineita tutkimuskohteena olevassa scrum-projektissa. "Kun noudatetaan tuotteen kehitysjonoa ja prioriteettijärjestystä aikataulupaineita ei synny, koska asiakas tietää, että jos jotkut työtehtävät jää toteuttamatta sprintin aikana, ne otetaan työn alle heti seuraavassa sprintissä" (H3). H2:n ja H3:n vastausten pohjalta voidaan todeta, että keskittyminen kehitystiimin yhteiseen suoritukseen vähensi jonkin verran kehittäjiin kohdistuvia paineita, vaikka tutkimuskohteena olevassa scrum-projektissa kehittäjiin kohdistui ylipäätään hyvin vähän paineita.

Kun haastateltavilta kysyttiin *haluavatko kehittäjät työskennellä itsenäisesti vai tehdä yhteistyötä*, heidän vastauksensa jakaantuivat hyvin paljon. H1 kertoi, että kun kehittäjät valitsivat työtehtävän sprintin tehtävälialta, he toteuttivat sen itsenäisesti. "Ei semmoista omimista tapahdu, mutta tietenkin jos jonkin työtehtävän ottaa itselleen työn alle, ei siihen muut sitten koske, vaan silloin se on yksittäisen kehittäjän vastuulla" (H1). H2 uskoi kehittäjien haluavan tehdä yhteistyötä. H3 korosti, että scrum-projektissa ei ollut vaihtoehtoja, vaan kehittäjien täytyi tehdä yhteistyötä. "Ei ole oikein vaihtoehtoja, eli yhdessä tehdään" (H3). H4 puolestaan kertoi kehittäjien työtilanteen ja muiden projektien vaikuttavan siihen, halusivatko kehittäjät työskennellä itsenäisesti vai tehdä yhteistyötä. "Joko tai, se vähän riippuu muiden projektien tilanteesta" (H4). Haastatelta-

vien eriävät vastaukset saattoi selittää se, että haastateltavat saattoivat käsittää eri tavalla, mitä kehitystiimin yhteistyöllä tarkoitettiin. Haastateltavien vastausten pohjalta voidaan kuitenkin todeta, että oli mahdollista, että yksittäiset kehittäjät halusivat työskennellä itsenäisesti eivätkä halunneet tehdä yhteistyötä muiden kehittäjien kanssa.

Viimeinen teemaan liittyvä haastattelukysymys oli *vaikuttavatko kehittäjien vuorovaikutustaidot kehittäjien yhteistyöhön tai ryhmähengen muodostumiseen?* Jokainen haastateltava kertoi, että kehittäjien vuorovaikutustaidot vaikuttivat merkittävästi kehittäjien yhteistyöhön ja ryhmähengen muodostumiseen. ”Kyllä, esimerkiksi jos joku on voimakastahtoinen tai hankala” (H3). H3 kuitenkin huomautti, että tutkimuksen kohdeorganisaatiossa tämä ei ollut ongelma, sillä kohdeorganisaatioon rekrytoitiin ulospäinsuuntautuneita ja sosiaalisia henkilöitä. ”Mutta me ei olla tuotekehitysfirmat, jossa on vain nörttejä, jotka on nenä kiinni näytössä, vaan meille haetaan nimenomaan sellaisia ihmisiä, jotka ovat ulospäinsuuntautuneita ja sosiaalisia” (H3).

6.7 Kehitystiimin työn läpinäkyvyys

Seitsemäs teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin työn läpinäkyvyys*. Kehitystiimin työn läpinäkyvyys tarkoittaa sitä, että kehittäjät ovat jatkuvasti tietoisia toistensa tekemästä työstä, sprintin etenemisestä ja etenemistä hidastavista tekijöistä (Schwaber & Sutherland, 2011; Sutherland, 2010). Ensimmäinen teemaan liittyvä haastattelukysymys olikin *ovatko kehittäjät jatkuvasti tietoisia toistensa tekemästä työstä, sprintin etenemisestä ja etenemistä hidastavista tekijöistä?* Jokaisen haastateltavan mielestä kehitystiimin työn läpinäkyvyys toteutui tutkimuskohteena olevassa scrum-projektissa. ”Kyllä, meillä on yhteinen sprintin tehtävälista, josta me nähdään kokoajan mitä työtehtäviä on työn alla ja mitä on vielä tekemättä ja sitten päiväpalaverissa me kuullaan, jos jollain on esteitä tai muita projekteja, joihin he joutuvat osallistumaan” (H2).

Haastateltavilta kysyttiin myös *mitä motivoi kehitystiimin työn ja sprinttien edistymisen jatkuva seuranta kehittäjiä aikaansaamaan jotain joka päivä?* Haastateltavat olivat yksimielisiä siitä, että kehitystiimin työn ja sprinttien edistymisen jatkuva seuranta motivoi kehittäjiä aikaansaamaan jotain joka päivä. ”Kyllä ehdottomasti, se on yksi tärkeimmistä asioista, koska siitä tulee sellainen kuvitelma, että jotain oikeasti kiinnostaa mitä sä teet” (H3). ”Se motivoi hirveästi, kun näkee, miten sprintin tehtävälista tyhjenee” (H4). H2 ja H4 toivat myös esille, että kehitystiimin työn läpinäkyvyyden ansiosta scrum-projektin eteneminen ei ollut kiinni ainoastaan yhden kehittäjän työpanoksesta, vaan kehittäjät pystyivät tarvittaessa jatkamaan toistensa työtehtäviä. ”Mielestäni on parempi tietää kokoajan mitä toinen tekee. Jos tulee esimerkiksi sairastumisia ja deadline lähes tyy, muut kehittäjät osaavat sitten jatkaa siitä.” (H4.)

Kolmas teemaan liittyvä haastattelukysymys oli *voivatko kehittäjät raportoida kohtaamistaan ongelmista heti niiden ilmaannuttua ja pyytää tarvittaessa apua*

muilta kehittäjiltä tai scrummasterilta? Jokainen haastateltava toi esille, että kehittäjien oli mahdollista raportoida kohtaamistaan ongelmista heti niiden ilmaannuttua ja pyytää tarvittaessa apua muilta kehittäjiltä tai scrummasterilta. ”Kyllä ja ilmenevistä ongelmista raportoidaan rehellisesti heti niiden ilmaannuttua eikä yritetä yksin korjata niitä” (H4).

Ongelmista raportoidaan sekä päiväpalaverissa että heti. Jos on joku sellainen ongelma, joka pysäyttää työt, silloin apua kysytään yleensä heti. Mutta jos on joku sellainen ongelma, joka ei tällä hetkellä haittaa työn tekemistä, mutta olisi kiva selvittää, apua kysytään yleensä päiväpalaverissa. (H3.)

Kun haastateltavilta kysyttiin *vähentääkö työn läpinäkyvyys tuotteeseen liittyviä virheitä ja helpottaa ja nopeuttaa virheiden korjaamista*, jokainen haastateltava kertoi, että kehitystiimin työn läpinäkyvyys nopeutti jonkin verran tuotteeseen liittyvien virheiden korjaamista. ”Korjausaikaan se voi vaikuttaa jonkin verran, kun ollaan kaikki tietoisia siitä virhetilanteesta ja saatetaan sitten nopeammin tarttua siihen sen korjaamiseen” (H1). Kukaan haastateltavista ei kuitenkaan uskonut kehitystiimin työn läpinäkyvyyden vähentävän tuotteeseen liittyviä virheitä tai helpottavan virheiden korjaamista. ”Lopulliseen toteutusjälkeen sillä varmaan ei ole niin suurta merkitystä, sillä se on yksittäinen kehittäjä joka sen ratkaisun tai korjauksen tekee, eikä scrum voi hirveästi vaikuttaa kehittäjien koodaustaitoon” (H1).

Viides teemaan liittyvä haastattelukysymys oli *ilmeneekö kehittäjien heikko osaaminen, huono työnjälki tai sovituisia työtehtävistä ja aikatauluista poikkeaminen helposti scrum-projektissa?* Haastateltavat olivat yksimielisiä siitä, että kehittäjien heikko osaaminen, huono työnjälki sekä sovituisia työtehtävistä ja aikatauluista poikkeaminen tuli nopeasti esille scrum-projektissa. ”Se tulee nopeammin esille kaikille osapuolille, myös asiakkaalle” (H1). Seuraavat teemaan liittyvät haastattelukysymykset olivat *lisääkö työn läpinäkyvyys kehittäjiin kohdistuvia paineita ja voiko työn läpinäkyvyys vaikuttaa negatiivisesti työskentelyilmapiiriin?* Kukaan haastateltavista ei ollut sitä mieltä, että kehitystiimin työn läpinäkyvyys lisäsi kehittäjiin kohdistuvia paineita tai vaikutti negatiivisesti työskentelyilmapiiriin. ”Jos on jokin hankala homma, muut auttaa ja jos joku tekee virheen, se korjataan. Tämä on kuitenkin vain työtä, etkä sä ole huono ihminen, jos et osaa jotain asiaa, sä olet ehkä tyhmä tyyppi, jos et opi sun virheistä.” (H3.)

Me ollaan suhteellisen kannustavia ja varsinkin nyt kun ollaan tutustuttu on helppo sanoa toiselle, että hei tässä olisi vielä tämä tekemättä. Kun syntyy se keskinäinen luottamus on paljon helpompi antaa myös palautetta ja kritiikkiä koska silloin ymmärtää sen, ettei toinen tarkoita pahaa tai halua loukata vaan tehdä työn paremmin. (H3.)

Viimeinen teemaan liittyvä haastattelukysymys oli *voivatko kehittäjät tuntea scrum-projektin tapahtumiin osallistumisen ja tekemänsä työn esittelyn muille kehittäjille epämukavaksi?* Kukaan haastateltavista ei ollut kohdannut tilannetta, jossa kehittäjät olisivat tunteneet scrum-projektin tapahtumiin osallistumista tai tekemänsä työn esittelyä muille kehittäjille epämiellyttäväksi. ”Vaikka olisi spesi-

fimpikin työtehtävä, sen esittäminen on luontevaa, muut kehittäjät saattavat vain kysyä tarkennusta” (H4). H3 kertoi tämän johtuvan siitä, että kehittäjät suunnittelivat työtehtävät yhdessä sprintin suunnittelupalaverissa.

Se riippuu siitä, millä tasolla työtehtävistä kerrotaan. Mehän sprintin suunnittelupalaverissa yhdessä suunnitellaan, kirjoitetaan ylös ja käydään läpi sprintin aikana toteutettavat työtehtävät, jolloin kaikki tietävät mistä työtehtävästä on kyse ja silloin me yleensä päiväpalaverissa vain kerrotaan minkä työtehtävän olemme toteuttaneet. Me ei siis mennä tekniselle tasolle. (H3.)

6.8 Kehitystiimin työmäärä ja työskentelytahti

Kahdeksas teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin työmäärä ja työskentelytahti*. Ensimmäinen teemaan liittyvä haastattelukysymys oli *auttaako työskentely sprinteissä pitämään kehitystiimin työmäärän kohtuullisena koko scrum-projektin ajan?* Jokaisen haastateltavan mielestä työskentely sprinteissä auttoi pitämään tutkimuskohteena olevassa scrum-projektissa toimivan kehitystiimin työmäärän kohtuullisena koko scrum-projektin ajan. ”Kaiken työn, mitä me tehdään, täytyy olla työtehtävänä sprintin tehtävälillä ja jos se ei ole, me lisätään se tehtävälille ja annetaan sille työmääräarvio” (H4). ”Tälläkään hetkellä ei ole sellaista isoa työmäärää jonossa, joka pitäisi saada puolen vuoden päästä valmiiksi, vaan parin seuraavan viikon työ on tiedossa ja me tiedämme, että pystymme toteuttamaan sen” (H2). Haastateltavien vastausten pohjalta voidaan myös olettaa, että kohtuullisen työmäärän ansiosta kehitystiimin työskentelytahti pysyi tasaisena ja miellyttävänä.

Pääsääntöisesti se auttaa, että me suunnitellaan työtehtävät ja annetaan niille työmääräarviot sprintin alussa. Jos työtehtävän toteuttaa arvioitua nopeammin, se on vain palkitsevaa, että voi hetken hengähtää ennen seuraavan työtehtävän aloittamista. Ja jos puolestaan aletaan jäämään jälkeen, voidaan sprintin tehtäväliltä pudottaa jokin työtehtävä pois. (H3.)

Haastatteluissa tuli myös esille, että työskentely sprinteissä ja useat scrum-projektin tapahtumat ja tuotokset auttoivat kehitystiimiä pysymään aikataulusa. ”Säännölliset deadlinet, jolloin asiakkaalle täytyy esittää jotain motivoivat kummasti” (H1). ”Se on positiivinen puoli, että kun pitää (sprintin tehtäväliltä) priorisointijärjestyksestä kiinni, ei tule sitä tilannetta, että aloittaisi helpoimmista työtehtävistä ja jättäisi vaikeat ja ikävät työtehtävät viimeiseksi ja tekisi ne vasta sitten kun on pakko. Näin tulee tehtyä kaikki työtehtävät ajallaan.” (H2.)

