

SAVONIA



OPINNÄYTETYÖ - AMMATTIKORKEAKOULUTUTKINTO
TEKNIIKAN ALA

SCRUM-VIITEKEHYS: OSAAMINEN JA KÄYTÄNNÖT PONSSEN KEHITYSTIIMEISSÄ

TEKIJÄ Emilia Hakomäki

Koulutusala Tekniikan ja liikenteen ala		
Tutkinto-ohjelma Tietotekniikan tutkinto-ohjelma		
Työn tekijä Emilia Hakomäki		
Työn nimi Scrum-viitekehys: osaaminen ja käytännöt Ponsen kehitystiimeissä		
Päiväys	12.3.2025	Sivumäärä/Liitteet 43/4
Yhteistyötaho Ponsse Oyj		
<p>Ketteryys ja iteratiiviset kehitysmenetelmät ovat vakiinnuttaneet asemansa ohjelmistokehityksessä, tarjoten joustavuutta ja tehokkuutta sekä projektinhallintaan että tiimityöskentelyyn. Scrum on yksi yleisimmin käytetyistä viitekehyksistä ketterässä kehityksessä. Menetelmän tehokkuus riippuu kuitenkin siitä, kuinka hyvin tiimit ymmärtävät ja soveltavat Scrumin periaatteita käytännössä. Tämän opinnäytetyön tilaajana toimi Ponsse Oyj. Työn tavoitteena oli tutkia, miten hyvin digitaalisten palveluiden kehitystiimien jäsenet tuntevat Scrumin teoriaan kuuluvat pilarit ja arvot ja kuinka nämä näkyvät tiimien arjessa. Toisena tavoitteena oli selvittää, voidaanko lyhyellä interventiolla vahvistaa yhden tiimin Scrum-osaamista. Opinnäytetyön kolmantena tavoitteena oli selvittää, miten kehitystiimin jäsenet kokevat Ponsen nykyisten Scrum-käytäntöjen tukevan heidän työtään.</p> <p>Tutkimus toteutettiin toimintatutkimuksena, ja aineistonkeruumenetelmänä käytettiin kyselylomaketta. Tutkimuksessa hyödynnettiin myös havainnointia ja interventiota. Tutkimuksen ensimmäisessä vaiheessa toteutettiin kysely, joka kohdistettiin Scrum-viitekehystä työssään käyttäville tiimeille. Kyselyllä kartoitettiin tiimien jäsenten tietämystä Scrumista ja kokemuksia nykyisistä Scrum-käytännöistä. Tutkimuksen toisessa vaiheessa järjestettiin interventio yhdelle kehitystiimeistä. Interventio koostui kolmesta koulutuskerrasta, joiden aikana käytiin läpi Scrum-viitekehysten pilarit, arvot ja tapahtumat. Opinnäytetyön tekijä osallistui myös tiimin Scrum-tapahtumiin ja mukautti koulutuksen sisältöä havaintojensa perusteella. Tutkimuksen lopuksi toteutettiin vielä loppukysely interventioon osallistuneelle tiimille.</p> <p>Tutkimuksen tulokset osoittavat, että Scrumin pilarien ja arvojen ymmärtäminen ja toteutuminen arjessa vaihteli tiimeittäin. Lyhyellä interventiolla pystyttiin parantamaan yhden kehitystiimin Scrum-osaamista. Suurin osa tutkimukseen osallistuneista koki, että Ponsen nykyiset Scrum-käytännöt tukevat tiimin toimintaa. Samalla tutkimuksessa tunnistettiin sekä vahvuuksia että heikkouksia Ponsen Scrum-käytännöissä, joiden pohjalta esitettiin kehitysideoita. Tutkimuksen perusteella voidaan todeta, että Scrum-viitekehystä ei tulisi toteuttaa kaavamaisesti, vaan sen soveltaminen ja muokkaaminen tiimin tarpeisiin on tärkeää. Jatkotutkimuksen osalta aiheetta olisi syytä tutkia syvällisemmin. Pidempikestoinen tutkimus, jossa hyödynnettäisiin koulutuksen lisäksi esimerkiksi haastatteluja, voisi antaa paremman kuvan tiimin tilanteesta ja osaamisen kehittymisestä. Opinnäytetyö oli merkityksellinen tutkijan ammatillisen kasvun näkökulmasta. Ponselle tutkimus tarjosi laajan kuvan Scrumin käytöstä kehitystiimeissä sekä mahdollisia kehitysideoita Scrum-käytäntöjen parantamiseksi.</p>		
Avainsanat Ohjelmistotuotanto, perinteinen ohjelmistokehitys, ketterät menetelmät, Scrum-viitekehys, toimintatutkimus		

SISÄLTÖ

1	JOHDANTO.....	5
2	OHJELMISTOTUOTANNON LÄHESTYMISTAVAT: VESIPUTOUSMALLISTA KETTERÄÄN KEHITYKSEEN	6
2.1	Ohjelmistotuotanto ja ohjelmistokehityksen elinkaarimalli	6
2.2	Perinteiset ohjelmistokehitysmenetelmät	8
2.3	Ketterät menetelmät	10
3	SCRUM-VIITEKEHYS	13
3.1	Scrum-viitekehityksen teoria	13
3.2	Scrum-tiimi ja roolit.....	14
3.3	Scrumin tuotokset ja tapahtumat.....	15
3.4	Scrumin vahvuudet ja heikkoudet	17
4	TUTKIMUKSEN TOTEUTUS.....	19
4.1	Tutkimuksen tavoite ja tutkimuskysymykset	19
4.2	Toimintatutkimus tutkimusmenetelmänä.....	19
4.3	Tutkimusaineiston keruu: kyselyt ja interventio.....	20
4.4	Aineiston käsittely ja analyysi.....	21
5	TUTKIMUKSEN TULOKSET	23
5.1	Scrum-viitekehitys Ponsen kehitystiimeissä.....	23
5.2	Intervention vaikutus kehitystiimin Scrum-osaamiseen	30
5.3	Scrum-käytäntöjen vahvuudet ja heikkoudet	31
6	POHDINTA.....	35
6.1	Tutkimustulosten tarkastelu.....	35
6.2	Tutkimuksen eettisyys ja luotettavuus.....	37
6.3	Jatkotutkimusehdotukset ja opinnäytetyön merkitys	39
	LÄHTEET	41
	LIITE 1: KYSELYLOMAKE.....	44
	LIITE 2: TIETOSUOJAILMOITUS	50
	LIITE 3: SUOSTUMUSLOMAKE INTERVENTIOON OSALLISTUMISESTA	54
	LIITE 4: MIRO-TAULUT	55

KUVALUETTELO

Kuva 1. Ohjelmistokehityksen elinkaarimalli (mukaillen Gupta ym. 2021)	7
Kuva 2. Vesiputousmalli (mukaillen Royce 1970)	8
Kuva 3. Spiraalimalli (mukaillen Sommerville 2011; Tsui ym. 2022).....	9
Kuva 4. Scrum-viitekehyksen pilarit ja arvot.....	14
Kuva 5. Scrum-viitekehyksen tapahtumat ja tuotokset (mukaillen Novriansyah 2024).....	16
Kuva 6. Toimintatutkimuksen perusmalli (Jyrkämä 2021).....	20
Kuva 7. Scrum-viitekehyksen pilarien ja arvojen tuntemus kehitystiimeissä.....	23
Kuva 8. Läpinäkyvyyden ymmärtäminen kehitystiimeissä	24
Kuva 9. Tarkastelun ymmärtäminen kehitystiimeissä	26
Kuva 10. Mukauttamisen ymmärtäminen kehitystiimeissä	27
Kuva 11. Yhteisten arvojen merkitys tiimin menestykselle	30
Kuva 12. Scrum-tapahtumien hyödyllisyys.....	31
Kuva 13. Ponsen Scrum-käytäntöjen vahvuudet.....	32
Kuva 14. Ponsen Scrum-käytäntöjen heikkoudet.....	33

1 JOHDANTO

Ohjelmistotuotanto ja -kehitys ovat kehittyneet valtavasti suhteellisen lyhyen historiansa aikana, sillä ensimmäiset ohjelmistot suunniteltiin vasta 1950-luvun puolivälissä. Alan nopea kehitys on tuonut mukanaan uusia menetelmiä, työkaluja ja lähestymistapoja, jotka ovat muokanneet tapoja suunnitella, kehittää ja ylläpitää ohjelmistoja. Erityisesti 2000-luvun alkuvuosina ohjelmistokehityksen perinteinen vesiputousmalli, joka oli pitkään vallitseva kehitysprosessi, on saanut väistyä joustavampien ja iteratiivisempien lähestymistapojen tieltä. Tämä muutos on osittain seurausta jatkuvasti kasvavasta tarpeesta vastata entistä nopeammin markkinoiden ja asiakkaiden vaatimuksiin sekä tarpeesta kehittää ohjelmistoja entistä nopeammin ja laadukkaammin. Tämän muutoksen keskiössä on ketterän ohjelmistokehityksen julistus (Agile Manifesto), joka julkaistiin vuonna 2001 ja toi mukanaan uudenlaisen ajattelutavan ohjelmistokehitykseen. Julistus määrittelee ketterän ohjelmistokehityksen peruseriaatteen, jotka keskittyvät joustavuuteen, yhteistyöhön ja asiakaslähtöisyyteen.