Toinen teemaan liittyvä haastattelukysymys oli *väheneekö ylitöiden määrä scrumin ansiosta?* Kukaan haastateltavista ei osannut kertoa, väheneekö ylitöiden määrä scrumin ansiosta, sillä tutkimuskohteena olevassa scrum-projektissa toimivat kehittäjät eivät yleensä tehneet ylitöitä. ”En osaa sanoa, koska me ei tehdä ylitöitä” (H3). H1 ja H2 toivat kuitenkin esille, että scrum-projektin ul-

kopuoliset tekijät, kuten kehittäjien muut samanaikaiset projektit tai poissaolot, saattoivat lisätä kehittäjien työmäärää ja aiheuttaa ylitöitä. ”Enemmän kuormitukseen vaikuttaa projektin ulkopuoliset tekijät, kuten muut projektit, lomat ja sairaspöissaolot” (H1). ”Ylitöiden tekeminen ei ole ollut yleistä meidän tiimissä, joten siinä mielessä on vaikea verrata, mutta juuri nyt sattuu olemaan sellainen tilanne, että teimme muutaman viikon aikana hieman ylitöitä johtuen muutamasta päällekkäisestä projektista” (H1). H3 ja H4 puolestaan huomauttivat, että asiakkaan kiireelliset vaatimukset saattoivat lisätä kehittäjien työmäärää ja aiheuttaa ylitöitä. ”Joskus on toki kiireellisiäkin sprinttejä, jos asiakas haluaa saada paljon asioita toteutettua yhden sprintin aikana” (H3). ”Ylitöitä syntyy, jos asiakas esittää kiireellisiä vaatimuksia eikä kehittäjät millään ehdi toteuttaa työtä normaalin työajan puitteissa” (H4).

Kolmas teemaan liittyvä haastattelukysymys oli *onko realististen työmääräarvioiden tekeminen helppoa?* Jokaisen haastateltavan mielestä realististen työmääräarvioiden tekeminen oli kehitystiimille helppoa. ”Kyllä se aika toimivasti tuntuu sujuvan” (H1). H3 ja H4 kuitenkin huomauttivat, että realististen työmääräarvioiden tekeminen saattoi olla vaikeaa, jos asiakkaan tuotetta koskevat vaatimukset olivat epäselvät tai työtehtävistä ei ollut aikaisempaa kokemusta. ”Alussa se oli hieman hankalaa, kun ei tiedetty mitä asiakas haluaa” (H4).

Se on ihan siitä kiinni, paljonko sä olet tehnyt sitä työtä ja tunnetko tuotteen. Tällä porukalla kun tunnetaan jo käytetyt työkalut se on helppoa. Mutta jos me ei tiedetä, miten jokin (tuotteen) toiminnallisuus toteutetaan, sitten pitää vain osata laittaa sen toteuttamiselle enemmän tunteja. Eli alkuvaikeuksien jälkeen se on suhteellisen helppoa. (H3.)

Haastateltavilta kysyttiin myös *voiko kehitystiimi sitoutua tekemään liian paljon työtä yhden sprintin aikana?* H1 ja H2 kertoivat, ettei kehitystiimi ollut toistaiseksi sitoutunut toteuttamaan liian paljon työtä yhden sprintin aikana. ”Toistaiseksi me ei olla sitouduttu tekemään liikaa työtä, ehkä me ollaan varovaisia niissä työmääräarvioissa” (H2). H3 ja H4 olivat kuitenkin sitä mieltä, että oli mahdollista, että kehitystiimi sitoutui tekemään liian paljon työtä yhden sprintin aikana, mutta huomauttivat, että asiakas oli hyvin ymmärtäväinen, jos kehittäjät eivät ehtineet toteuttaa kaikkia sovittuja työtehtäviä sprintin aikana. ”Alimman prioriteetin työtehtäviä voidaan myös pudottaa pois sprintin tehtävälialta, jolloin me ollaan sitten ilmoitettu asiakkaalle, että tätä me ei nyt ehditä toteuttaa ja he ovat olleet hyvin joustavia” (H4).

Voi. Mutta kyllä me jo sprintin suunnittelupalaverissa, kun suunnitellaan työtehtävät ja annetaan työmääräarviot, nähdään se tilanne, että pitääkö joitakin työtehtäviä tiputtaa pois. Ja asiakas on ymmärtäväinen, koska se työ on läpinäkyvää myös heidän suuntaan. Ja asiakas luottaa meihin juuri siitä syystä. (H3.)

Viides teemaan liittyvä haastattelukysymys oli *voivatko scrum-projektin tapahtumat pitkittyä?* Jokainen haastateltava kertoi, että scrum-projektin tapahtumat pitkittyivät, jos tapahtumaan osallistuvat henkilöt poikkesivat käsiteltävistä

aiheista. ”Voivat kyllä, välillä menee ihan turhaa aikaa kun poiketaan asiasta” (H4). H2 ja H3 kuitenkin korostivat, että silloin kun tapahtumat noudattivat scrumin viitekehystä, tapahtumat eivät pitkittyneet. ”Päiväpalavereiden ideahan on se, että niissä puhutaan vain ne sovitut asiat, mutta joskus se kuitenkin lipsahtaa esimerkiksi pohdintaan teknisestä toteutuksesta” (H2). ”Voivat, mutta silloin kun homma pyörii niin kuin pitää, niin ei käy” (H3).

Viimeiset teemaan liittyvät haastattelukysymykset olivat *onko scrum-projektin tapahtumille varattu sopivasti aikaa ja onko sopivan ajankohdan löytäminen scrum-projektin tapahtumille helppoa?* H2, H3 ja H4 olivat sitä mieltä, että scrum-projektin tapahtumille oli varattu sopivasti aikaa käsiteltäviin asioihin nähden ja että sopivan ajankohdan löytäminen scrum-projektin tapahtumille oli helppoa. H1 oli kuitenkin eri mieltä muiden haastateltavien kanssa. Syynä H1:n eriäviin vastauksiin oli todennäköisesti se, että H1 toimi tutkimuskohteena olevassa scrum-projektissa scrummasterina ja oli vastuussa scrum-projektin tapahtumien järjestämisestä. H1 kertoi, että päiväpalaverit olivat sopivan pituisia, mutta sprintin suunnittelupalaverit ja sprinttikatselmuksset olivat haastavia ajankäytön suhteen. ”Sprinttikatselmuksset ja varsinkin sprintin suunnittelupalaverit ovat haastavampia ajankäytön suhteen, kun mietitään, mikä on se riittävä aika asioiden läpikäymiseen ja suunnitteluun” (H1). H1 oli myös sitä mieltä, että sopivan ajankohdan löytäminen scrum-projektin tapahtumille oli vaikeaa.

Ei ole helppoa. Kaikilla on usein muita kalenterivaroituksia, erityisesti itselläni. Ja sen lisäksi asiakkaan tuotemistajan ja muiden asiakkaan projektiryhmään kuuluvien avainhenkilöiden kanssa sopivan ajankohdan löytäminen on haastavaa. (H1.)

6.9 Kehitystiimin tuottaman dokumentaation määrä

Yhdeksäs teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin tuottaman dokumentaation määrä*. Scrum ei ota kantaa siihen, minkälaista dokumentaatiota kehitystiimin pitäisi tuottaa (Coram & Bohner, 2005; Overhage ym., 2011). Scrum-projekteissa kehitystiimi ei yleensä tuota dokumentaatiota, jota jouduttaisiin päivittämään scrum-projektin edetessä (Selic, 2009). Scrum-projekteissa kehittäjien ei tarvitse tehdä esimerkiksi vaatimusmäärittelyjä tai projektisuunnitelmia, joissa ei muuttuvien vaatimusten ja teknologia- ja liiketoimintaympäristön muutosten vuoksi olisi mahdollista pitäytyä (Zhi-gen ym., 2009). Ensimmäinen teemaan liittyvä haastattelukysymys oli *minkälaista dokumentaatiota kehitystiimi tuottaa?* Haastateluissa tuli esille, että tutkimuskohteena olevassa scrum-projektissa toimiva kehitystiimi tuotti lähinnä tuotteen tekniseen toteutukseen liittyvää dokumentaatiota. Tekninen dokumentaatio ei kuitenkaan ollut liian yksityiskohtaista, vaan kehittäjät pitivät dokumentaation lyhyenä, selkeänä, rakenteeltaan loogisena ja käyttivät paljon ruudunkaappauskuvia, joiden avulla heidän oli helppo esitellä asiakkaalle kehittämäänsä tuotetta. H1 kertoi, että kehitystiimi tuotti melko vähän hallinnollista dokumentaatiota. ”Esimerkiksi päiväpalavereista ei

synny mitään dokumentaatiota, eikä oikeastaan ole tarpeenkaan, mutta sprinttien suunnittelupalavereista ja sprinttikatselmuksista tehdään jonkinlaiset muistiot” (H1). H3 puolestaan korosti, että kehitystiimin tuottaman dokumentaation määrä oli vähäinen perinteisiin ohjelmistokehitysprojekteihin verrattuna. ”Mutta vanhanaikaisiin ohjelmistokehitysprojekteihin nähden dokumentaatiota syntyy vähän” (H3).

Haastateltavilta kysyttiin myös *kuinka paljon dokumentaation tuottaminen vie kehittäjien aikaa?* H1 ja H2 kertoivat, että dokumentaation tuottaminen vei jonkin verran kehittäjien aikaa, mutta huomauttivat dokumentoinnin olevan olennainen osa ohjelmistokehitystä. ”Tottakai jonkin verran, mutta dokumentointi on vähän pakollista, samoin kuin suunnittelu ja testaus, scrum ei poista sitä, eikä sen kuuluisikaan poistaa” (H1). H3 ja H4 puolestaan kertoivat, ettei dokumentaation tuottaminen vienyt paljoa kehittäjien aikaa, sillä dokumentointi sisällytettiin valmiin määritelmään ja työmääräarvioihin.

Ei vie, koska se kuuluu meillä valmiin määritelmään, eli työtehtävä ei ole valmis, ennen kuin se on dokumentoitu. Muuten sitä ei tulisi tehtyä. Jos on esimerkiksi kahden työtunnin työtehtävä, dokumentointi ja testitapauksen tekeminen vie noin 15 minuuttia, jos sitäkään. Se aika, joka niihin menee on 10-15% koko kokonaistyömäärästä. Ja kun siihen tottuu, siinä ei mene kauaa. (H3.)

Kolmas teemaan liittyvä haastattelukysymys oli *syntyykö kehitystiimin keskuudessa dokumentaation lisäksi myös hiljaista tietoa?* Jokainen haastateltava kertoi, että kehittäjät jakoivat scrum-projektiin liittyvää tietoa toisillensa, minkä seurauksena kehitystiimin keskuudessa syntyi hiljaista tietoa. ”Kyllä, nimenomaan sellaista tietoa, joka ei ole kenenkään omassa päässä, vaan koko kehitystiimin päässä” (H2). Haastateltavilta kysyttiin myös *voiko oleellista tietoa kadota, jos kehittäjät vaihtuvat tai lähtevät kehitystiimistä kesken scrum-projektin?* Haastateltavat olivat yksimielisiä siitä, ettei oleellista tietoa voinut kadota, vaikka kehittäjät vaihtuivat tai lähtivät kehitystiimistä kesken scrum-projektin. ”Ei yksittäisen kehittäjän poistuessa, koska jaamme toisillemme sen verran projektiin liittyvää tietoa” (H2). Haastateltavien vastausten pohjalta voidaan todeta, että kehitystiimin keskuudessa syntyneen hiljaisen tiedon ansiosta oleellista tietoa ei voinut kadota, vaikka kehittäjät olisivat vaihtuneet tai lähteneet kehitystiimistä kesken scrum-projektin.

Viimeinen teemaan liittyvä haastattelukysymys oli *helpottaako olemassa oleva dokumentaatio uusien tai kokemattomampien kehittäjien perehdyttämistä?* H1, H3 ja H4 olivat sitä mieltä, että olemassa oleva, kehitystiimin tuottama dokumentaatio helpotti uusien tai kokemattomampien kehittäjien perehdytystä. ”No ainakin siinä mielessä, että aikalailla kaikki tekniseen ratkaisuun liittyvä on dokumentoitu” (H1). ”Kyllä, ei tarvita mitään monien päivien perehdytyksiä, vaan dokumentaation pohjalta on mahdollista lähteä toteuttamaan työtehtäviä” (H4). H2 oli kuitenkin eri mieltä muiden haastateltavien kanssa. ”Dokumentaatio ei varmasti ole niin kattavaa, että siitä olisi perehdytyksessä apua” (H2). Vaikka haastateltavien vastaukset erosivat jonkin verran toisistaan, haastateltavien vastausten pohjalta voidaan olettaa, että uusien tai kokemattomampien

kehittäjien perehdyttäminen olisi voinut olla vaikeaa, jos dokumentaatiota ei olisi ollut.

6.10 Kehitystiimin yhteistyö tuoteomistajan kanssa

Kymmenes teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin yhteistyö tuoteomistajan kanssa*. Ensimmäiset teemaan liittyvät haastattelukysymykset olivat *onko tuoteomistaja sitoutunut osallistumaan scrum-projektiin ja tekeekö kehitystiimi säännöllistä yhteistyötä tuoteomistajan kanssa?* Haastatteluluissa tuli esille, että tutkimuskohteena olevassa scrum-projektissa tuoteomistaja oli sitoutunut osallistumaan scrum-projektiin ja kehitystiimi teki säännöllistä yhteistyötä tuoteomistajan kanssa. ”Kyllä. Tuoteomistaja on mukana sprinttien suunnittelupalavereissa ja sprinttikatselmuksissa. Tuoteomistaja ei osallistu päiväpalavereihin, mutta me ollaan tarvittaessa yhteyksissä tuoteomistajaan myös sprintin aikana.” (H2.) ”Kyllä, tuoteomistajan kanssa keskustellaan paljon toteutuksessa olevista toiminnallisuuksista” (H4).

Meillä on tuoteomistajana asiakkaan puolelta asiakkaan projektipäällikkö. Sprintin suunnittelupalavereissa ja sprinttikatselmuksissa on mukana asiakkaan puolelta myös pari muuta henkilöä, vaikka näinhän ei pitäisi olla, mutta asiakkaalle on kauden vaikea perustella, että teiltä saa osallistua vain yksi henkilö. Eli se ei toimi ihan niin, että olisi vain yksi tuoteomistaja, mutta tämä pieni ryhmä asiakkaan puolelta on kuitenkin sitoutunut osallistumaan scrum-projektin tapahtumiin ja tekee työnsä. (H3.)

Kolmas teemaan liittyvä haastattelukysymys oli *saako kehitystiimi tuoteomistajalta säännöllisesti palautetta tekemästään työstä?* Jokainen haastateltava kertoi, että kehitystiimi sai tuoteomistajalta säännöllisesti palautetta tekemästään. ”Kyllä, ainakin sprinttikatselmuksien yhteydessä, joissa katsotaan mitä on tehty, he kommentoivat näyttääkö silti miltä piti” (H2). ”Kyllä, sekä suullista palautetta että kirjallista palautetta asiakastytyväisyyskyselyjen muodossa” (H4). Haastateltavilta kysyttiin myös *kasvattaako yhteistyö tuoteomistajan kanssa kehitystiimin luottamusta siihen, että kehitettävä tuote vastaa asiakkaan vaatimuksia?* Jokaisen haastateltavan mielestä yhteistyö tuoteomistajan kanssa kasvatti kehitystiimin luottamusta siihen, että kehitettävä tuote vastasi asiakkaan vaatimuksia. ”Kyllä, hyvä puoli on siinä, että me päästään asiakkaan kanssa kokoajan keskustelemaan siitä, mitä tehdään” (H2). Haastatteluluissa tuli myös esille, että säännöllisen yhteistyön ansiosta kehitystiimin oli mahdollista kysyä nopeasti selvennystä tai neuvoa tuoteomistajalta esimerkiksi tuotetta koskeviin vaatimuksiin liittyen. ”Oikeastaan samantien ollaan asiakkaaseen yhteydessä, jos tulee isompi ongelma tai epäselvyys siitä, mitä asiakas tarkoittaa” (H4).

Viides teemaan liittyvä haastattelukysymys oli *edellyttääkö yhteistyö tuoteomistajan kanssa kehittäjiltä asiakkaan toimialaan ja liiketoimintaan liittyvää tietoa?* Jokainen haastateltava oli sitä mieltä, että tutkimuskohteena olevassa scrum-projektissa yhteistyö tuoteomistajan kanssa ei edellyttänyt kehittäjiltä asiakkaan

toimialaan ja liiketoimintaan liittyvää tietoa. Haastateltavat kuitenkin huomauttivat, että asiakkaan toimialaan ja liiketoimintaan liittyvästä tiedosta olisi ollut hyötyä. ”Ei edellytä tässä projektissa, mutta olisi se etu” (H3). ”Kyseisessä projektissa ei ole niin operatiivinen järjestelmä kyseessä, joten se ei ole niin keskeistä, kuin ehkä jossain toisessa ohjelmistoprojektissa, jossa suunniteltaisiin esimerkiksi liiketoimintaa vahvasti tukevaa järjestelmää. Mutta tottakai siitä on paljon hyötyä.” (H1.) Haastateltavien vastausten pohjalta voidaan olettaa, että asiakkaan toimialaan ja liiketoimintaan liittyvän tiedon puuttuminen olisi voinut toisenlaisessa scrum-projektissa vaikeuttaa yhteistyötä tuoteomistajan kanssa.

Viimeinen teemaan liittyvä haastattelukysymys oli *edellyttääkö yhteistyö tuoteomistajan kanssa kehittäjiltä hyviä vuorovaikutus- ja esiintymistaitoja?* Jokainen haastateltava oli ehdottomasti sitä mieltä, että yhteistyö tuoteomistajan kanssa edellytti kehittäjiltä hyviä vuorovaikutus- ja esiintymistaitoja. H2, H3 ja H4 toivat esille, että yksi merkittävä haaste oli kehittäjien käyttämä liian tekninen kieli, jota tuoteomistajan saattoi olla vaikea ymmärtää. ”Kyllä. Siinä on tietenkin kehittämisen varaa, sillä kehittäjät saattavat sortua puhumaan teknisiä yksityiskohtia ja käyttämään teknisiä termejä, joita tuoteomistaja ei ymmärrä.” (H2.) ”En tiedä täytyykö kehittäjien osata pitää mitään hienoja puheita, eikä se vaadi mitään sellaisia taitoja, että osaako käyttää tilaa oikein tai puhua mikrofonin oikein. Mutta kyllä siinä kohtaa, kun esitellään mitä ollaan tehty, kehittäjien täytyy osata kertoa se asiakkaan ymmärtämällä kielellä.” (H3.) Haastateltavien vastausten pohjalta voidaan olettaa, että jos kehittäjien vuorovaikutus- ja esiintymistaidot olisivat olleet huonot, yhteistyö tuoteomistajan kanssa olisi voinut olla vaikeaa. H1 toi kuitenkin esille, että säännöllinen yhteistyö tuoteomistajan kanssa paransi kehittäjien esiintymistaitoja ja ammattitaitoa.

Sprinttikatselmuksissa esitysvastuu on annettu kehittäjille, eli he itse demoavat edellisen sprintin lopputuloksen asiakkaalle, mitä ei perinteisissä projekteissa välttämättä tulisi kehittäjille ollenkaan tehtäväksi. Enemmän kuitenkin näen sen scrumin hyvänä puolena, sillä kehittäjät kasvavat sekä esiintymistaidoissaan että ammattillisesti joutuessaan esittämään ratkaisun muille ja miettimään asioita kaksi kertaa, esimerkiksi toimiiko ratkaisu todella toivotulla tavalla. (H1.)

6.11 Vaatimusten muuttuminen scrum-projektin aikana

Viimeinen teema, johon liittyviä hyötyjä ja haasteita tapaustutkimuksessa tutkittiin kehitystiimin näkökulmasta oli *vaatimusten muuttuminen scrum-projektin aikana*. Scrum-projekteissa kehitettävää tuotetta koskevat vaatimukset eivät ole ennalta määriteltyjä, vaan tuoteomistaja voi muuttaa kehitettävää tuotetta koskevia vaatimuksia scrum-projektin edetessä. Sprintin aikana toteutettavia tuotteen toiminnallisuuksia ei kuitenkaan voida muuttaa kesken sprintin, vaan tuoteomistaja voi tuoda esille muuttuneet vaatimukset sekä teknologia- ja liiketoimintaympäristön muutokset sprintin lopuksi järjestettävässä sprinttikatselmuk-

sessä. (Schwaber & Sutherland, 2011; Sutherland, 2010.) Haastatteluissa tuli esille, että näin oli myös tutkimuskohteena olevassa scrum-projektissa.

Ensimmäinen teemaan liittyvä haastattelukysymys oli *uskovatko kehittäjät voivansa selvittää kehitettävää tuotetta koskevat vaatimukset scrum-projektin alussa?* Kukaan haastateltavista ei ollut kohdannut tilannetta, jossa kehittäjät olisivat uskoneet voivansa selvittää kehitettävää tuotetta koskevat vaatimukset jo scrum-projektin alussa. Kehittäjistä kaksi oli käyttänyt aikaisemmin perinteisiin ohjelmistokehitysmenetelmiin kuuluvaa vesiputousmallia, jonka mukaisissa ohjelmistokehitysprojekteissa kehitettävää tuotetta koskevat vaatimukset määriteltiin jo ohjelmistokehitysprojektin alussa, ennen suunnittelu- ja toteutusvaiheita, mutta tämä ei vaikuttanut heidän vastauksiinsa. H2 ja H3 toivat esille, että scrum-projekteissa kehittäjät eivät pyrkineet selvittämään vaatimuksia näin aikaisessa vaiheessa, sillä vaatimusten muuttuminen scrum-projektin aikana kuului niin vahvasti scrumin ideologiaan. ”Eivät, siitäkin johtuen, että ei ole edes yritetty selvittää niitä” (H2).

Koko scrumin ideahan on se, että asiakas ohjaa laivaa. Ja kun (scrum-projektin) alussa on tehty jonkinlainen tuotteen kehitysajon, joka toimii ikään kuin toiveiden tynnyrinä, ja lähdetään sitten rakentamaan tuotetta sen pohjalta, niin kyllä jo parin sprintin jälkeen nähdään, että monet alkuperäisistä vaatimuksista ovat jääneet pois. (H3.)

Toinen teemaan liittyvä haastattelukysymys oli *miten kehittäjät suhtautuvat siihen, että tuoteomistaja voi muuttaa tuotetta koskevia vaatimuksia scrum-projektin aikana?* Jokainen haastateltava kertoi, että kehittäjät suhtautuivat positiivisesti kehitettävää tuotetta koskevien vaatimusten muuttumiseen scrum-projektin aikana. H1 ja H2 korostivat, että vaatimusten muuttuminen kesken ohjelmistokehitysprojektin oli todella yleistä, joten oli hyvä, että oli olemassa ohjelmistokehityksen viitekehys, joka loi lähtökohdat muuttuvien vaatimusten huomioimiselle. ”Se on vaan fakta, että niin useimmissa ohjelmistoprojekteissa käy ja on sitten kanava sille” (H1). ”On siitä se hyöty, että muutoksiahan tulee aina ja nyt se tiedetään jo etukäteen, että niitä tulee olemaan” (H2). Haastateltavilta kysyttiin myös *huomioiko kehitystiimi muuttuneet vaatimukset nopeasti?* Jokainen haastateltava korosti, että kehitystiimi huomioi muuttuvat vaatimukset hyvin nopeasti, jo seuraavassa sprintissä. ”Kyllä, heti kun seuraava sprintti alkaa” (H2). ”Kesken sprintin toiminnallisuuksia ei saa muuttaa, kuten scrum määrää, mutta me tehdään sitten (muuttuneista vaatimuksista) uusi tuotteen kehitysajon kohta, joka nostetaan korkeimmalle prioriteetille ja valitaan heti seuraavaan sprinttiin” (H3). Haastatteluissa tuli myös esille, että muutostarpeiden nopean huomioimisen ansiosta kehitettävä tuote ja asiakkaan tarpeet vastasivat paremmin toisiinsa. ”Kun muutoksia tulee, ollaan lähempänä asiakkaan tarvetta, kuin jos oltaiisiin tehty alussa määrittely, jota noudatettaisiin orjallisesti” (H1). ”Useimmiten asiakas on tyytyväinen siitä, että ne saa sen, mitä tarvitsee, eikä sitä, mikä on aluksi keksitty, kun on jotain pitänyt keksiä” (H3).

Viimeinen teemaan liittyvä haastattelukysymys oli *voiko tuoteomistaja määrittellä tuotetta koskevat vaatimukset huonosti?* Jokainen haastateltava kertoi, että hyvin yleistä, että tuoteomistaja määritteli tuotetta koskevat vaatimukset huo-

nosti. "En ole täysin varma, tietääkö tuoteomistaja itse, minkälainen sen lopullisen tuotteen pitäisi olla" (H2). "Käytännössähän asiakas osaa sanoa jotain vastaus, kun he näkevät jotain konkreettista, koska he eivät osaa päässään kuvitella ratkaisua niin kuin me (kehittäjät), jotka vastaamme sen toteutuksesta" (H3). H3 toi kuitenkin esille, että mitä enemmän tuoteomistaja teki yhteistyötä kehitystiimin kanssa, sitä paremmin hän osasi esittää kehitettävää tuotetta koskevia vaatimuksia. "Kun me projektin aikana opitaan yhteiset työskentelytavat niin me huomataan, että hän (tuoteomistaja) osaa kertoa paremmin, mitä hän tai he haluavat ja hän myös ymmärtää, että meiltä saa pyytää asioita" (H3). H1 huomautti, että jos tuoteomistaja oli määritellyt tuotetta koskevat vaatimukset huonosti, sprinttikatselmuksissa jouduttiin tekemään korjausliikkeitä. "Yleensä siinä katselmointivaiheessa joudutaan tekemään sitten niitä korjausliikkeitä sen takia, että se yhteisymmärrys ei kuitenkaan ole ollut niin hyvä" (H1). H3 kertoi, että kehittäjät joutuivat usein opastamaan tuoteomistajaa kehitettävää tuotetta koskeviin vaatimuksiin liittyen. "Tässä projektissa me enemmänkin ohjataan tuoteomistajaa" (H3). H2, H3 ja H4 toivat esille, että kehittäjät joutuivat usein myös pyytämään tuoteomistajaa tarkentamaan vaatimuksiaan ja luomaan esimerkkiratkaisuja, joiden avulla voitiin havainnollistaa tuoteomistajalle, miten kehitettävä tuote voisi toimia.

Kyllä me siinä kohtaa nostamme kädet pystyyn ja sanomme, ettei me näillä tiedoilla päästä etenemään ja kyselemme lisätietoa. Olemme myös tehneet esimerkkiratkaisuja, että asiakas näkee minkälainen ratkaisu todellisuudessa on, jonka jälkeen olemme joko iteroineet ratkaisua eteenpäin valmiimmaksi tai haudanneet sen. (H4.)

6.12 Yhteenveto

Tämän luvun tarkoitus oli tuottaa yksityiskohtaista, kirjallisuuskatsauksen tutkimustuloksia täsmentävää, scrumin hyötyihin ja haasteisiin kehitystiimin näkökulmasta liittyvää tietoa tutkimuskohteeksi valittua scrum-projektia tutkimalla. Tapaustutkimuksessa esille tulleet scrumin hyödyt ja haasteet kehitystiimin näkökulmasta on tiivistetty seuraavaan taulukkoon (taulukko 3).