Ketteriin menetelmiin kuuluvat muun muassa Scrum, Kanban, Extreme Programming, Lean ja SAFe, mutta ne eroavat toisistaan siinä, miten niitä käytännössä sovelletaan ja mitä ne painottavat. Scrum on erityisesti saanut suosiota iteratiivisten ja inkrementaalisten kehitysprosessien tarjoajana, kun taas Kanban keskittyy työn virran hallintaan ja jatkuvaan parantamiseen. Extreme Programming puolestaan korostaa koodin laatua ja tiivistä yhteistyötä asiakkaan kanssa. Lean keskittyy arvon tuottamiseen asiakkaille minimoimalla hukkaa ja tehostamalla prosesseja. SAFe taas tarjoaa laajemman mallin ketterän kehityksen skaalaukseen suuremmissa organisaatioissa ja monimutkaisissa projekteissa.

Tämän opinnäytetyön aiheena on *Scrum-viitekehitys: osaaminen ja käytännöt Ponssen kehitystiimeissä*. Työn tavoitteena on tutkia, miten Ponssen digitaalisten palveluiden kehitystiimit tuntevat Scrumin teorian ja miten se toteutuu käytännössä. Toisena tavoitteena on selvittää, voidaanko lyhyellä, kolmen sprintin mittaisella interventiolla, vaikuttaa yhden kehitystiimin käsitykseen Scrumista. Lisäksi tutkimuksen tavoitteena on selvittää, miten Ponssen nykyiset Scrum-käytännöt tukevat kehitystiimejä. Työn toimeksiantajana on Ponsse Oyj, joka on vieremäläinen metsäkoneiden valmistukseen erikoistunut yritys. Ponssen palveluja ovat metsäkoneiden valmistuksen ja myynnin lisäksi koulutus, huolto ja varaosat sekä digitaaliset palvelut. Ponssen digitaalisten palveluiden kehitystiimeissä hyödynnetään Scrum-viitekehystä.

Opinnäytetyö on saanut alkunsa tutkijan omasta kiinnostuksesta ketteriin kehitysmenetelmiin ja erityisesti Scrum-viitekehitykseen, sillä tutkija on työskennellyt eräässä Ponssen digitaalisten palveluiden kehitystiimissä Scrum Masterina. Myös työn toimeksiantaja on kiinnostunut kehitystiimien nykytilanteesta ja siitä, miten tiimien ketteryyttä, toimivuutta ja tuottavuutta voitaisiin kehittää. Tutkimus koostuu kolmesta osasta: alkukyselystä Scrumia hyödyntäville kehitystiimeille, kolmen sprintin mittaisesta interventiosta yhdelle tiimille sekä loppukyselystä, joka toteutetaan interventioon osallistuneelle tiimille. Työn tulokset tarjoavat arvokasta tietoa Scrum-viitekehityksen nykytilanteesta Ponsella ja lisäksi työ auttaa tunnistamaan nykyisten käytäntöjen vahvuudet ja kehittämiskohteet.

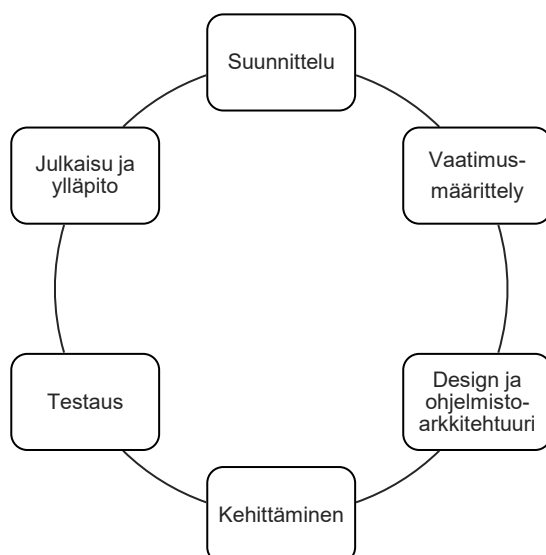
2 OHJELMISTOTUOTANNON LÄHESTYMISTAVAT: VESIPUTOUSMALLISTA KETTERÄÄN KEHITYKSEEN

2.1 Ohjelmistotuotanto ja ohjelmistokehityksen elinkaarimalli

Ohjelmistotuotanto (software engineering) terminä esiteltiin Saksassa vuonna 1968 NATO:n Software Engineering konferenssissa, kun 1960-luvun lopulla herättiin ”ohjelmistokriisiin” tietotekniikan alalla. Tietokonelaitteiston laskentateho ja muistikapasiteetti olivat kehittyneet nopeasti, mikä mahdollisti suurien ja monimutkaisten ohjelmistojen kehittämisen. Kuitenkin varhaiset kokemukset ohjelmistojen tekemisestä olivat riittämättömät ja tämän vuoksi projektit viivästyivät, ylittivät budjetin tai eivät koskaan valmistuneet. Myöskään ohjelmistojen ylläpitoon tai tukeen ei ollut osaamista. Konferenssin järjestäjät tunnistivat tarpeen kurinalaiselle lähestymiselle ohjelmistojen tuotantoon ja muodostivat näin ohjelmistotuotannon käsitteen. (Sommerville 2011, 5; Tsui, Karam & Bernai 2022, luku 3.)

Ohjelmistotuotannolle ei ole yhtä selkeää määritelmää. SWEBOK (Software Engineering Body of Knowledge) on IEEE Computer Society:n julkaisema ohjeistus, joka määrittelee ohjelmistotuotannon systemaattiseksi, kurinalaiseksi ja mitattavissa olevaksi tavaksi ohjelmistojen kehittämiseen, opeointiin ja ylläpitoon (Washizaki 2024, 39). Sommervillen (2011, 5) mukaan ohjelmistotuotanto ei ole vain ohjelmoinnin suorittamista vaan se kattaa myös ohjelmistojen määrittelyn, suunnittelun ja kehittämisen. Ohjelmistokehitys (software development) on prosessi, joka sisältää käyttäjän vaatimusten määrittelyn, analysoinnin, suunnittelun ja toteutuksen. Ohjelmistokehitys on osa ohjelmistotuotantoa, joka taas kattaa koko prosessin, mukaan lukien projektinhallinnan, aikataulutuksen ja arvioinnin, henkilöstöhallinnan ja monia muita osa-alueita. (Dooley 2011, 1; Foster 2011, luku 1.)

Ohjelmistotuotannon yhteydessä puhutaan usein ohjelmistokehityksen elinkaarimallista ja elinkaaren vaiheista (Software Development Life Cycle, SDLC). Elinkaarimalli on jäsenelty lähestymistapa ohjelmistojen kehittämiseen, joka sisältää vaiheet ohjelmiston suunnittelusta käyttöönottoon ja ylläpitoon saakka. Mallin tavoitteena on mahdollistaa korkealaatuisten ohjelmistojen kehittäminen sekä kustannustehokkaasti että aikataulussa pysyen. (Foster 2011, luku 1; Gupta, Rawal & Barge 2021, 73.) Lähteestä riippuen ohjelmistokehityksen elinkaarimalli koostuu yleensä 5–8 eri vaiheesta, mutta mallien tarkoitus on aina sama: varmistaa laadukas, hallittu ja tehokas ohjelmistokehitysprosessi. Gupta ym. (2021, 73–74) jaottelevat ohjelmistokehityksen elinkaaren kuuteen eri vaiheeseen: suunnittelu, vaatimusten määrittely, design ja ohjelmistoarkkitehtuuri, kehittäminen, testaus, julkaisu ja ylläpito (kuva 1).