TAULUKKO 3 Tapaustutkimuksessa esille tulleet scrumin hyödyt ja haasteet kehitystiimin näkökulmasta

	Hyödyt ja haasteet
Scrumin käyttöönotto	<ul style="list-style-type: none"> + Oikein noudatettuna scrum tuo useita hyötyjä ohjelmistokehitysprojekteihin. + Kehittäjät pitävät työskentelystä scrum-projekteissa. - Kehittäjien mielestä scrumin viitekehys voi olla monimutkainen ja vaikeasti opittava. - Scrumin käyttöönotto voi olla työlästä, aikaavievää ja haastavaa. - Kehittäjät voivat vastustaa scrumin käyttöönottoa. - Jos scrumitiimi muuttaa scrumin viitekehystä tai käyttää vain osaa viitekehysten rooleista, tapahtumista ja tuotoksista, scrumin hyödyt voi-

	vat jäädä saavuttamatta.
Kehitystiimin koko	<ul style="list-style-type: none"> + Scrum soveltuu hyvin pienten kehitystiimien käyttöön. + Kehitystiimin pieni koko mahdollistaa suoran ja jatkuvan vuorovaikutuksen sekä nopean tiedon jakamisen kaikkien kehittäjien välillä. - Yrityksillä ei välttämättä ole tarpeeksi resursseja muodostaa optimaalisen kokoisia kehitystiimejä. - Jos kehittäjät toimivat samaan aikaan useissa eri projekteissa, heidän voi olla vaikea osallistua scrum-projektiin.
Kehitystiimin työskentely-ympäristö	<ul style="list-style-type: none"> + Kehitystiimit pyrkivät työskentelemään avoimessa työtilassa. + Työskentely avoimessa työtilassa mahdollistaa kasvokkain käytävän keskustelun ja tukee kehittäjien välistä vuorovaikutusta ja yhteistyötä. + Viestintäteknologian avulla voidaan tukea eri paikassa työskentelevien kehittäjien välistä vuorovaikutusta ja yhteistyötä. - Avoimessa työtilassa on paljon häiriötekijöitä, jotka voivat vaikeuttaa kehittäjien työntekoa. - Kehittäjien ei ole aina mahdollista työskennellä samassa paikassa.
Kehitystiimin monitaitoisuus	<ul style="list-style-type: none"> + Kehitystiimi muodostuu pienestä joukosta ammattilaisia, joilla on tarvittava osaaminen toteuttaa scrum-projekti ilman ulkopuolista apua. + Kehittäjät voivat toteuttaa sekä työtehtäviä, joiden toteuttamiseen heillä on tarvittava osaaminen että työtehtäviä, joiden toteuttamisen he haluavat oppia. + Työskentely monitaitoisissa kehitystiimissä tukee kehittäjien välistä tiedon jakamista ja parantaa kehittäjien osaamista. + Osaamisen kehittyessä scrum-projektin eteneminen ei ole kiinni ainoastaan yhden kehittäjän työpanoksesta, vaan kehittäjät osaavat toteuttaa kaikkia scrum-projektiin liittyviä työtehtäviä ja pystyvät tarvittaessa jatkamaan toistensa työtä. + Scrum-projekteihin voidaan valita myös kokemattomampia kehittäjiä, sillä kehittäjät voivat kehittää osaamistaan scrum-projektin aikana. - Koska kehitystiimit ovat pieniä, kehittäjiltä vaaditaan usein laajaa osaamista, mikä sumentaa kehittäjien rooleja.
Kehitystiimin itseohjautuvuus	<ul style="list-style-type: none"> + Kukaan ei johda kehitystiimin toimintaa, vaan kehitystiimi päättää itse työnjaosta ja työskentelytavoistaan. + Työskentely itseohjautuvassa kehitystiimissä lisää kehittäjien mahdollisuuksia vaikuttaa omiin työtehtäviinsä ja parantaa kehitystiimin toiminnan koordinoitua. + Työskentely itseohjautuvassa kehitystiimissä parantaa kehittäjien motivaatiota, sitoutumista scrum-projektiin, tehokkuutta ja tuottavuutta. + Kehitystiimi tarkastelee ja kehittää toimintaansa säännöllisesti, koko scrum-projektin ajan. + Useat scrumin tapahtumat ja tuotokset tukevat itseohjautuvan kehitystiimin toimintaa.
Kehitystiimin yhteistyö	<ul style="list-style-type: none"> + Vastuu tuotteen kehittämisestä kuuluu koko kehitystiimille ja kehittäjät tekevät paljon yhteistyötä tuotteen toteuttamisen parissa. + Kehittäjien tekemä yhteistyö parantaa kehittäjien välistä vuorovaikutusta ja tukee ryhmähengen muodostumista. + Keskittyminen kehittäjien yksilösuoritusten sijaan kehitystiimin yhteiseen suoritukseen vähentää kehittäjiin kohdistuvia paineita. - Jos kehittäjät haluavat työskennellä itsenäisesti tai heidän vuorovaikutustaitonsa ovat huonot, kehittäjien välinen vuorovaikutus ja yhteistyö voi olla vähäistä ja vaikeaa, mikä voi estää ryhmähengen muodostumisen.

Kehitystiimin työn läpinäkyvyys	<ul style="list-style-type: none"> + Kehittäjät ovat jatkuvasti tietoisia toistensa tekemästä työstä, sprintin etenemisestä ja etenemistä hidastavista tekijöistä. + Kehitystiimin työn ja sprinttien edistymisen jatkuva seuranta motivoi kehittäjiä aikaansaamaan jotain joka päivä. + Työn läpinäkyvyyden ansiosta scrum-projektin eteneminen ei ole kiinni ainoastaan yhden kehittäjän työpanoksesta, vaan kehittäjät pystyvät tarvittaessa jatkamaan toistensa työtä. + Kehittäjät voivat raportoida kohtaamistaan ongelmista ja pyytää apua muilta kehittäjiltä tai scrummasterilta nopeasti, mikä nopeuttaa virheiden korjaamista. - Työn läpinäkyvyyden vuoksi kehittäjien heikko osaaminen, huono työnjälki sekä sovitusta työtehtävistä ja aikatauluista poikkeaminen ilmenevät nopeasti.
Kehitystiimin työmäärä ja työskentelytahti	<ul style="list-style-type: none"> + Sprinttien ansiosta kehitystiimin työmäärä pysyy kohtuullisena koko scrum-projektin ajan, mikä takaa kehitystiimille tasaisen ja miellyttävän työskentelytahdin. + Työskentely sprinteissä ja useat scrumin tapahtumat ja tuotokset auttavat kehitystiimiä pysymään aikataulussa. - Scrum-projektin ulkopuoliset tekijät, kuten kehittäjien muut samanaikaiset projektit tai poissaolot, sekä asiakkaan kiireelliset vaatimukset voivat lisätä kehittäjien työmäärää ja aiheuttaa ylitöitä. - Kehitystiimin voi olla vaikea tehdä realistisia työmääräarvioita, jos asiakkaan tuotetta koskevat vaatimukset ovat epäselvät tai työtehtävistä ei ole aikaisempaa kokemusta. - Kehitystiimi voi sitoutua tekemään liian paljon työtä yhden sprintin aikana. - Scrumin tapahtumat voivat pitkittyä, jos tapahtumiin osallistuvat henkilöt poikkeavat käsiteltävistä aiheista tai tapahtumat eivät noudata scrumin viitekehystä. - Scrumin tapahtumille voi olla varattu liikaa tai liian vähän aikaa käsiteltäviin asioihin nähden. - Sopivan ajankohdan löytäminen scrumin tapahtumille voi olla vaikeaa.
Kehitystiimin tuottaman dokumentaation määrä	<ul style="list-style-type: none"> + Scrum-projekteissa tuotetun dokumentaation määrä on perinteisiin ohjelmistokehitysprojekteihin verrattuna vähäinen. + Dokumentaation sijaan kehitystiimin keskuudessa syntyy hiljaista tietoa kehitystiimin yhteistyön myötä. + Hiljaisen tiedon ansiosta oleellista tietoa ei voi kadota, jos kehittäjät vaihtuvat tai lähtevät kehitystiimistä kesken scrum-projektin. - Ilman dokumentaatiota uusien tai kokemattomampien kehittäjien perehdyttäminen voi olla vaikeaa.
Kehitystiimin yhteistyö tuoteomistajan kanssa	<ul style="list-style-type: none"> + Kehitystiimi tekee säännöllistä yhteistyötä tuoteomistajan kanssa ja saa tuoteomistajalta jatkuvasti palautetta tekemästään työstä. + Säännöllisen yhteistyön ansiosta kehitystiimi on varma siitä, että kehitettävä tuote vastaa asiakkaan vaatimuksia. + Säännöllisen yhteistyön ansiosta kehitystiimi voi kysyä nopeasti selvennystä tai neuvoa tuoteomistajalta. + Säännöllinen yhteistyö tuoteomistajan kanssa parantaa kehittäjien esiintymistaitoja ja ammattitaitoa. - Jos kehittäjien asiakkaan toimialaan ja liiketoimintaan liittyvä tieto tai vuorovaikutus- ja esiintymistaidot ovat huonot, yhteistyö tuoteomistajan kanssa voi olla vaikeaa.

Vaatimusten muuttuminen scrum-projektin aikana	<ul style="list-style-type: none">+ Tuoteomistaja voi muuttaa tuotetta koskevia vaatimuksia scrum-projektin edetessä.+ Kehitystiimi voi olla varma siitä, ettei tuotetta koskevat vaatimukset muutu kesken sprintin.+ Kehitystiimin on mahdollista vastata muuttuneisiin vaatimuksiin ja teknologia- ja liiketoimintaympäristön muutoksiin nopeasti, heti seuraavassa sprintissä.+ Muutostarpeiden nopean huomioimisen ansiosta kehitetty tuote ja asiakkaan tarpeet vastaavat paremmin toisiaan.- Jos tuoteomistaja on määritellyt tuotetta koskevat vaatimukset huonosti, kehittäjien täytyy opastaa tuoteomistajaa tuotetta koskeviin vaatimuksiin liittyen, kysyä vaatimuksiin liittyvää tarkennusta tai luoda esimerkkiratkaisuja, joiden avulla havainnollistetaan kehitettävää tuotetta tuoteomistajalle, jotta vältettäisiin korjausliikkeet ja ylimääräisen työn tekeminen.
---	--

Tutkielman Tulokset ja johtopäätökset -luvussa (luku 7) verrataan kirjallisuuskatsauksen ja empiirisen osuuden tutkimustuloksia toisiinsa ja esitetään tutkimustulosten pohjalta vastaus tutkielman tutkimuskysymykseen *mitä hyötyjä ja haasteita scrumiin liittyy kehitystiimin näkökulmasta?*

7 TULOKSET JA JOHTOPÄÄTÖKSET

Tässä luvussa verrataan tutkielman kirjallisuuskatsauksen ja empiirisen osuuden tutkimustuloksia toisiinsa ja esitetään tutkimustulosten pohjalta vastaus tutkielman tutkimuskysymykseen *mitä hyötyjä ja haasteita scrumiin liittyy kehitystiimin näkökulmasta?* Luvussa esitetään myös johtopäätöksiä siitä, miten tutkielmassa esille tulleet scrumin haasteet voidaan ratkaista ja tukea näin tutkielmassa esille tulleiden hyötyjen saavuttamista.

7.1 Tutkimustulosten vertailu

Tutkielman kirjallisuuskatsauksen tarkoitus oli muodostaa käsitys scrumin hyödyistä ja haasteista kehitystiimin näkökulmasta yhdistelemällä scrumia ja ketteriä menetelmiä käsittelevien tieteellisten julkaisujen tarjoamaa tutkimustietoa. Tutkielman empiirisen osuuden tarkoitus oli puolestaan tuottaa yksityiskohtaista, kirjallisuuskatsauksen tutkimustuloksia täsmentävää, scrumin hyötyihin ja haasteisiin kehitystiimin näkökulmasta liittyvää tietoa tutkimuskohteeksi valittua scrum-projektia tutkimalla. Kirjallisuuskatsauksessa ja empiirisessä osuudessa päädyttiin hyvin samankaltaisiin tutkimustuloksiin ja empiirisessä osuudessa tulikin esille hyvin paljon samoja scrumiin liittyviä hyötyjä ja haasteita kehitystiimin näkökulmasta, joita kirjallisuuskatsauksessa tuli esille. Joitakin eroja kirjallisuuskatsauksen ja empiirisen osuuden tutkimustulosten välillä kuitenkin esiintyi. Merkittävimmät erot kirjallisuuskatsauksen ja empiirisen osuuden tutkimustulosten välillä on kuvattu tässä luvussa.

Ensimmäinen aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *scrumin käyttöönotto*. Kirjallisuuskatsauksen ja empiirisen osuuden scrumin käyttöönottoa koskevien tutkimustulosten välillä oli yksi merkittävä ero. Kirjallisuuskatsauksessa korostettiin, että scrum oli yksinkertainen ja helposti opittava viitekehys. Empiirisessä osuudessa tuli kuitenkin esille, että oli yksilöllistä, kuinka yksinkertaisena ja helposti opittavana kehittäjät pitivät scrumin viitekehystä ja kehittäjien mielestä scrumin

viitekehitys saattoi olla jopa monimutkainen ja vaikeasti opittava. Empiirisen osuuden tutkimustulokset osoittavatkin, ettei voida tehdä yleistystä scrumin viitekehityksen yksinkertaisuudesta ja helposti opittavuudesta, niin kuin kirjallisuuskatsauksessa on tehty.

Toinen aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin koko*. Kirjallisuuskatsauksessa ja empiirisessä osuudessa päädyttiin hyvin samankaltaisiin kehitystiimin kokoa koskeviin tutkimustuloksiin. Merkittäviä eroja kirjallisuuskatsauksen ja empiirisen osuuden kehitystiimin kokoa koskevien tutkimustulosten välillä ei löytynyt.

Kolmas aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin työskentely-ympäristö*. Empiirisessä osuudessa tuli esille yksi merkittävä kehitystiimin työskentely-ympäristöön liittyvä haaste, jota ei tullut esille kirjallisuuskatsauksessa. Empiirinen osuus osoitti, ettei kehittäjien ollut aina mahdollista työskennellä samassa paikassa. Tätä haastetta ei otettu huomioon tutkielman kirjallisuuskatsauksessa, sillä eri paikassa työskentelevien kehittäjien muodostamat, niin sanotut hajaantuneet kehitystiimit rajattiin kirjallisuuskatsauksessa tutkimuksen ulkopuolelle.

Neljäs aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin monitaitoisuus*. Kirjallisuuskatsauksessa tuli esille, että jos kehittäjille ei tarjottu mahdollisuutta erikoistua tietyn työtehtävän asiantuntijoiksi, heidän asemansa urakehitys- ja rekrytointitilanteissa saattoi kärsiä. Empiirinen osuus kuitenkin osoitti, että kehittäjien laajasta osaamisesta oli kehittäjille hyötyä ja kehittäjät suhtautuivat positiivisesti osaamisensa kehittämiseen ja uusien työtehtävien opetteluun. Kirjallisuuskatsauksessa tuli myös esille, että scrum-projekteihin saattoi olla vaikea löytää kehittäjiä, joilla oli tarpeeksi laaja osaaminen ja että kokeneemmat kehittäjät saattoivat joutua neuvomaan kokemattomampia kehittäjiä, mikä vei aikaa varsinaiselta työnteolta. Empiirisessä osuudessa tuli kuitenkin esille, että scrum-projekteihin voitiin valita myös kokemattomampia kehittäjiä, sillä kehittäjien oli mahdollista kehittää osaamistaan scrum-projektin aikana ja että kokeneemmat kehittäjät suhtautuivat positiivisesti kokemattomempien kehittäjien neuvomiseen.

Viides aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin itseohjautuvuus*. Kirjallisuuskatsauksessa tuli esille, että jos scrumista ei ollut aikaisempaa kokemusta, scrummaster ja muut scrum-projektin sidosryhmät saattoivat vastustaa päätöksentekovalan antamista kehitystiimille ja kehitystiimin saattoi olla vaikea ottaa vastuu päätöksenteosta, jolloin kehitystiimin itseohjautuvuus ei toteutunut. Empiirisessä osuudessa tuli kuitenkin esille, että tutkimuskohteena olevassa scrum-projektissa ei oltu kohdattu tällaista haastetta. Tätä kirjallisuuskatsauksessa esille tullutta haastetta ei voida kuitenkaan täysin kumota empiirisen osuuden tutkimustulosten pohjalta, sillä tutkimuskohteena olevassa scrum-projektissa toimivilla kehittäjillä ja scrummasterilla oli sekä koulutuksen että käytännön tuomaa kokemusta scrumista, minkä ansiosta heidän oli todennäköisesti helppo omaksua omat roolinsa. Kaikissa scrum-projekteissa näin ei kuitenkaan tapah-

du. Empiirisessä osuudessa tuli esille myös kaksi kehitystiimin itseohjautuvuuteen liittyvää hyötyä, joita ei tullut esille kirjallisuuskatsauksessa. Ensimmäinen hyöty oli, että kehitystiimi tarkasteli ja kehitti toimintaansa säännöllisesti, koko scrum-projektin ajan. Toinen hyöty oli, että useat scrumin tapahtumat ja tuotokset tukivat itseohjautuvan kehitystiimin toimintaa.

Kuudes aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin yhteistyö*. Kirjallisuuskatsauksessa tuli esille, että kehittäjien oli mahdollista toteuttaa työtehtäviä joko yksin tai yhdessä muiden kehittäjien kanssa. Empiirisessä osuudessa tuli kuitenkin esille, että tutkimuskohteena olevassa scrum-projektissa toimivat kehittäjät toteuttivat työtehtäviä itsenäisesti. Tätä kirjallisuuskatsauksessa esille tullutta hyötyä ei kuitenkaan voida kumota empiirisen osuuden tutkimustulosten pohjalta, sillä on mahdollista, että muissa scrum-projekteissa kehittäjät toteuttavat työtehtäviä yhdessä muiden kehittäjien kanssa.

Seitsemäs aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin työn läpinäkyvyys*. Kirjallisuuskatsauksessa tuli esille, että kehittäjien oli mahdollista raportoida kohtaamistaan ongelmista ja pyytää apua muilta kehittäjiltä tai scrummasterilta nopeasti, mikä vähensi tuotteeseen liittyviä virheitä, helpotti ja nopeutti virheiden korjaamista ja auttoi kehitystiimiä pysymään aikataulussa. Empiirisessä osuudessa tuli kuitenkin esille, että kehittäjien mahdollisuus raportoida kohtaamistaan ongelmista ja pyytää apua nopeasti nopeutti tuotteeseen liittyvien virheiden korjaamista, mutta ei vähentänyt virheitä tai helpottanut virheiden korjaamista, sillä yksittäiset kehittäjät olivat vastuussa työtehtävien toteuttamisesta ja virheiden korjaamisesta eikä kehitystiimin työn läpinäkyvyys voinut vaikuttaa kehittäjien tekniseen osaamiseen. Kirjallisuuskatsauksessa tuli myös esille, että työn läpinäkyvyys saattoi lisätä kehittäjiin kohdistuvia paineita ja vaikuttaa negatiivisesti työskentelyilmapiiriin. Empiirisessä osuudessa tuli kuitenkin esille, että tutkimuskohteena olevassa scrum-projektissa ei oltu kohdattu tämänkaltaisia haasteita. Näitä kirjallisuuskatsauksessa esille tulleita haasteita ei kuitenkaan voida täysin kumota empiirisen osuuden tutkimustulosten pohjalta, sillä tutkimuskohteena olevassa scrum-projektissa toimivat kehittäjät olivat tunteet toisensa kauan, minkä ansiosta he uskalsivat antaa palautetta ja kritiikkiä toisilleen ja näkivät sen hyvänä tapana kehittää kehitystiimin toimintaa. Kaikissa scrum-projekteissa näin ei kuitenkaan tapahdu. Kirjallisuuskatsauksessa huomautettiin myös, että kehittäjät saattoivat tuntea scrumin tapahtumiin osallistumisen ja tekemänsä työn esittelyn muille kehittäjille epämiellyttäväksi, jos he toteuttivat hyvin spesifejä työtehtäviä. Empiirisessä osuudessa tuli kuitenkin esille, ettei kyseinen haaste voinut pitää paikkaansa, elleivät kehittäjät esitelleet tekemänsä työtä muille kehittäjille hyvin teknisellä tasolla, sillä kehittäjät suunnittelivat työtehtävät yhdessä sprintin suunnittelupalaverissa ja olivat näin tietoisia toistensa tekemästä työstä.

Kahdeksas aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin työmäärä ja työskentelytahti*. Kirjallisuuskatsauksessa korostettiin, että scrumin ansiosta ylitöiden määrä väheni

huomattavasti. Empiirisessä osuudessa tuli kuitenkin esille, että scrum-projektin ulkopuoliset tekijät, kuten kehittäjien muut samanaikaiset projektit ja poissaolot, sekä asiakkaan kiireelliset vaatimukset saattoivat lisätä kehittäjien työmäärää ja aiheuttaa ylitöitä. Empiirisessä osuudessa tuli myös esille yksi kehitystiimin työmäärään ja työskentelytahtiin liittyvä hyöty, jota ei tullut esille kirjallisuuskatsauksessa. Empiirinen osuus osoitti, että työskentely sprinteissä ja useat scrumin tapahtumat ja tuotokset auttoivat kehitystiimiä pysymään aikataulussa.

Yhdeksäs aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin tuottaman dokumentaation määrä*. Kirjallisuuskatsauksessa tuli esille, että scrum-projekteissa tuotetun dokumentaation määrä oli vähäinen. Empiirisessä osuudessa tuli kuitenkin esille, että kehitystiimin täytyi tuottaa tuotteen tekniseen toteutukseen liittyvää dokumentaatiota aivan kuten perinteisten ohjelmistokehitysmenetelmien mukaisissa ohjelmistokehitysprojekteissa, mutta muun dokumentaation määrä oli scrum-projekteissa perinteisiin ohjelmistokehitysprojekteihin verrattuna vähäinen. Kirjallisuuskatsauksessa tuli myös esille, että ilman dokumentaatiota oleellista tietoa saattoi kadota, jos kehittäjät vaihtuivat tai lähtivät kehitystiimistä kesken scrum-projektin. Empiirinen osuus kuitenkin osoitti, että kehitystiimin keskuudessa syntyvän hiljaisen tiedon ansiosta oleellista tietoa ei voinut kadota yksittäisen kehittäjän lähtiessä kehitystiimistä.

Kymmenes aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *kehitystiimin yhteistyö tuoteomistajan kanssa*. Empiirisessä osuudessa tuli esille yksi kehitystiimin yhteistyöhön tuoteomistajan kanssa liittyvä hyöty, jota ei tullut esille kirjallisuuskatsauksessa. Empiirinen osuus osoitti, että säännöllinen yhteistyö tuoteomistajan kanssa paransi kehittäjien esiintymistaitoja ja ammattitaitoa.

Viimeinen aihepiiri, johon liittyviä hyötyjä ja haasteita tutkielmassa tutkittiin kehitystiimin näkökulmasta oli *vaatimusten muuttuminen scrum-projektin aikana*. Kirjallisuuskatsauksessa tuli esille, että perinteisiä ohjelmistokehitysmenetelmiä käyttäneet kehittäjät saattoivat uskoa pystyvänsä selvittämään kehitettävää tuotetta koskevat vaatimukset jo scrum-projektin alussa ja suhtautua negatiivisesti vaatimusten muuttumiseen scrum-projektin aikana. Empiirisessä osuudessa tuli kuitenkin esille, että vaikka kehittäjät olisivat aikaisemmin käyttäneet perinteisiä ohjelmistokehitysmenetelmiä, he suhtautuivat positiivisesti vaatimusten muuttumiseen scrum-projektin aikana. Kirjallisuuskatsauksessa esille tullutta haastetta ei voida kuitenkaan täysin kumota empiirisen osuuden tutkimustulosten pohjalta, sillä tutkimuskohteena olevassa scrum-projektissa toimivilla kehittäjillä ja scrummasterilla oli monien vuosien työkokemus IT-alalta sekä koulutuksen ja käytännön tuomaa kokemusta scrumista, minkä ansiosta he todennäköisesti tiedostivat, kuinka yleistä vaatimusten muuttuminen ohjelmistokehitysprojekteissa oli ja kuinka kannattavaa muuttuneiden vaatimusten nopea huomioiminen oli. Kaikissa scrum-projekteissa näin ei kuitenkaan ole. Empiirisessä osuudessa tuli myös esille yksi haaste, jota ei tullut esille kirjallisuuskatsauksessa. Empiirinen osuus osoitti, että jos tuoteomistaja oli

määritellyt tuotetta koskevat vaatimukset huonosti, kehittäjien täytyi opastaa tuoteomistajaa tuotetta koskeviin vaatimuksiin liittyen, kysyä vaatimuksiin liittyvää tarkennusta tai luoda esimerkkiratkaisuja, joiden avulla voitiin havainnollistaa kehitettävää tuotetta tuoteomistajalle, jotta voitiin välttää korjausliikkeet ja ylimääräisen työn tekeminen.

7.2 Scrumin hyödyt ja haasteet kehitystiimin näkökulmasta

Tutkielman kirjallisuuskatsauksessa ja empiirisessä osuudessa tuli esille useita scrumiin liittyviä hyötyjä kehitystiimin näkökulmasta. Osa scrumiin liittyvistä hyödyistä kuitenkin kumottiin kirjallisuuskatsauksen ja empiirisen osuuden tutkimustulosten vertailun pohjalta. Merkittävimmät scrumiin liittyvät hyödyt kehitystiimin näkökulmasta on kuvattu seuraavassa luettelossa. Oikein noudatettuna scrum:

- lisää kehittäjien tyytyväisyyttä ohjelmistokehitystä kohtaan
- tukee kehittäjien välistä vuorovaikutusta
- tukee kehittäjien välistä tiedon jakamista
- parantaa kehittäjien osaamista
- tukee kehittäjien yhteistyötä
- tukee ryhmähengen muodostumista kehitystiimin keskuudessa
- lisää kehittäjien työnjakoon ja työskentelytapoihin liittyvää päätösvaltaa
- parantaa kehittäjien motivaatiota
- parantaa kehittäjien sitoutumista ohjelmistokehitysprojektiin
- parantaa kehitystiimin tehokkuutta ja tuottavuutta
- parantaa kehitystiimin tuloksia
- vähentää kehittäjiin kohdistuvia paineita
- parantaa kehittäjien tietoisuutta toistensa tekemästä työstä, scrum-projektin etenemisestä ja etenemistä hidastavista tekijöistä
- kasvattaa kehittäjien luottamusta toisiinsa ja scrum-projektin onnistumiseen
- parantaa kehitettävän tuotteen vastaavuutta asiakkaan sekä teknologia- ja liiketoimintaympäristön tarpeisiin ja vaatimuksiin
- parantaa kehitettävän tuotteen laatua
- nopeuttaa kehitettävään tuotteeseen liittyvien virheiden korjaamista
- tekee kehitettävään tuotteeseen liittyvien muutostarpeiden toteuttamisesta helppoa, nopeaa ja edullista
- kasvattaa kehittäjien tyytyväisyyttä kehitettävää tuotetta kohtaan
- pitää kehitystiimin työmäärän kohtuullisena koko scrum-projektin ajan
- tekee kehitystiimin työskentelytahdista tasaisen ja miellyttävän
- auttaa kehitystiimiä pysymään aikataulussa
- vähentää kehitystiimin tuottaman dokumentaation määrää
- tukee hiljaisen tiedon syntymistä kehitystiimin keskuudessa

Tutkielman kirjallisuuskatsauksessa ja empiirisessä osuudessa tuli esille myös useita scrumiin liittyviä haasteita kehitystiimin näkökulmasta. Myös osa scrumiin liittyvistä haasteista kumottiin kirjallisuuskatsauksen ja empiirisen osuuden tutkimustulosten vertailun pohjalta. Scrumiin liittyvien haasteiden ratkaiseminen on tärkeää, jotta scrumiin liittyvät hyödyt voidaan saavuttaa. Merkittävimmät scrumiin liittyvät haasteet kehitystiimin näkökulmasta on kuvattu seuraavassa luettelossa:

- kehitystiimin muutosvastarinta scrumin käyttöönottoa kohtaan
- scrumin viitekehyksen muuttaminen scrumin käyttöönoton yhteydessä
- kehitystiimin liian pieni tai liian suuri koko
- kehittäjien muut samanaikaiset projektit
- kehittäjien työskentely eri paikoissa
- avoimessa työtilassa työskentelyyn liittyvät häiriötekijät
- kehittäjien roolien sumentuminen
- työnjakoon ja työskentelytapoihin liittyvän päätösvallan pysyminen scrummasterilla tai muilla scrum-projektin sidosryhmillä
- kehittäjien halu työskennellä itsenäisesti
- kehittäjien huonot vuorovaikutus- ja esiintymistaidot
- kehittäjien heikon osaamisen, huonon työnjäljen sekä sovitusta aikatauluista ja työtehtävistä poikkeamisen nopea ilmeneminen ja kehittäjiin kohdistuvien paineiden lisääntyminen
- scrumin tapahtumien pitkittyminen
- sopivan pituuden ja ajankohdan löytäminen scrumin tapahtumille
- dokumentaation vähäisyys uusien tai kokemattomampien kehittäjien perehdyttämisen yhteydessä
- kehittäjien huono asiakkaan toimialaan ja liiketoimintaan liittyvä tieto
- tuoteomistajan kiireisyys ja sitoutumattomuus scrum-projektiin
- kehittäjien negatiivinen suhtautuminen vaatimusten muuttumiseen scrum-projektin aikana
- kehitettävää tuotetta koskevien vaatimusten huono määrittely

7.3 Scrumin haasteiden ratkaiseminen ja hyötyjen saavuttaminen

Tämän tutkielman tarkoitus oli muodostaa käsitys scrumin hyödyistä ja haasteista kehitystiimin näkökulmasta. Scrumin haasteiden tiedostaminen ja ratkaiseminen on tärkeää, jotta scrumin hyödyt ja hyötyjen tuoma lisäarvo voidaan saavuttaa. Scrumia ja ketteriä menetelmiä käsittelevissä tieteellisissä julkaisuissa ja tutkielman empiirisessä osuudessa tulee esille useita keinoja ratkaista scrumiin liittyviä haasteita ja tukea näin scrumiin liittyvien hyötyjen saavuttamista.

Ensimmäinen aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja empiirisessä osuudessa on esitetty keinoja on *scrumin käyttöönotto*.

Schatz ja Abdelshafi (2005) korostavat, että scrumtiimin täytyy tiedostaa, että scrumin käyttöönotto vaatii paljon työtä ja aikaa ja huomauttavat, että yrityksen täytyy kannustaa scrumtiimiä scrumin käyttöönotossa, sillä isojen muutoksien yhteydessä ihmiset etsivät usein syytä palata takaisin vanhoihin ja tuttuihin käytäntöihin. Conboy ym. (2011) ja Reifer (2002) korostavat, että on tärkeää, että yritys tukee scrumin käyttöönottoa tarjoamalla scrumtiimille scrumiin liittyvää koulutusta ja tuomalla esille scrumiin liittyviä positiivisia kokemuksia. Myös tutkielman empiirisessä osuudessa tulee esille, että scrumtiimille tarjottavan scrumiin liittyvän koulutuksen avulla voidaan tukea scrumin käyttöönottoa. Conboy ym. (2011) tuovat esille, että henkilöt, joilla on aikaisempaa kokemusta scrumista, voivat osallistua scrum-projekteihin ja ohjata muita scrumtiimin jäseniä scrumin käyttöönoton yhteydessä. Schatz ja Abdelshafi (2005) puolestaan tuovat esille, että scrumtiimi voi hankkia myös scrumtiimin ulkopuolisen henkilön tarkastelemaan ja ohjaamaan scrumin käyttöönottoa, sillä ulkopuolisen henkilön on helpompi antaa rehellistä ja puolueetonta palautetta scrumtiimille. Schatz ja Abdelshafi (2005) korostavat myös, ettei scrumtiimin pitäisi muuttaa scrumin viitekehystä tai scrumiin liittyviä käsitteitä scrumin käyttöönoton yhteydessä, sillä viitekehysten ja käsitteiden täsmällinen noudattaminen kannustaa scrumtiimiä ajattelemaan täysin uudella tavalla, jolloin scrumtiimi todennäköisemmin unohtaa ja on valmis luopumaan aikaisemmin käytetyistä ohjelmistokehitysmenetelmistä. Conboy ym. (2011) huomauttavat, että yksi tapa tukea scrumin viitekehysten noudattamista on vaihdella kehittäjiä eri kehitystiimien välillä, jolloin kehittäjien on mahdollista tarkastella, noudatetaanko scrumin viitekehystä samalla tavalla eri scrum-projekteissa ja tuoda esille, miten scrumtiimi voisi parantaa toimintaansa.

Toinen aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja empiirisessä osuudessa on esitetty keinoja on *kehitystiimin koko*. Schwaber ja Sutherland (2011) korostavat, että kehitystiimi on optimaalisen kokoinen silloin, kun siihen kuuluu 3-9 kehittäjää. Ambler (2002) tuo esille, että jos kehitettävä tuote edellyttää paljon työtä ja suuremman kehitystiimin, kehitystiimit voidaan muodostaa siten, että kehitettävän tuotteen parissa työskentelee useita pieniä rinnakkaisia kehitystiimejä. Moe ym. (2009) korostavat, että on tärkeää, että kehittäjät toimivat vain yhdessä projektissa kerrallaan. Tutkielman empiirisessä osuudessa tulee kuitenkin esille, että jos kehittäjät joutuvat scrum-projektin lisäksi osallistumaan myös muihin projekteihin, heidän kannattaa sovitaa aikataulunsa niin, että he pystyvät tekemään scrum-projektiin liittyvää työtä samaan aikaan sekä osallistua scrum-projektin tapahtumiin ja seurata sprintin tehtävälistaa, jolloin he ovat jatkuvasti tietoisia muiden kehittäjien tekemästä työstä ja scrum-projektin etenemisestä.

Kolmas aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *kehitystiimin työskentely-ympäristö*. Moe ym. (2009) sekä Schatz ja Abdelshafi (2005) tuovat esille, että kehitystiimi kannattaa sijoittaa samaan työtilaan. Cho (2008) korostaa, että työtilan, jossa kehitystiimi työskentelee, pitäisi olla mahdollisimman avoin, jolloin kehittäjät eivät ole eristyksissä toisistaan, vaan he näkevät ja kuulevat

toisensa koko ajan. Moe ym. (2009) kuitenkin huomauttavat, että samassa työtilassa työskentelevien kehittäjien pitäisi tarvittaessa antaa työskennellä rauhassa, ilman keskeytyksiä. Cohn ja Ford (2003) tuovat esille, että jos kehitystiimin ei ole mahdollista työskennellä samassa työtilassa, kehittäjille pitäisi tarjota mahdollisuus työskennellä samassa paikassa ainakin muutama scrum-projektin ensimmäinen viikko. Tutkielman empiirisessä osuudessa puolestaan tulee esille, että eri paikassa työskentelevien kehittäjien välistä vuorovaikutusta ja yhteistyötä voidaan tukea nykyaikaisen viestintäteknologian avulla.

Neljäs aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *kehitystiimin monitaitoisuus*. Moe ym. (2009) tuovat esille, että yksi tapa tukea kehitystiimin monitaitoisuutta on arvostaa kehittäjien laajaa osaamista ja rekrytoida yritykseen ihmisiä, joilla on potentiaalia ja halua kehittää osaamistaan. Conboy ym. (2011) huomauttavat, että kehittäjille täytyy tarjota mahdollisuus kehittää osaamistaan, mutta kehittäjien pitäisi myös pyrkiä erikoistumaan tietyn työtehtävän ammatillaisiksi, jotta heidän asemansa urakehitys- ja rekrytointitilanteissa ei kärsisi. Tutkielman empiirisessä osuudessa tulee esille, että kehitystiimeihin voidaan valita kokemattomampia kehittäjiä, sillä kehittäjät voivat kehittää osaamistaan scrum-projektin aikana. Cohn ja Ford (2003) huomauttavat kuitenkin, että scrum-projekteihin ei pitäisi valita liikaa kokemattomia kehittäjiä, sillä he saattavat hidastaa koko kehitystiimin toimintaa tai jäädä jälkeen muista kehittäjistä. Ambler (2002) puolestaan korostaa, että scrum-projekteihin pitäisi valita muutamia kokeneita kehittäjiä, joilla on scrum-projektin edellyttämä osaaminen ja jotka osaavat ja haluavat neuvoa muita kehittäjiä, jolloin scrum-projekteihin voidaan huoletta valita myös kokemattomampia kehittäjiä ja luottaa siihen, että heidän osaamisensa paranee scrum-projektin aikana.

Viides aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *kehitystiimin itseohjautuvuus*. Hoda (2013) huomauttaa, että on tyypillistä, että nykyaikaiset ohjelmistokehitystiimit haluavat olla itseohjautuvia. Hoda (2013) korostaakin, että jos yritykseen on rekrytoitu osaavia kehittäjiä, heidän osaamistaan ei kannata aliarvioida, vaan heille pitäisi antaa vastuu työnjakoon ja työskentelytapoihinsa liittyvästä päätöksenteosta. Berczuk ja Lv (2010) tuovat esille, että kehitystiimin itseohjautuvuutta voidaan tukea tarjoamalla kehittäjille ja scrummastereille scrumiin liittyvää koulutusta, jonka avulla voidaan tukea uusien roolien omaksumista. Myös tutkielman empiirinen osuus osoittaa, että scrumiin liittyvän koulutuksen avulla voidaan tukea scrumin roolien omaksumista. Berczuk ja Lv (2010) tuovat esille, että toinen tapa tukea kehitystiimien itseohjautuvuutta on tarjota kehittäjille mahdollisuus tutustua toisiinsa ennen scrum-projektin alkamista ja vaikuttaa kehitystiimin muodostamiseen, eli kehittäjien valintaan.

Kuudes aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *kehitystiimin yhteistyö*. Tutkielman empiirisessä osuudessa tulee esille, että kehitystiimin yhteistyötä voidaan tukea rekrytoimalla yritykseen ja valitsemalla scrum-projekteihin ulospäinsuuntautuneita ja sosiaalisia henkilöitä. Conboy ym. (2011)

tuovat esille, että kehitystiimin toimintaa pitäisi mitata ja palkita kehitystiimin yhteisen suorituksen perusteella, mikä kannustaa kehittäjiä tekemään yhteistyötä. Berczuk ja Lv (2010) puolestaan korostavat, että scrum-projekteissa pitäisi pyrkiä poistamaan kehitystiimin yhteistyötä ja ryhmähengen muodostumista estävät tekijät.

Seitsemäs aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *kehitystiimin työn läpinäkyvyys*. Conboy ym. (2011) korostavat, että kehittäjien pitäisi tuntea olonsa turvalliseksi uskaltaakseen tuoda esille kohtaamiaan ongelmia ja työntekoa hidastavia tekijöitä. Tutkielman empiirisessä osuudessa puolestaan tulee esille, että kehittäjien pitäisi olla kannustavia toisiaan kohtaan, jolloin he uskaltaavat tuoda esille kohtaamiaan ongelmia ja työntekoa hidastavia tekijöitä, eivätkä koe saamaansa palautetta negatiiviseksi, vaan näkevät palautteen keinona parantaa kehitystiimin toimintaa. Cohn ja Ford (2003) tuovat esille, että tuoteomistajan ei pitäisi kritisoida tai tuomita kehittäjiä sovitusta aikatauluista tai työtehtävistä poikkeamisesta, vaan pitäisi ainoastaan pyrkiä poistamaan kehittäjien työntekoa hidastavat tekijät, jolloin kehitystiimin työn läpinäkyvyys ei kasvata kehittäjiin kohdistuvia paineita.

Kahdeksas aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *kehitystiimin työmäärä ja työskentelytahti*. Schatz ja Abdelshafi (2005) tuovat esille, että ohjelmistokehittäjillä on usein tapana tehdä ylitöitä riippumatta siitä, mitä ohjelmistokehitysmenetelmää he käyttävät, joten scrum-projekteissa toimivilta kehittäjiltä pitäisi kieltää ylitöiden tekeminen, sillä scrum mahdollistaa projektin toteuttamisen tekemättä ylitöitä. Tutkielman empiirisessä osuudessa puolestaan tulee esille, että kaiken kehittäjien tekemän työn pitäisi olla työtehtävänä sprintin tehtävälillä, jolloin voidaan varmistaa, että kehittäjät työskentelevät vain scrum-projektiin liittyvien työtehtävien parissa.

Yhdeksäs aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *kehitystiimin tuottaman dokumentaation määrä*. Selic (2009) tuo esille, että ohjelmistokehitysprojekteissa dokumentaation tarkoitus on kertoa niille henkilöille, jotka eivät tunne kehitettävää tuotetta, kuinka tuote on kehitetty ja kuinka tuote toimii. Selic (2009) korostaakin, että tuotteen tekniseen toteutukseen liittyvän dokumentaation tuottaminen on tärkeää jokaisessa ohjelmistokehitysprojektissa. Selic (2009) kuitenkin huomauttaa, että ohjelmistokehitystiimin ei kannata tuottaa dokumentaatiota, jota jouduttaisiin päivittämään ohjelmistokehitysprojektin aikana, sillä tämänkaltaisen dokumentaation tuottaminen ja ylläpitäminen vie turhaan kehittäjien aikaa. Maiden ja Jones (2010) korostavat, ettei kehitystiimin tuottaman dokumentaation pitäisi olla liian yksityiskohtaista, sillä yksityiskohtainen dokumentaatio vähentää scrumtiimin vuorovaikutusta ja scrum-projektiin liittyvän tiedon jakamista. Maiden ja Jones (2010) korostavatkin, että dokumentaation pitäisi olla pääpiirteittäistä ja helposti saatavilla, esimerkiksi elektronisessa muodossa, mikä tukee scrumtiimin vuorovaikutusta ja scrum-projektiin liittyvän tiedon jakamista. Tutkielman empiirisessä osuudessa tulee

esille, että kehittäjien kannattaa pitää tuotettu dokumentaatio lyhyenä, selkeänä ja rakenteeltaan loogisena, jolloin se riittää kuvaamaan asiakkaalle ja uusille kehittäjille, kuinka tuote on kehitetty ja kuinka tuote toimii, mutta sen tuottaminen ei vie kohtuuttomasti kehittäjien aikaa.