Kuva 1. Ohjelmistokehityksen elinkaarimalli (mukaiillen Gupta ym. 2021)

Suunnittelu: Tämä on ohjelmistokehityksen tärkein vaihe, jossa keskitytään projektin suunnitteluun ja analyysiin. Asiakkailta ja sidosryhmiltä kerätään liiketoimintavaatimuksia, jotta voidaan määrittellä projektin toteutettavuus ja vaatimukset. Vaihe sisältää myös riskien arvioinnin ja laadunvarmistuksen suunnittelun.

Vaatimusten määrittely: Vaatimusanalyysin valmistuttua laaditaan vaatimusmäärittelydokumentti, jonka asiakas tai muu sidosryhmä hyväksyy. Tämä dokumentti ohjaa koko kehitysprosessia.

Design ja ohjelmistoarkkitehtuuri: Tässä vaiheessa ohjelmiston arkkitehtuuri suunnitellaan vaatimusmäärittelydokumentin pohjalta. Suunnittelussa otetaan huomioon ohjelmiston koko rakenne ja esimerkiksi ohjelmointikieli, modulaarisuus ja komponentit. Lisäksi ohjelmiston riskit, kustannukset ja suorituskyky huomioidaan tässä vaiheessa.

Kehittäminen: Itse ohjelmiston kehitys tapahtuu tässä vaiheessa ja se noudattaa aiemmissa vaiheissa tehtyjä dokumentteja. Huolellinen analyysi- ja suunnittelutyö mahdollistavat sujuvan kehittämisen.

Testaus: Testausvaiheessa varmistetaan, että ohjelmisto täyttää määritellyt vaatimukset. Testauksessa käytetään esimerkiksi yksikkö- ja integraatiotestejä sekä järjestelmätestausta. Tavoitteena on tunnistaa ja korjata virheet, jotta tuote vastaa laatuvaatimuksia.

Julkaisu ja ylläpito: Kun ohjelmisto on testattu hyvin, siirrytään julkaisuvaiheeseen. Ohjelmistolle tehdään hyväksyntätestaus asiakkaan tai sidosryhmien toimesta ja tuote toimitetaan asiakkaalle. Julkaisun jälkeen ohjelmisto siirtyy ylläpitovaiheeseen, jossa sitä ylläpidetään ja kehitetään käyttäjiltä tulleiden palautteiden perusteella. Ylläpito voi sisältää myös ohjelmiston päivityksiä, korjauksia ja uusien versioiden julkaisuja, jotta ohjelmisto pysyy turvallisena ja ajan tasalla, jolloin tuote vastaa edelleen asiakkaiden tarpeisiin. (Bluestein 2023; Gupta ym. 2021, 73–74; Futurum Technology 2024; Tutorials Point n.d.)

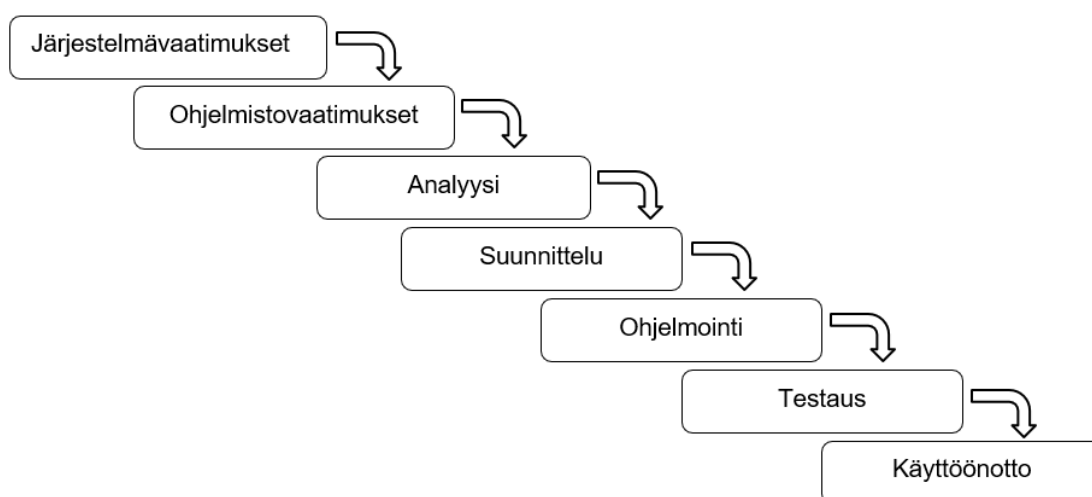
Ohjelmistokehityksen elinkaarimallin käyttämisen etuina on tunnistettu ohjelmiston parempi laatu, kehitysprosessin läpinäkyvyyden lisääntyminen sidosryhmille, parempi projektinhallinta sekä kehittäjien että sidosryhmien näkökulmasta ja tehokkaampi riskienhallinta (Bhatt 2023; Futurum Technology 2024). Heikkouksia elinkaarimallissa Ghoshin (2024) mukaan ovat aikataulujen venyminen, jäykkä rakenne, korkeat alkuinvestoinnit ja liiallinen keskittyminen itse kehitysprosessiin lopputuotteen sijasta.

Ohjelmistokehitysmalleja on luotu useita erilaisia ja niitä voidaan jaotella eri tavoin. Tunnettu jako ohjelmistokehityksessä on jako perinteisiin suunnittelulähtöisiin/ennakoiviin menetelmiin ja uudempiin ketteriin/mukautuviin kehitysmalleihin. Suunnittelulähtöisissä menetelmissä prosessi etenee tiukasti vaiheittain, ja siirtyminen seuraavaan vaiheeseen edellyttää edellisen vaiheen hyväksymistä. Ketterissä malleissa prosessi sen sijaan on iteratiivinen ja dokumentointi on vähäisempää. Projektin toimiala, laajuus, aikataulu ja tiimi määrittävät, mikä kehitysmalli sopii parhaiten kyseiseen projektiin. (Dooley 2011, luku 2; Stephens 2015, 267.)

2.2 Perinteiset ohjelmistokehitysmenetelmät

Vaikka modernit ja ketterät menetelmät ovat yleistyneet, perinteiset lähestymistavat tarjoavat edelleen vakautta, selkeyttä ja ennakoitavuutta projektien läpivientiin. Perinteisiin menetelmiin luetellaan yleensä kuuluviksi vesiputousmalli, v-malli, RUP, prototyypimalli ja spiraalimalli. Seuraavaksi käydään tarkemmin läpi vesiputousmalli ja spiraalimalli sekä tarkastellaan näiden vahvuuksia ja heikkouksia.

Ohjelmistokehityksen **vesiputousmallin** (kuva 2) esitteli ensimmäisen kerran Royce vuonna 1970. Malli kuvaa ohjelmistokehityksen elinkaarta, jossa kehitys etenee lineaarisesti vaiheittain: jokaisen vaiheen tuotos siirtyy seuraavan vaiheen pohjaksi vasta, kun edellinen vaihe on saatu valmiiksi. Vesiputousmallin jokaisessa vaiheessa syntyy dokumentti, joka tulee hyväksyä ennen siirtymistä seuraavaan vaiheeseen.