Kymmenes aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *kehitystiimin yhteistyö tuoteomistajan kanssa*. Conboy ym. (2011) korostavat, että kehitystiimin ja tuoteomistajan yhteistyötä voidaan tukea tarjoamalla kehittäjille vuorovaikutus- ja esiintymistaitoihin sekä asiakkaan toimialaan ja liiketoimintaan liittyvää koulutusta. Conboy ym. (2011) tuovat myös esille, että jos kehitystiimin yhteistyö tuoteomistajan kanssa on vaikeaa, kehittäjiä voidaan kannustaa käyttämään esimerkiksi dokumentaatiota tai esitysmateriaalia tukemaan vuorovaikutusta tuoteomistajan kanssa. Tutkielman empiirisessä osuudessa tulee esille, että mitä enemmän kehittäjät tekevät yhteistyötä tuoteomistajan kanssa, sitä enemmän heidän esiintymistaitonsa kehittyvät.

Viimeinen aihepiiri, johon liittyvien haasteiden ratkaisemiseksi kirjallisuudessa ja tutkielman empiirisessä osuudessa on esitetty keinoja on *vaatimusten muuttuminen scrum-projektin aikana*. Tutkielman empiirinen osuus osoittaa, että mitä enemmän koulutuksen ja käytännön tuomaa kokemusta kehittäjillä on ohjelmistokehityksestä ja scrumista, sitä paremmin kehittäjät suhtautuvat vaatimusten muuttumiseen scrum-projektin aikana. Orr (2004) tuo esille, että asiakas ja loppukäyttäjät eivät välttämättä tiedä, mitä he haluavat kehitettävältä tuotteelta ja mitä he voivat saada, joten kehittäjien tehtävä on auttaa asiakasta määrittelemään, mitä he todella tarvitsevat, sen sijaan, että he vain kirjoittaisivat asiakkaan vaatimukset ylös. Orr (2004) suosittelee myös prototyyppien käyttämistä, joiden avulla voidaan havainnollistaa asiakkaalle, miten kehitettävä tuote voisi toimia ja auttaa näin asiakasta vaatimusten määrittelyssä. Tutkielman empiirisessä osuudessa puolestaan tulee esille, että jos tuoteomistaja on määrittellyt tuotetta koskevat vaatimukset huonosti, kehittäjien pitäisi opastaa tuoteomistajaa tuotetta koskeviin vaatimuksiin liittyen, kysyä vaatimuksiin liittyvää tarkennusta tai luoda esimerkkiratkaisuja, joiden avulla voidaan havainnollistaa kehitettävää tuotetta tuoteomistajalle, jotta vältettäisiin korjausliikkeet ja ylimääräisen työn tekeminen.

7.4 Yhteenveto

Tämän luvun tarkoitus oli verrata tutkielman kirjallisuuskatsauksen ja empiirisen osuuden tutkimustuloksia toisiinsa ja esittää kirjallisuuskatsauksen ja empiirisen osuuden tutkimustulosten pohjalta tiivistetty vastaus tutkielman tutkimuskysymykseen *mitä hyötyjä ja haasteita scrumiin liittyy kehitystiimin näkökulmasta?* Luvussa esitettiin myös johtopäätöksiä siitä, miten tutkielmassa esille tulleet scrumin haasteet voidaan ratkaista ja tukea näin tutkielmassa esille tulleiden hyötyjen saavuttamista.

8 YHTEENVETO

Tämän pro gradu -tutkielman tarkoitus oli luoda käsitys scrumin hyödyistä ja haasteista kehitystiimin näkökulmasta. Scrumin hyötyihin ja haasteisiin liittyvää tietoa hankittiin kirjallisuuskatsauksen ja eräässä IT-alan palveluyrityksessä toteutetun tapaustutkimuksen avulla. Scrumin hyötyihin ja haasteisiin kehitystiimin näkökulmasta liittyvä kokonaisvaltainen tutkimus voidaan nähdä tarpeelliseksi ja ajankohtaiseksi, sillä monet aikaisemmat aihetta käsittelevät tutkimukset ovat hyvin rajattuja ja koskevat vain tiettyyn scrumiin liittyvään aihepiiriin, kuten scrumin käyttöönottoon tai kehitystiimin itseohjautuvuuteen liittyviä hyötyjä ja haasteita.

On tärkeää tunnistaa ohjelmistokehitysprojektin luonne ja valita ohjelmistokehitysprojektin toteuttamiseen parhaiten soveltuva ohjelmistokehitysmenetelmä, sillä valittu ohjelmistokehitysmenetelmä vaikuttaa merkittävästi ohjelmistokehitysprojektin onnistumiseen. Scrum on ohjelmistokehityksen viitekehys, joka on kehitetty helpottamaan erityisesti monimutkaisten ohjelmistokehitysprojektien hallintaa ja toteuttamista. Scrumin viitekehukseen tutustuminen on tärkeää, jotta ymmärrettäisiin, minkälaisen ohjelmistokehitysprojektien toteuttamiseen scrum soveltuu ja mitä scrum-projekteihin osallistuvilta henkilöiltä edellytetään.

Scrumin hyödyillä tarkoitetaan lisäarvoa, jonka scrum tuo ohjelmistokehitysprojekteihin. Scrumin haasteilla puolestaan tarkoitetaan scrum-projekteihin liittyviä ongelmia. Scrumin hyötyihin tutustuminen on tärkeää, sillä hyödyt tiedostamalla yritykset voivat arvioida, kannattaako käytössä olevista ohjelmistokehitysmenetelmistä siirtyä scrumiin ja tarkastella, toteutuvatko scrumin hyödyt yrityksen omissa scrum-projekteissa. Myös scrumin haasteisiin tutustuminen on tärkeää, sillä scrumin haasteet tiedostamalla yritykset voivat ennustaa ja varautua scrumin käyttöönoton yhteydessä mahdollisesti ilmeneviin haasteisiin sekä tutkia, liittyykö yrityksen omiin scrum-projekteihin haasteita, jotka ovat mahdollisesti jääneet huomaamatta. Scrumin haasteiden tiedostaminen ja ratkaiseminen on tärkeää, jotta scrumin hyödyt ja hyötyjen tuoma lisäarvo voidaan saavuttaa.

Tutkielman yhteydessä toteutettu kirjallisuuskatsaus ja tapaustutkimus osoittivat, että scrumiin liittyy useita eri hyötyjä ja haasteita kehitystiimin näkökulmasta. Kirjallisuuskatsauksessa ja tapaustutkimuksessa esille tulleet scrumin hyödyt ja haasteet on esitetty tiivistetyssä muodossa tutkielman Scrumin hyödyt ja haasteet kehitystiimin näkökulmasta -luvussa (luku 7.2). Tutkielmassa tuotiin esille myös kirjallisuuskatsauksessa ja tapaustutkimuksessa esille tulleita keinoja ratkaista tutkielmassa esille tulleita scrumiin liittyviä haasteita ja tukea näin tutkielmassa esille tulleiden scrumiin liittyvien hyötyjen saavuttamista. Nämä keinot on esitetty tutkielman Scrumin haasteiden ratkaiseminen ja hyötyjen saavuttaminen -luvussa (luku 7.3).

Mielenkiintoinen jatkotutkimusaihe olisi, mitä hyötyjä ja haasteita scrumin viitekehystä mukaileviin ohjelmistokehitysprojekteihin liittyy ja onko yrityksissä, joissa scrumin viitekehystä on käytetty vain osittain, jatkettu scrumin viitekehysten mukailemista, siirrytty noudattamaan scrumin viitekehystä vai siirrytty käyttämään muita ohjelmistokehitysmenetelmiä. Toinen mielenkiintoinen jatkotutkimusaihe olisi, mitä hyötyjä ja haasteita liittyy scrum-projekteihin, joissa kehitystiimit ovat hajaantuneita, eli joissa toimivat kehittäjät työskentelevät eri paikoissa.

LÄHTEET

- Abbas, N., Gravell, A. M. & Wills, G. B. (2008). Historical Roots of Agile Methods: Where did Agile Thinking Come from? Teoksessa Abrahamsson, P., Baskerville, R., Conboy, K., Fitzgerald, B., Morgan, L. & Wang, X. (toim.), *Proceedings of the 9th International Conference on Agile Processes in Software Engineering and Extreme Programming, Limerick, Ireland, Jun 10-14* (s. 94-103). Berlin: Springer-Verlag.
- Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J. (2002). *Agile Software Development Methods: Review and Analysis*. Espoo: VTT.
- Agile Alliance. (2001). *Manifesto for Agile Software Development*. Haettu 1.9.2012 osoitteesta <http://agilemanifesto.org/>
- Ambler, S. (2002). Agile Development Best Dealt with in Small Groups. *Computing Canada*, 28(9), 9.
- Ambler, S. (2005). Quality in an Agile World. *Software Quality Professional*, 7(4), 34-40.
- Berczuk, S. & Lv, Y. (2010). We're All in This Together. *IEEE Software*, 27(6), 12-15.
- Boehm, B. (2002). Get Ready for Agile Methods, with Care. *Computer*, 35(1), 64-69.
- Bradbury, D. (2007). Scrum Down to Get a Project Moving. *Computer Weekly*, 30-31.
- Cao, L. & Ramesh, B. (2008). Agile Requirements Engineering Practices: An Empirical Study. *IEEE Software*, 25(1), 60-67.
- Ceschi, M., Sillitti, A., Succi, G. & Panfilis, S. (2005). Project Management in Plan-Based and Agile Companies. *IEEE Software*, 22(3), 21-27.
- Cho, J. (2008). Issues and Challenges of Agile Software Development with Scrum. *Issues in Information Systems*, 9(2), 188-195.
- Cohn, M. & Ford, D. (2003). Introducing an Agile Process to an Organization. *Computer*, 36(6), 1-10.
- Conboy, K., Coyle, S., Wang, X. & Pikkarainen, M. (2011). People over Process: Key Challenges in Agile Development. *IEEE Software*, 28(4), 48-57.
- Coram, M. & Bohner, S. (2005). The Impact of Agile Methods on Software Project Management. Teoksessa O'Neilt, T., Peng, J. & Rozenblit, J. (toim.), *Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems, Greenbelt, Maryland, April 4-7*. Los Alamitos, California: IEEE Computer Society.
- Dybå, T. & Dingsøy, T. (2008). Empirical Studies of Agile Software Development: A Systematic Review. *Information and Software Technology*, 50(9-10), 833-859.
- Eriksson, P. & Koistinen, K. (2005). *Monenlainen tapaustutkimus*. Haettu 1.1.2013 osoitteesta

http://www.ncrc.fi/files/4957/2005_04_verkkojulkaisu_tapaustutkimus.pdf

- Eskelinen, A., Heiramo, P., Heljo, S., Koskela, L., Lekman, L., Lindström, J., Nyman, R., Ruuskanen, S., Taipale, M., Tarvainen, A., Tikka, A., Virtanen, P. & Ziegler, L. (2012). *Suomenkielinen scrum-sanasto*. Haettu 1.1.2013 osoitteesta <http://scrumwell.files.wordpress.com/2012/01/suomenkielinen-scrum-sanasto-2012-v1-2.pdf>
- Felker, C., Slamova, R. & Davis, J. (2012). Integrating UX with Scrum in an Undergraduate Software Development Project. Teoksessa *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education, Raleigh, North Carolina, USA, February 29-March 3* (s. 301-306). New York, USA: ACM.
- Fernandez, D. J. & Fernandez, J. D. (2008). Agile Project Management: Agilism Versus Traditional Approaches. *The Journal of Computer Information Systems*, 49(2), 10-17.
- Figueiredo, A. M. (2009). An Executive Scrum Team. Teoksessa Dubinsky, Y., Dybå, T., Adolph, S. & Sidky, A. (toim.), *Proceedings of Agile 2009 Conference, Chicago, USA, August 24-28* (s. 147-170). Los Alamitos, California: IEEE Computer Society.
- Highsmith, J. & Cockburn, A. (2001). Agile Software Development: The Business of Innovation. *Computer*, 34(9), 120-122.
- Hirsjärvi, S. & Hurme, H. (2011). *Tutkimushaastattelu: Teemahaastattelun teoria ja käytäntö*. Helsinki: Yliopistopaino.
- Hirsjärvi, S., Remes, P. & Sajavaara, P. (2007). *Tutki ja kirjoita*. Helsinki: Tammi.
- Hoda, R. (2013). Power to the People. *IEEE Software*, 30(2), 92-93.
- Hämäläinen, J. (2012). *Scrumin hyödyt ja haasteet tuotteen omistajan ja kehitystiimin näkökulmasta*. Tietojärjestelmätieteen kandidaatintutkielma. Jyväskylän yliopisto.
- Järvinen, P. & Järvinen, A. (2011). *Tutkimustyön metodeista*. Tampere: Opinpajan kirja.
- Largent, D. L. (2010). Getting and Staying Agile. *XRDS: Crossroads, The ACM Magazine for Students*, 17(1), 38-41.
- Li, J., Moe, N. B. & Dybå, T. (2010). Transition from a Plan-Driven Process to Scrum: A Longitudinal Case Study on Software Quality. Teoksessa *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, Bolzano, Italy, September 16-17* (s. 117-126). New York, USA: ACM.
- Maiden, N. & Jones, S. (2010). Agile Requirements: Can We Have Our Cake and Eat it Too? *IEEE Software*, 27(3), 87-88.
- Mann, C. & Maurer, F. (2005). A Case Study on the Impact of Scrum on Overtime and Customer Satisfaction. Teoksessa Kawada, S. (toim.), *Proceedings of Agile 2005 Conference, Denver, Colorado, July 24-29* (s. 70-79). Los Alamitos, California: IEEE Computer Society.
- Marchenko, A. & Abrahamsson, P. (2008). Scrum in a Multiproject Environment: An Ethnographically-Inspired Case Study on the Adoption Challenges. Teoksessa Melnik, G., Kruchten, P. & Poppendieck, M. (toim.), *Proceedings*