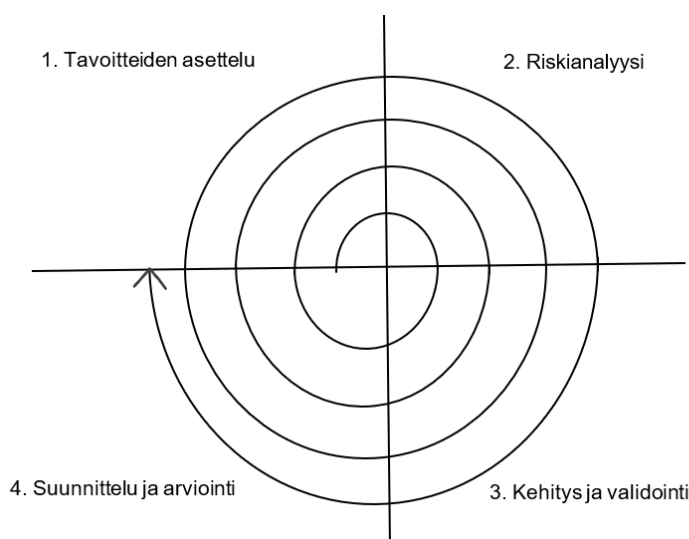
Kuva 2. Vesiputousmalli (mukaillen Royce 1970)

Vesiputousmallin merkittäviä heikkouksia ovat sen joustamattomuus ja kustannukset. Vesiputousmalli perustuu siihen oletukseen, että vaatimukset voidaan määrittellä projektin alussa. Muutokset projektin myöhemmissä vaiheissa eivät ole ainoastaan kalliita, vaan vaativat myös usein paluuta

kehitysprosessin alkuvaiheisiin. Asiakas ja loppukäyttäjät eivät ole tiiviisti mukana projektin eri vaiheissa, mikä voi heikentää asiakastytyvyyttä. Yleensä asiakas osallistuu projektin alkuvaiheen vaatimusmäärittelyyn, ja välietappeja projektin edistymisestä ei ole. Tämä vaikeuttaa palautteen saamista ja asiakkaan muuttuvien tarpeiden huomioimista, jolloin lopputuote ei välttämättä vastaa asiakkaan toiveita. Vesiputousmallissa riskienhallinta ja virheiden myöhäinen havaitseminen ovat myös heikkouksia. Malli olettaa, että riskit voidaan tunnistaa projektin alussa ja ohjelmiston testaaminen tapahtuu vasta kehityksen lopuksi. Lisäksi pitkät toimitusajat ja resurssien tehoton hyödyntäminen nähdään mallin heikkouksina. Jokainen vaihe on saatettava loppuun ennen siirtymistä seuraavaan, ja osa tiimin jäsenistä voi joutua odottamaan edellisen vaiheen valmistumista, ennen kuin voivat aloittaa oman työnsä. (Dwivedi, Katiyar & Goel 2022, 143; Hossain 2023, 5–6; Pargaonkar 2023, 123.)

Vahvuutena vesiputousmallissa on sen selkeä ja jäsenelty rakenne; selkeä kehitysmalli helpottaa projektin suunnittelua ja hallintaa. Projektin lineaarinen eteneminen lisää ennakoitavuutta ja mahdollistaa tarkat aikataulut. Vesiputousmallissa laaditaan useita dokumentteja, jotka tukevat esimerkiksi tuotteen myöhempää ylläpitoa. Malli sopii hyvin esimerkiksi pieniin projekteihin, joissa on pysyvät vaatimukset. (Dwivedi ym. 2022, 143; Hossain 2023, 4–5; Pargaonkar 2023, 123.) Vaikka vesiputousmalli tarjoaa selkeän rakenteen ja ennakoitavuutta, se ei välttämättä sovellu projekteihin, joissa vaatimukset muuttuvat tai joissa tarvitaan enemmän joustavuutta.

Boehmin vuonna 1988 esittelemä **spiraalimalli** yhdistää vesiputousmallin ja iteratiivisen lähestymistavan. Malli painottaa riskienhallintaa, ja ohjelmiston kehitys etenee spiraalimaisesti (kuva 3). Mallissa on neljä päävaihetta, jotka toistuvat jokaisessa kierroksessa: tavoitteiden asettaminen, riskianalyysi, kehitys ja validointi sekä suunnittelu ja arviointi. Spiraali alkaa tavoitteiden määrittelyllä ja menetelmien sekä rajoitteiden arvioinnilla. Toisessa vaiheessa tehdään riskianalyysi ja mahdollisesti toimenpiteitä riskien lieventämiseksi, esimerkiksi prototyyppi. Kehitys- ja validointivaiheessa tapahtuu itse koodaus, testaus ja dokumentointi. Neljännessä vaiheessa, suunnittelu ja arviointi, projekti arvioidaan ja päätetään, jatketaanko sitä. Jos projekti saa jatkoa, laaditaan suunnitelma seuraavalle spiraalin kierrokselle. Näitä neljää vaihetta toistetaan, kunnes projekti on valmis. (Sommerville 2011, 48–50; Tsui ym. 2022, luku 4.)



Kuva 3. Spiraalimalli (mukaan Somerville 2011; Tsui ym. 2022)

Spiraalimallin heikkouksina pidetään muun muassa sen monimutkaisuutta. Malli koostuu useista iteraatioista, joista jokainen sisältää useita vaiheita. Projektille on vaikea ennustaa tarkkaa valmistusajankohtaa. Spiraalimalli vaatii huolellista resurssien suunnittelua, sillä kustannukset voivat helposti nousta. Tämä johtuu siitä, että tarkkaa aikataulua ei voida ennustaa, ja kehitys vaatii useita työntekijöitä. Yksi heikkous mallissa on myös riskianalyysi, jonka onnistuminen vaatii erittäin tarkkaa ja erikoistunutta osaamista. Epäonnistunut riskianalyysi voi vaikuttaa koko projektin etenemiseen ja valmistumiseen. (Hossain 2023, 9.)

Spiraalimalli sopiikin paremmin suuriin ja monimutkaisiin projekteihin, joissa riskianalyysi on tärkeässä roolissa. Mallissa tuotetaan paljon dokumentaatiota, mikä varmistaa projektin etenemisen ja tehtyjen päätösten selkeän kirjaamisen. Tämä dokumentaatio voi olla hyvin arvokasta myöhemmässä vaiheessa tuotteen elinkaarta. Vahvuutena spiraalimallissa on myös prototyyppien varhainen kehittäminen, joka mahdollistaa varhaisen testaamisen ja asiakkaalta palautteen saamisen. Asiakas on mukana koko kehitysprosessin ajan, mikä varmistaa, että tuote vastaa asiakkaan odotuksia ja tarpeita. (Dwivedi 2022, 143; Hossain 2023, 8; Tsui ym. 2022, luku 4.) Verrattuna vesiputousmalliin spiraalimalli on joustavampi ja asiakaslähtöisempi. Spiraalimallissa kehitys tapahtuu vaiheittain, ja ohjelmisto kehittyy jatkuvasti, jolloin asiakkaiden tarpeisiin voidaan vastata nopeammin. Sen sijaan vesiputousmallissa ohjelmistokehitys etenee lineaarisesti suunnittelusta valmiiksi tuotteeksi ilman mahdollisuutta muuttaa suuntaa myöhemmissä vaiheissa.

2.3 Ketterät menetelmät

Jo 1990-luvulla ”kevyet menetelmät” alkoivat nousta suosioon, kun kehittäjät turhautuivat raskaisiin ja jäykkiin suunnittelulähtöisiin menetelmiin, jotka sopivat parhaiten isoihin ja monimutkaisiin projekteihin (Sommerville 2011, 58). Kevyet menetelmät, kuten Extreme Programming ja Scrum, keskittyivät enemmän itse ohjelmiston kehittämiseen kuin laajaan suunnitteluun ja dokumentointiin. Ominaisista kevyissä menetelmissä oli myös kehitystiimin ja sidosryhmien välinen tiivis yhteistyö, itseohjautuvat kehitystiimit sekä nopea, vaihteittainen kehitys ja julkaisu. (Agile Alliance, n.d.)

Nykyään tunnettu termi ”ketterät kehitys” sai alkunsa vuonna 2001, kun joukko ohjelmistokehittäjiä kokoontui Utahiin keskustelemaan ohjelmistokehityksen haasteista ja käyttämistään menetelmistä. Samassa tapaamisessa syntyi myös Agile Alliance -järjestö, jonka tavoitteena on edistää ketteriä menetelmiä. Lisäksi julkaistiin myös ketterän ohjelmistokehityksen julistus (Agile Manifesto) sekä 12 periaatetta, jotka ohjaavat ketterää ohjelmistokehitystä. (Agile Alliance 2001; Agile Alliance, n.d.)