- of Agile 2008 Conference, Toronto, Canada, August 4-8 (s. 15-26). Los Alamitos, California: IEEE Computer Society.
- Mellor, S. J. (2005). Adapting Agile Approaches to Your Project Needs. *IEEE Software*, 22(3), 17-20.
- Metsämuuronen, J. (2006). *Laadullisen tutkimuksen käsikirja*. Jyväskylä: Gummerus.
- Moe, N. B., Dingsøyr, T. & Dyba, T. (2009). Overcoming Barriers to Self-Management in Software Teams. *IEEE Software*, 26(6), 20-26.
- Monteiro, C. V. F., da Silva, F. Q. B., dos Santos, I. R. M., Farias, F., Cardozo, E. S. F., do A. Leitão, A. R. G., Neto, D. N. M. & Pernambuco Filho, M. J. A. (2011). A Qualitative Study of the Determinants of Self-Managing Team Effectiveness in a Scrum Team. Teoksessa *Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering, Waikiki, Honolulu, USA, May 21* (s. 16-23). New York, USA: ACM.
- Orr, K. (2004). Agile Requirements: Opportunity or Oxymoron? *IEEE Software*, 21(3), 71-73.
- Overhage, S., Schlauderer, S., Birkmeier, D. & Miller, J. (2011). What Makes IT Personnel Adopt Scrum? A Framework of Drivers and Inhibitors to Developer Acceptance. Teoksessa Sprague, R. H. (toim.), *Proceedings of the 44th Annual Hawaii International Conference on System Sciences, Koloa, Kauai, Hawaii, January 4-7* (s. 1-10). Los Alamitos, California: IEEE Computer Society.
- Pikkarainen, M. & Wang, X. (2011). An Investigation of Agility Issues in Scrum Teams Using Agility Indicators. Teoksessa Linger, H., Song, W. W. W., Wan, C., Wojtkowski, G., Wojtkowski, W., Xu, S. & Zhong, Y. (toim.), *Proceedings of Information Systems Development, Asian Experiences* (s. 449-459). New York: Springer.
- Pries-Heje, L. & Pries-Heje, J. (2011a). Agile and Distributed Project Management: A Case Study Revealing Why Scrum Is Useful. Teoksessa Tuunainen, V. K., Rossi, M. & Nandhakumar, J. (toim.), *Proceedings of the 19th European Conference on Information Systems, Helsinki, Finland, June 9-11* (s. 217). AIS Electronic Library.
- Pries-Heje, L. & Pries-Heje, J. (2011b). Why Scrum Works: A Case Study from an Agile Distributed Project in Denmark and India. Teoksessa O'Conner, L. (toim.), *Proceedings of 2011 Agile Conference, Salt Lake City, Utah, August 8-12* (s. 20-28). Los Alamitos, California: IEEE Computer Society.
- Rally Software Development Corporation, Schwaber, K. & Scrum Alliance. (2005). *A CIO's Playbook for Adopting the Scrum Method of Achieving Software Agility*. Haettu 1.9.2012 osoitteesta http://www.rallydev.com/documents/CIO_Playbook_For_Adopting_Scrum_080805.pdf
- Rehman, A. & Hussain, R. (2007). Software Project Management Methodologies/Frameworks Dynamics: A Comparative Approach. Teoksessa *Proceedings of 2007 International Conference on Information and Emerging Technologies, Karachi, Pakistan, July 6-7* (s. 1-5). Los Alamitos, California: IEEE Computer Society.

- Reifer, D. J. (2002). How Good Are Agile Methods? *IEEE Software*, 19(4), 16-18.
- Rising, L. & Janof, N. S. (2000). The Scrum Software Development Process for Small Teams. *IEEE Software*, 17(4), 26-32.
- Schatz, B. & Abdelshafi, I. (2005). Primavera Gets Agile: A Successful Transition to Agile Development. *IEEE Software*, 22(3), 36-42.
- Schwaber, K. & Sutherland, J. (2011). *The Scrum Guide*. Haettu 1.9.2012 osoitteesta <http://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum%20Guide%20-%20FI.pdf>
- Selic, B. (2009). Agile Documentation, Anyone? *IEEE Software*, 26(6), 11-12.
- Sutherland, J. (2010). *Scrum Handbook*. Haettu 1.9.2012 osoitteesta <http://jeffsutherland.com/scrumhandbook.pdf>
- Wang, X., Lane, M., Conboy, K. & Pikkarainen, M. (2009). Where Agile Research Goes: Starting from a 7-Year Retrospective. *ACM SIGSOFT Software Engineering Notes*, 34 (5), 28-30.
- Williams, L. & Cockburn, A. (2003). Agile Software Development: It's about Feedback and Change. *Computer*, 36(6), 39-43.
- Zhi-gen, H., Quan, Y. & Xi, Z. (2009). Research on Agile Project Management with Scrum Method. Teoksessa Guerrero, J. E. (toim.), *Proceedings of 2009 IITA International Conference on Services Science, Management and Engineering, Zhangjiajie, China, July 11-12* (s. 26-29). Los Alamitos, California: IEEE Computer Society.

LIITE 1 HAASTATTELURUNKO

Johdatus

- Tutkijan taustatiedot
- Tutkimuksen tarkoitus, hyöty ja luottamuksellisuus
- Haastattelun eteneminen

Haastateltavan taustatiedot

- Kuinka monta vuotta olet työskennellyt IT-alalla?
- Mitä ohjelmistokehitysmenetelmiä olet käyttänyt?
- Minkälaista kokemusta sinulla on scrumista?
- Kuinka monessa scrum-projektissa olet työskennellyt?
- Ovatko scrum-projektit noudattaneet scrumin viitekehystä?
- Kuvailisitko käynnissä olevaa scrum-projektia ja omaa rooliasi siinä?

Scrumin käyttöönotto

- Onko scrumin viitekehys yksinkertainen ja helposti opittava?
- Onko scrumin käyttöönotto työlästä, aikaavievää tai haastavaa?
- Voivatko kehittäjät vastustaa scrumin käyttöönottoa?
- Voiko scrumitiimi muuttaa scrumin viitekehystä tai käyttää viitekehystä vain osittain?
- Pitävätkö kehittäjät työskentelystä scrum-projekteissa?

Kehitystiimin koko

- Onko kehitystiimi sopivan kokoinen?
- Kuinka usein kehittäjät ovat vuorovaikutuksessa toistensa kanssa?
- Onko yrityksellä tarpeeksi resursseja muodostaa optimaalisen kokoisia kehitystiimejä?
- Toimivatko kehittäjät samaan aikaan useissa eri projekteissa?

Kehitystiimin työskentely-ympäristö

- Työskentelevätkö kehittäjät samalla työpaikalla?
- Työskentelevätkö kehittäjät avoimessa työtilassa?
- Tukeeko työskentely avoimessa työtilassa kehittäjien välistä vuorovaikutusta ja yhteistyötä?
- Vaikuttavatko avoimeen työtilaan liittyvät häiriötekijät kehittäjien työntekoa?

Kehitystiimin monitaitoisuus

- Onko kehitystiimillä tarvittava osaaminen toteuttaa scrum-projekti ilman ulkopuolista apua?
- Onko kehittäjillä selkeät roolit vai edellytetäänkö kehittäjiltä laajaa osaamista?
- Toteuttavatko kehittäjät vain oman osaamisalueensa mukaisia työtehtäviä?
- Neuvovatko kokeneemmat kehittäjät kokemattomampia kehittäjiä?
- Tukeeko työskentely monitaitoisessa kehitystiimissä kehittäjien välistä tiedon jakamista?
- Parantaako työskentely monitaitoisessa kehitystiimissä kehittäjien osaamista?
- Onko scrum-projektiin helppo löytää kehittäjiä, joilla on tarvittava osaaminen?

Kehitystiimin itseohjautuvuus

- Päättääkö kehitystiimi itse työnjaosta ja työskentelytavoistaan?
- Antaako scrummaster ja muut scrum-projektin sidosryhmät työnjakoon ja työskentelytapoihin liittyvän päätöksentekovallan kehitystiimille?
- Lisääkö työskentely itseohjautuvassa kehitystiimissä kehittäjien mahdollisuuksia vaikuttaa omiin työtehtäviinsä?
- Parantaako työskentely itseohjautuvassa kehitystiimissä kehitystiimin toiminnan koordinointia?
- Parantaako työskentely itseohjautuvassa kehitystiimissä kehittäjien motivaatiota, scrum-projektiin sitoutumista, tehokkuutta tai tuottavuutta?

Kehitystiimin yhteistyö

- Tekevätkö kehittäjät yhteistyötä muiden kehittäjien kanssa?
- Toteuttavatko kehittäjät työtehtäviä yhdessä?
- Parantaako kehittäjien tekemä yhteistyö kehittäjien vuorovaikutustaitoja tai kehittäjien välistä vuorovaikutusta?
- Tukeeko kehittäjien tekemä yhteistyö ryhmähengen muodostumista?
- Keskitytäänkö scrum-projektissa kehittäjien yksilösuoritukseen vai kehitystiimin yhteiseen suoritukseen?
- Haluavatko kehittäjät työskennellä itsenäisesti vai tehdä yhteistyötä?
- Vaikuttavatko kehittäjien vuorovaikutustaidot kehittäjien yhteistyöhön tai ryhmähengen muodostumiseen?

Kehitystiimin työn läpinäkyvyys

- Ovatko kehittäjät jatkuvasti tietoisia toistensa tekemästä työstä, sprintin etenemisestä ja etenemistä hidastavista tekijöistä?
- Mintoiko kehitystiimin työn ja sprinttien edistymisen jatkuva seuranta kehittäjiä aikaansaamaan jotain joka päivä?
- Voivatko kehittäjät raportoida kohtaamistaan ongelmista heti niiden ilmaantua ja pyytää tarvittaessa apua muilta kehittäjiltä tai scrummasterilta?
- Vähentääkö työn läpinäkyvyys tuotteeseen liittyviä virheitä ja helpottaa ja nopeuttaa virheiden korjaamista?

- Ilmeneekö kehittäjien heikko osaaminen, huono työnjälki tai sovitusta työtehtävistä ja aikatauluista poikkeaminen helposti scrum-projektissa?
- Lisääkö työn läpinäkyvyys kehittäjiin kohdistuvia paineita?
- Voiko työn läpinäkyvyys vaikuttaa negatiivisesti työskentelyilmapiiriin?
- Voivatko kehittäjät tuntea scrum-projektin tapahtumiin osallistumisen ja tekemänsä työn esittelyn muille kehittäjille epämukavaksi?

Kehitystiimin työmäärä ja työskentelytahti

- Auttaako työskentely sprinteissä pitämään kehitystiimin työmäärän kohtuullisena koko scrum-projektin ajan?
- Väheneekö ylitöiden määrä scrumin ansiosta?
- Onko realististen työmääräarvioiden tekeminen helppoa?
- Voiko kehitystiimi sitoutua tekemään liian paljon työtä yhden sprintin aikana?
- Voivatko scrum-projektin tapahtumat pitkittyä?
- Onko scrum-projektin tapahtumille varattu sopivasti aikaa?
- Onko sopivan ajankohdan löytäminen scrum-projektin tapahtumille helppoa?

Kehitystiimin tuottaman dokumentaation määrä

- Minkälaista dokumentaatiota kehitystiimi tuottaa?
- Kuinka paljon dokumentaation tuottaminen vie kehittäjien aikaa?
- Syntyykö kehitystiimin keskuudessa dokumentaation lisäksi myös hiljaista tietoa?
- Voiko oleellista tietoa kadota, jos kehittäjät vaihtuvat tai lähtevät kehitystiimistä kesken scrum-projektin?
- Helpottaako olemassa oleva dokumentaatio uusien tai kokemattomampien kehittäjien perehdyttämistä?

Kehitystiimin yhteistyö tuoteomistajan kanssa

- Onko tuoteomistaja sitoutunut osallistumaan scrum-projektiin?
- Tekeekö kehitystiimi säännöllistä yhteistyötä tuoteomistajan kanssa?
- Saako kehitystiimi tuoteomistajalta säännöllisesti palautetta tekemästään työstä?
- Kasvattaako yhteistyö tuoteomistajan kanssa kehitystiimin luottamusta siihen, että kehitettävä tuote vastaa asiakkaan vaatimuksia?
- Edellyttääkö yhteistyö tuoteomistajan kanssa kehittäjiltä asiakkaan toimialaan ja liiketoimintaan liittyvää tietoa?
- Edellyttääkö yhteistyö tuoteomistajan kanssa kehittäjiltä hyviä vuorovaikutus- ja esiintymistaitoja?

Vaatimusten muuttuminen scrum-projektin aikana

- Uskovatko kehittäjät voivansa selvittää kehitettävää tuotetta koskevat vaatimukset scrum-projektin alussa?
- Miten kehittäjät suhtautuvat siihen, että tuoteomistaja voi muuttaa tuotetta koskevia vaatimuksia scrum-projektin aikana?
- Huomioiko kehitystiimi muuttuneet vaatimukset nopeasti?

- Voiko tuoteomistaja määritellä tuotetta koskevat vaatimukset huonosti?

Tuleeko mieleesi muita scrumiin liittyviä hyötyjä ja haasteita, joita ei tämän haastattelun yhteydessä ole tullut esille?