Löydämme parempia tapoja tehdä ohjelmistokehitystä, kun teemme sitä itse ja autamme muita siinä. Kokemuksemme perusteella arvostamme:

Yksilöitä ja kanssakäymistä enemmän kuin menetelmiä ja työkaluja

Toimivaa ohjelmistoa enemmän kuin kattavaa dokumentaatiota

Asiakasyhteistyötä enemmän kuin sopimusneuvotteluja

Vastaamista muutokseen enemmän kuin pitäytymistä suunnitelmassa

Jälkimmäisilläkin asioilla on arvoa, mutta arvostamme ensiksi mainittuja enemmän. (Agile Alliance 2001)

Ketterä kehitys ei itsessään ole mikään ohjelmistokehityksen malli, metodi tai viitekehys vaan se tarkoittaa enemmän mentaliteettia, joka perustuu ketterän ohjelmistokehityksen julistukseen. Olenaista ketterille menetelmille on nopeat julkaisut ja iteraatiot, inkrementaalinen suunnittelu, loppukäyttäjien osallistuminen prosessiin, minimalistinen dokumentointi ja vapaamuotoinen kommunikatio (Tsui ym. 2022, luku 5). Seuraavaksi käydään läpi joitakin tunnettuja ketterän kehityksen menetelmiä ja luvussa 3 syvennytään tarkemmin Scrum-viitekehukseen.

Extreme Programming (XP) on ketterän ohjelmistokehityksen menetelmä, jonka Kent Beck kehitti vuonna 1998. Nimi extreme tulee siitä, että menetelmä korostaa nopeaa ja iteratiivista ohjelmistokehitystä, jossa parhaaksi havaitut menetelmät vietään äärimmäiselle tasolle. Esimerkiksi ohjelmistomuutokset integroidaan ja testataan useita kertoja päivässä. Arvot luovat XP:n pohjan. Arvoja ovat kommunikaatio, yksinkertaisuus, palaute, rohkeus ja kunnioitus. (Sommerville 2019, luku 2.) Beck ja Andres (2004) esittelevät kirjassaan 13 erilaista XP:n käytäntöä, joista kuitenkin kehitystiimit poimivat ne tekniikat, jotka parhaiten organisaatiolle sopivat. Sommerville (2019, luku 2) on jakanut XP:n käytännöt yleisesti omaksuttuihin käytäntöihin ja vähemmän käytössä oleviin.

Yleisesti omaksuttuja XP:n käytäntöjä ovat inkrementaalinen suunnittelu, käyttäjätarinat, pienet julkaisut, testilähtöinen kehitys, jatkuva integrointi ja refaktorointi. Järjestelmän vaatimukset määritellään asiakkaan kanssa ja ne kirjoitetaan käyttäjätarinoina. Näiden tarinoiden toteutus etenee inkrementaalisesti, ja jokaiseen inkrementtiin sisältyvät tarinat määräytyvät käytettävissä olevan ajan ja prioriteettien mukaan. Pienet ja tiheät julkaisut varmistavat, että ohjelmiston vähimmäistoiminnallisuus toimitetaan nopeasti, ja jokainen versio lisää tuotteen arvoa. XP:n ydinperiaatteisiin kuuluu testilähtöinen kehitys, jossa testit kirjoitetaan ennen varsinaista koodia, mikä varmistaa ohjelman luotettavuuden ja jatkuvasti testatun version saatavuuden. Jatkuva integraatio edistää sujuvaa kehitysprosessia, sillä jokainen valmis työ testataan ja liitetään välittömästi osaksi ohjelmaa. Viimeisenä yleisenä käytäntönä refaktorointi pitää huolen siitä, että koodin selkeys, luotettavuus, rakenne, turvallisuus ja ylläpidettävyys paranevat jatkuvasti. (Sommerville 2019, luku 2.)

Lean Software Development ja Kanban. Lean on enemmänkin filosofinen lähestymistapa ohjelmistokehitykseen, kun taas Kanban on menetelmä, jonka avulla Leanin periaatteet vietään käytäntöön jokapäiväisessä työssä. Leanin juuret ovat 1900-luvun alkupuolella Toyotan tuotannossa ja tuotekehitysjärjestelmissä. Ohjelmistotuotantoon Leanin lanseerasivat Mary ja Tom Poppendieck vuonna 2003. Leanin seitsemän keskeistä periaatetta ovat hukkan poistaminen, laadun sisään rakentaminen, tiedon luominen, päätösten tekeminen mahdollisimman myöhään, nopea julkaisu, ihmisten arvostaminen ja kokonaisuuden optimointi. Leanissa pyritään tunnistamaan asiakkaan tarpeet ja kehittämään sekä toimittamaan vain se, mitä asiakas tarvitsee, mahdollisimman nopeissa sykleissä. Kaikki ylimääräinen koodi, jota asiakas ei ole tilannut, nähdään hukkana. (Poppendieck & Poppendieck 2003; Poppendieck & Poppendieck 2008.) Rodríguezin ym. (2019) tekemässä laajassa kirjallisuuskatsauksessa Leanin vahvuuksiksi luetellaan asiakastytyväisyyden ja tuottavuuden kasvu, lyhyemmät kehitysajat, parempi työn läpinäkyvyys sekä ongelmien juurisyiden ja pullonkaulojen tunnistaminen. Haasteita Lean-ajattelussa ovat Lean-kulttuurin syventäminen organisaatiossa, johdon osallistaminen kehitystehtäviin ja flow-tilan saavuttaminen kehitystyössä.

Kanban on menetelmä, joka perustuu jatkuvan työnkulun hallintaan ja optimointiin visuaalisten työkalujen avulla. Kanbanin avulla voidaan hallita ohjelmistokehityksen työnkulkua ja varmistaa, että

tiimi työskentelee rajatun määrän kanssa tehtäviä samanaikaisesti ja keskittyy vain tärkeisiin tehtäviin. Kanban-taulu on visuaalinen esitys työnkulusta, joka kuvaa kehityksen eri vaiheita, kuten "tehtävällä", "työn alla" ja "valmis". Jokaisella näistä vaiheista on oma WIP-raja (work in progress). WIP-raja määrittää, kuinka paljon työtä saa olla käynnissä kussakin vaiheessa. Jos esimerkiksi sarakkeessa "työn alla" on täysi WIP-raja käytössä, ei tiimi voi aloittaa uusia työtehtäviä ennen kuin nykyisiä tehtäviä on saatu päätökseen. Työtehtävät esitetään Kanban-taulussa lappuina, jotka ovat prioriteettijärjestyksessä ja siirtyvät vaiheesta toiseen sitä mukaa, kun työn eteneminen mahdollistaa seuraavan siirron. Tämä luo jatkuvan työnkulun ja auttaa tiimiä keskittymään vain tärkeisiin tehtäviin. Kanbanissa ei ole siis ketteriin menetelmiin yleensä kuuluvia sprinttejä, vaan kehitys etenee jatkuvasti vaihe vaiheelta ilman aikarajoitteita. (Ahmad, Markkula & Oivo 2013; Tsui ym. 2022, luku 5.) Kanbanin käytön hyötyinä on tunnistettu työn visualisointi, keskeneräisen työn rajoittaminen, työn läpimenoajan mittaaminen, selkeät säännöt ja tiimin työskentelyn jatkuva parantaminen (Rodríguez ym. 2019).

Muita tunnettuja ketteriä ohjelmistokehityksen menetelmiä tai viitekehyyksiä ovat esimerkiksi Crystal ja sen eri versiot, Feature Driven Development (FDD) ja Rational Unified Process (RUP), joista jokaisella on oma teoria, arvot ja käytännöt. Sommerville (2019, luku 2) on listannut kirjassaan kaikille ketterille menetelmille yhteisiä periaatteita. Ketterissä menetelmissä yhteistä on asiakkaan osallistaminen tiiviisti ohjelmistokehitysprojektiin; asiakas määrittelee ohjelmiston vaatimukset ja osallistuu sen arviointiin. Lisäksi ketterissä menetelmissä korostetaan jatkuvaa muutosten hyväksymistä ja mukautumista. Tuotteen ominaisuuksia ja vaatimuksia pidetään elävänä kokonaisuutena, joka voi muuttua projektin aikana, kun tiimi oppii lisää. Tämä vaatii joustavuutta niin kehitysprosessilta kuin tiimin toimintatavoilta. Ketterissä projekteissa ohjelmiston kehittäminen ja toimittaminen tapahtuu inkrementaalisesti. Tämä mahdollistaa jokaisen version testaamisen ja arvioimisen sekä nopean palautteen hyödyntämisen kehityksessä. Yksinkertaisuus on keskeinen periaate, sekä ohjelmiston että kehitysprosessin turhaa monimutkaisuutta pyritään välttämään. Lisäksi ketterät menetelmät painottavat ihmisten merkitystä prosessien sijaan. Kehitystiimeille annetaan vapaus löytää omat tapansa työskennellä, ja tiimin välistä vuorovaikutusta sekä yhteistyötä pidetään tärkeämpänä kuin tarkkojen prosessien noudattamista.

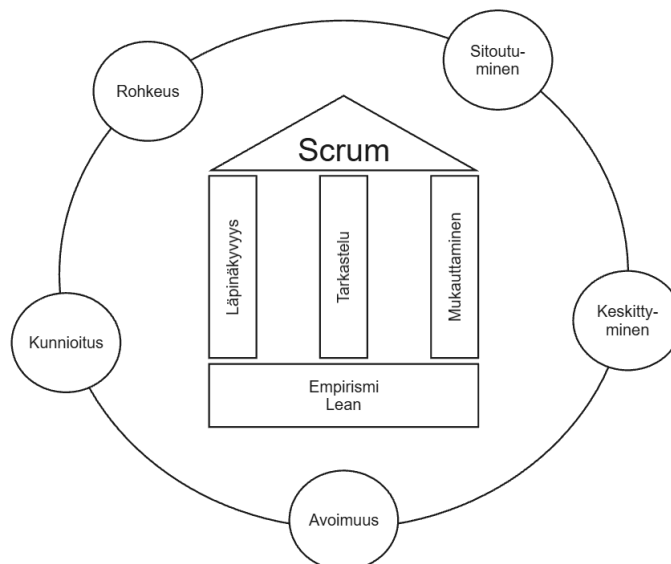
Kun valitaan menetelmää tai viitekehystä, on tärkeä arvioida projektin tavoitteet, organisaation kulttuuri ja tiimien kyvykkyydet. Joskus menetelmiä voidaan räätälöidä yhdistelemällä eri viitekehysten elementtejä, kuten Scrumbanissa. Lopuksi on hyvä muistaa, että mikään menetelmä ei ole täydellinen. Projektin onnistuminen riippuu suurelta osin tiimin kyvystä soveltaa valittua menetelmää ja mukauttaa toimintaansa tilanteen mukaan. Näin ketterä kehitys voi todella vastata asiakkaan tarpeisiin ja tuottaa lisäarvoa nopeasti ja laadukkaasti.

3 SCRUM-VIITEKEHYS

3.1 Scrum-viitekehyyksen teoria

Scrum terminä on esitelty ensimmäisen kerran vuonna 1986, kun Takeuchi and Nonaka vertasivat hyvin suoriutuvia, moniosaavia ja itseohjautuvia kehitystiimejä rugbyssa käytettyyn samannimiseen pelitaktiikkaan. Ken Schwaber ja Jeff Sutherland esittelivät Scrum-viitekehyyksen ensimmäisen kerran konferenssissa vuonna 1995 ja julkaisivat ensimmäisen Scrum-oppaan vuonna 2010. Scrum on kevyt ja joustava viitekehys, joka perustuu iteratiiviseen ja inkrementaaliseen lähestymistapaan. Scrumissa yhdistyvät Lean-ajattelu ja empirismi. Lean-ajattelu keskittyy olennaiseen ja pyrkii vähentämään hukkaa. Empirismi perustuu tiedon hankkimiseen kokemusten ja havaintojen kautta, joita käytetään päätöksenteon tukena. Scrumissa olennaista on itseohjautuvat ja moniosaavat tiimit, jotka toimittavat tuotteita tai palveluita lyhyissä jaksoissa. Scrum mahdollistaa nopean palautteen ja innovoinnin, jatkuvan parantamisen, nopeat muutokset, paremman asiakastytyvyyden ja lyhyemmän toimitusajan. (Schwaber & Sutherland 2020, 3; Scrum Alliance, n.d.) Scrumin teoreettinen perusta koostuu kolmesta pilarista ja viidestä arvosta (kuva 4). Scrumin käytännöt rakentuvat Scrum-tiimistä ja sen rooleista, viidestä tapahtumasta ja tuotoksista (kuva 5). Näitä osa-alueita tarkastellaan seuraavaksi yksityiskohtaisemmin.

Scrumin kolme peruspilaria ovat *läpinäkyvyys, tarkastelu ja mukauttaminen*. Scrum-viitekehyyksen mukaan työn tulee olla läpinäkyvää kaikille kehitystiimin tekijöille ja sidosryhmille, koska tehdyt päätökset perustuvat näihin. Myös Scrum-tiimin yhteisten sopimuksien, työtapojen ja organisaation vision tulee olla läpinäkyviä. Läpinäkyvyys mahdollistaa tarkastelun ja tarkastelu ilman läpinäkyvyyttä voi olla turhaa tai harhaanjohtavaa. Scrumissa tiimin tuotoksia ja etenemistä kohti asetettuja tavoitteita tarkastellaan säännöllisesti, jotta tunnistetaan mahdolliset poikkeamat tai ongelmat. Scrumin viisi tapahtumaa (sprintti, sprintin suunnittelu, katselmointi, retrospektiivi ja päivittäispalaveri) muodostavat tarkastelua tukevan rytmin. Tarkastelu mahdollistaa tiimin toiminnan mukauttamisen ja tarkastelu ilman mukauttamista on turhaa. Kolmas Scrumin peruspilareista on mukauttaminen. Kun tiimi on läpinäkyvyyden mahdollistamana ja tarkastelun seurauksena tunnistanut tarpeen muokata joko tiimin toimintatapoja tai itse kehitettävää tuotetta, tulee se tehdä niin nopeasti kuin mahdollista. Mukauttamisen edellytys on, että tiimit voivat toimia itseohjautuvasti ja heillä on tarvittavat valtuudet toiminnan mukauttamiseen. (Hoffmann, n.d.; Schwaber & Sutherland 2020, 3–4.)



Kuva 4. Scrum-viitekehyksen pilarit ja arvot

Scrumin menestyksekkäs käyttö ohjelmistotuotannossa sisältää arvojen ymmärtämisen ja niihin sitoutumisen. Kun organisaatio ja tiimi toimivat yhteisten arvojen mukaan, Scrumin empiirinen lähestymistapa (läpinäkyvyys, tarkastelu, mukauttaminen) pääsee oikeuksiinsa. Scrum-oppaassa (Schwaber & Sutherland 2020) Scrumin arvoiksi nimetään *sitoutuminen*, *keskittyminen*, *avoimuus*, *kunnioitus* ja *rohkeus*. Sitoutuminen on välttämätöntä ketterän kulttuurin rakentamiselle. Tiimin jäsenet sitoutuvat tavoitteiden saavuttamiseen ja toistensa tukemiseen. Scrumissa tiimi ja jokainen kehittäjä keskittyy ensisijaisesti sprintin työhön, jotta voivat parhaiten edistää yhteistä tavoitetta. Sprintin työt on sovittu etukäteen, joka mahdollistaa keskittymisen olennaiseen. Scrum-tiimi ja sidosryhmät ovat avoimia ongelmien ja haasteiden suhteen tai jos he tarvitsevat apua. Neljäs Scrumin arvo on kunnioitus, joka näkyy tiimin jäsenten välisenä kunnioituksena toisiaan kohtaan. Tiimin jäsenet ja sidosryhmät tunnistavat toistensa vahvuudet ja kyvyt sekä kunnioittavat toistensa ideoita ja näkökulmia. Viimeinen arvo on rohkeus. Tiimin jäsenillä tulee olla rohkeutta tarttua vaikeisiin haasteisiin, kyseenalaistaa vallitsevia käytänteitä, kokeilla uusia asioita, pyytää apua ja tehdä oikeaksi katsomiaan asioita. (Layton, Ostermiller & Kynaston 2022, 23–24; Schwaber & Sutherland 2020, 4; Scrum Alliance n.d.)

3.2 Scrum-tiimi ja roolit

Scrumin käytäntöön paneva voima on Scrum-tiimi, joka koostuu *tuoteomistajasta*, *Scrum Masterista* ja *kehittäjistä*. Tiimi vastaa käyttökelpoisten inkrementtien, ohjelmiston osien, tuottamisesta jokaisessa sprintissä. Tiimin koko on yleensä enintään kymmenen jäsentä, sillä kommunikaation ja tuottavuuden on huomattu heikentyvän tätä suuremmissa tiimeissä. Jäsenet omaavat kaikki tarvittavat taidot ja tiedot, sillä tiimit ovat monialaisia ja pystyvät kattamaan kaikki tuotteen kehityksessä tarvittavat osa-alueet. Itseohjautuvuus on myös tärkeä osa tiimiä, sillä he päättävät itse, kuka tekee minäkkin työn ja miten. Scrum-tiimi on vastuussa kaikkien tuotteeseen liittyvästä tekemisestä: kehittämisestä, testaamisesta, ylläpidosta ja viestinnästä sidosryhmille. (Layton ym. 2022, 11–12; Schwaber & Sutherland 2020, 5.)

Tuoteomistaja on aina yksi henkilö, ei ryhmä. Hän vastaa Scrum-tiimin tekemästä työstä ja siitä, että tiimi tuottaa mahdollisimman suuren arvon. Tuoteomistajan rooli on kaksisuuntainen; hän luo ja viestii tuotteen tavoitteen kehitystiimille sekä toimii aktiivisessa vuorovaikutuksessa sidosryhmien kanssa. Tuoteomistaja on vastuussa myös tuotteen kehitysjonosta, sen valmistelusta ja priorisoinnista. Tuotteen kehitysjonon on oltava läpinäkyvä, saatavilla ja ymmärrettävissä kaikille osapuolille. Jos sidosryhmät haluavat muuttaa tuotteen suuntaa, muutokset kehitysjonolle tapahtuvat tuoteomistajan kautta. (Layton, ym. 2022, 32–33; Schwaber & Sutherland 2020, 5.)

Scrum Masterin rooli on tukea kehitystiimiä sekä organisaatiota Scrumin teorian ja käytäntöjen ymmärtämisessä sekä varmistaa, että tiimissä toimitaan Scrumin periaatteiden mukaisesti. Scrum Master palvelee työssään Scrum-tiimiä, tuoteomistajaa ja organisaatiota. Hän huolehtii Scrum tapahtumien toteutumisesta, fasilitoinnista ja siitä, että tapahtumat ovat tuottavia ja pysyvät sovitussa aikarajoissa. Lisäksi Scrum Master auttaa ja tukee tiimiä tehokkaaseen työskentelyyn poistamalla esteitä ja varmistamalla tiimin jäsenten keskittymisen työhön sekä valmentamalla tiimiä itseohjautuvuuteen ja monialaisuuteen. (Schwaber & Sutherland 2020, 6.)

Scrum Master tukee tuoteomistajaa työssään auttamalla tuotteen tavoitteen ja kehitysjonon määrittelyssä ja hallinnassa. Tarvittaessa hän fasilitoi myös sidosryhmien välistä yhteistyötä. Organisaatiolle Scrum Master tarjoaa valmennusta ja tukea Scrum-viitekehyksen käyttöönotossa ja toteutuksessa. Hän auttaa organisaatiota ymmärtämään, miten havaintoihin perustuva työtapo voi tukea toiminnan tehostamista. Lisäksi hän poistaa esteitä sidosryhmien ja Scrum-tiimien välillä. (Schwaber & Sutherland 2020, 6–7.)

Kehittäjät ovat Scrum-tiimin ydin ja heillä on kaikki tarvittava osaaminen tai kyky oppia uutta, jotta he voivat tuottaa käyttökelpoisen inkrementin jokaisessa sprintissä. Kehittäjät vastaavat sprintin suunnitelman, eli sprintin kehitysjonon, tekemisestä ja tuotteen kehittämisestä sekä laadusta sprintin aikana. (Schwaber & Sutherland 2020, 5.)

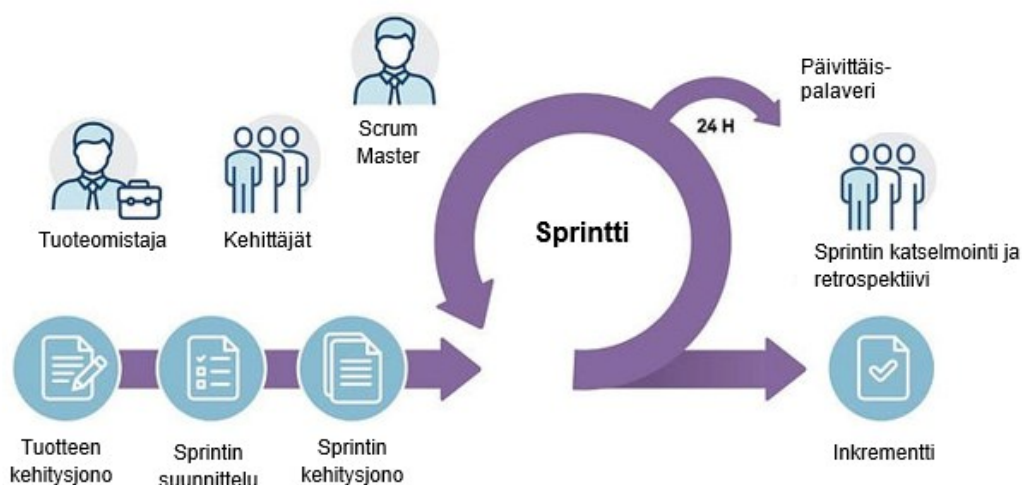
3.3 Scrumin tuotokset ja tapahtumat

Scrumissa on kolme tuotosta: *tuotteen kehitysjono*, *sprintin kehitysjono* ja *inkrementti* (kuva 5), jotka mahdollistavat läpinäkyvyyden tiimille ja sidosryhmille. Tuotteen kehitysjono perustuu tuotteen tavoitteeseen ja se on tuoteomistajan tärkeysjärjestykseen asettama lista työtehtävistä, jotka tähtäävät tuotteen parantamiseen. Tuotteen kehitysjono jalostuu jatkuvasti, kun Scrum-tiimi pilkkoo ja tarkentaa kehitysjonon tehtäviä kokonaisuuksiksi, jotka voidaan toteuttaa yhden sprintin aikana. Näitä tuotteen kehitysjonon jalostamiseen liittyviä palaverieita ei tunnisteta Scrum-oppaassa yhdeksi Scrumin viidestä tapahtumasta, sillä niille ei ole määritelty aikarajaa, toistuvuutta tai läsnäolijoita. Sen sijaan jalostaminen on jatkuvaa toimintaa, jossa työtehtävien kuvauksiin lisätään yksityiskohtia tai pohditaan kehitysjonon kohteiden järjestystä. (Bancroft-Connors n.d.; Schwaber & Sutherland 2020, 10–11.)

Sprintin kehitysjono muodostuu sprintin tavoitteesta ja tuotteen kehitysjonosta valituista työtehtävistä, jotka tiimi aikoo toteuttaa sprintin aikana. Sprintin kehitysjono on kehittäjien sprinttisuunnitelma, joka voi tarkentua sprintin aikana, kun kehittäjät oppivat lisää. Sprintin tavoitteen ei kuitenkaan tulisi muuttua sprintin aikana. Jos sprintin tavoitetta halutaan muuttaa, tulee tästä sopia tuoteomistajan kanssa. Inkrementti on sprintin lopuksi syntyvä valmis, lisäarvoa tuottava ja käyttökelpoi-

nen osa tuotetta. Yhden sprintin aikana voidaan tuottaa useampi inkrementti. Inkrementti ei ole erillinen tuote, vaan osa kokonaisuutta, joka rakentuu aiempien inkrementtien päälle. (Scrum Alliance n.d.; Schwaber & Sutherland 2020, 10–11.)

Scrumissa on viisi tapahtumaa (kuva 5), jotka luovat mahdollisuuden empiirisille toiminnalle; tarkastelulle ja mukauttamiselle. Scrumin tapahtumat luovat säännöllisyyttä ja vähentävät ulkopuolisten palaverien tarvetta. Jos Scrumin keskeisiä palavereja poistetaan, vähenee Scrumista saatu hyöty tai tekee siitä jopa kokonaan hyödyttämän. (Schwaber & Sutherland 2020, 7.)



Kuva 5. Scrum-viitekehyksen tapahtumat ja tuotokset (mukaiillen Novriansyah 2024)

Sprintti on Scrumin tapahtumien ydin, jonka aikana muut Scrumin tapahtumat toteutuvat. Sprintin pituus on yleensä 1–4 viikkoa ja jokaisesta sprintistä syntyy inkrementti. Sprintti alkaa *sprintin suunnittelulla*, jonka tavoitteena on vastata kolmeen kysymykseen: Miksi tämä sprintti on arvokas? Mitä tässä sprintissä voidaan saada valmiiksi? Miten valittu työ saadaan valmiiksi? Koko Scrum-tiimi osallistuu suunnitteluun. Tuoteomistajan vastuulla on varmistaa, että koko tiimi on valmiina keskustelemaan tuotteen kehitysjonon kohteista ja tuotteen tavoitteesta. Tuoteomistaja ehdottaa sprintin suunnittelussa, miten tuotteen arvoa voisi lisätä sprintin aikana, ja seuraavaksi koko tiimi yhdessä määrittelee sprintille tavoitteen. Kehittäjät itse valitsevat tuotteen kehitysjonosta ne työt, jotka uskovat saavansa valmiiksi sprintin aikana sekä määrittelevät, miten tämä työ saadaan tehdyksi. (Scrum Alliance n.d.; Schwaber & Sutherland 2020, 8.) Scrum-oppaassa sprintin suunnittelun pituudeksi suositellaan 8 tuntia, jos sprintti on kuukauden mittainen ja sprintin ollessa lyhyempi, myös suunnittelupalaverin kesto voi olla lyhyempi (Schwaber & Sutherland 2020, 8).

Päivittäispalaveri on 15 minuutin mittainen sprintin jokaisena päivänä samaan aikaan ja samassa paikassa toistuva palaveri, jossa tiimi voi tarkastella sprintin työn etenemistä ja tarvittaessa mukauttaa suunnitelmaa. Kehittäjät saavat itse päättää miten päivittäispalaveri järjestetään, kunhan tapahtuman huomio on sprintin tavoitteen saavuttamisessa ja tulevan työpäivän suunnittelussa. Päivittäispalaverissa kehittäjillä on mahdollisuus keskustella myös työssä ilmenneistä ongelmista ja avun tarpeesta sekä mahdollisuus tehdä nopeita päätöksiä. (Scrum Alliance n.d.; Schwaber & Sutherland 2020, 8.)

Jokaisen sprintin päättää sprintin katselmointi ja retrospektiivi. *Katselmoinnissa* Scrum-tiimi ja sidosryhmät tarkastelevat yhdessä, mitä sprintin aikana kehitettiin ja mitä seuraavaksi kannattaisi tehdä. Katselmoinnin aikana on mahdollista myös muokata tuotteen kehitysjonoa, jos muutostarpeita tunnistetaan. Viimeinen tapahtuma sprintissä on *retrospektiivi*, jossa tiimi tarkastelee kulunutta sprinttiä tiimin toiminnan näkökulmasta ja pohtii keinoja laadun ja tehokkuuden parantamiseksi. Retrospektiivissä voidaan tarkastella esimerkiksi sitä, mikä meni sprintin aikana hyvin, mitä ongelmia kohdattiin tai sitä voidaanko tiimin prosesseja tai työkaluja parantaa. Sprintin katselmoinnin pituudeksi suositellaan neljää tuntia ja retrospektiivin pituudeksi kolmea tuntia, jos sprintti on kuukauden mittainen. Jos sprintti on lyhyempi, niin myös katselmoinnin ja retrospektiivien pituudet ovat lyhyemmät. (Schwaber & Sutherland 2020, 8–9.)

3.4 Scrumin vahvuudet ja heikkoudet

Scrumin vahvuuksia ja heikkouksia verrattuna perinteisiin kehitysmenetelmiin tai toisiin ketteriin menetelmiin on tutkittu paljonkin. Akour, Alshurideh, Al Kurdi ja Hamadneh (2024) toteuttivat kyselytutkimuksen IT- ja teollisuudenalan yrityksissä, joissa Scrum oli aktiivisesti käytössä. Tutkimuksessa Scrumin yhdeksi vahvuudeksi tunnistettiin tiimityön lisääntyminen ja kommunikaation tehostuminen, mikä puolestaan taas vaikutti projektin lopputuloksen positiivisesti. Saman löydöksen ovat tehneet myös Ekechi, Okeke ja Adama (2024) sekä Carvalho ja Mello (2011) tekemissään laajoissa kirjallisuuskatsauksissaan. Scrum-viitekehyksen käytön on tunnistettu lisäävän myös kehitystiimin motivaatiota (Carvalho & Mello 2011).

Myös Scrumin keskeiset elementit, kuten säännölliset sprintit ja selkeät sprintin tavoitteet, parantavat tiimin keskittymistä ja tuottavuutta. Toistuvat katselmoinnit ja iteraatiot auttavat tunnistamaan tuotteen ongelmat nopeasti, mikä tukee laadun parantamista. Päivittäispalaverit ja retrospektiivit kannustavat tiimejä tunnistamaan ja ratkaisemaan ongelmia nopeasti, mikä taas vahvistaa tiimin yhteistyötä ja vastuullisuutta. Scrumin tapahtumat ovat positiivisessa yhteydessä projektin suorituskykymittareihin, mikä vahvistaa viitekehyksen tehokkuuden. (Akour ym. 2024; Ekechi ym. 2024.)

Sidosryhmien kanssa tehtävä yhteistyö ja asiakkaiden tiivis osallistuminen nousivat esille kaikissa aiemmin mainituissa kolmessa tutkimuksessa. Sidosryhmien kanssa tehtävä jatkuva yhteistyö lisää asiakastyytyväisyyttä, sillä tuote vastaa paremmin asiakkaiden tarpeita ja odotuksia. Asiakkaiden osallistuminen projektin jokaisessa vaiheessa mahdollistaa myös varhaisen palautteen ja nopean reagoinnin muuttuneisiin tarpeisiin. (Akour ym. 2024, 163–165; Carvalho & Mello 2011; Ekechi ym. 2024.) Ekechi ym. (2024) tuovat kuitenkin esille sen, että Scrumin käyttöönotto vaatii kattavaa koulutusta ja sitoutumista yrityksen johdolta, jotta Scrum juurtuu osaksi yhteisön kulttuuria ja Scrumista saatavat hyödyt, kuten tuottavuus, laadun parantuminen ja asiakastyytyväisyys, ovat saavutettavissa.

Vaikka Scrum-viitekehyksen käyttämisellä on tunnistettu monia vahvuuksia niin tiimille kuin lopputuotteellekin, liittyy siihen myös heikkouksia. Organisaation ja kehitystiimien siirtyminen Scrumiin perinteisistä kehitysmenetelmistä voi synnyttää muutosvastarintaa. Akour ym. (2024, 163–165) pitävätkin koulutusohjelmia tärkeinä Scrumin onnistumisen kannalta. Koulutusohjelmat voivat tukea tiimin jäseniä tarvittavien taitojen sekä ajattelutavan kehittämisessä, mikä puolestaan auttaa ketterien menetelmien omaksumista. Myös Akifin ja Majeedin (2012) tekemässä tutkimuksessa, jossa selvitettiin kahdessa Scrumia hyödyntävässä yrityksessä Scrumiin liittyviä haasteita, koulutuksen puute

nousi yhdeksi aiheeksi. Tutkimuksen mukaan koulutusta saaneet tiimit olivat paljon parempia itseohjautuvuudessa ja koulutuksen myötä tiimit ymmärsivät paremmin Scrumin toiminnan ja sen hyödyt omalle toiminnalle.

Scrum-tiimien tehokkuuteen vaikuttavat tiimin koko, viestintä ja roolien selkeys. Wonohardjon, Sunaryon, Sudiyonon, Suranathan ja Suharjiton (2019) tekemässä tutkimuksessa Scrum-tiimien koot vaihtelivat 7–19 jäsenen välillä. Liian suurissa tiimeissä tehokkuus saattoi laskea ja roolit olivat epäselviä. Ratkaisuna liian suurille tiimeille Wonohardjon ym. (2019) ehdottavat Scrum of Scrums -menetelmää, jossa iso tiimi jaetaan useammaksi tiimiksi ja kukin näistä tiimeistä tekee yhtä osaa projektista. Tällöin tiimien välinen riippuvuuksien hallinta on kuitenkin erityisen tärkeää. Myös Akour ym. (2024) toivat esille tutkimuksessaan, että roolien epäselvyys hidasti kehitystyötä. Erityisesti tuotemistajan ja Scrum Masterin roolit ja vastuut tulisi määritellä tarkasti. Cho (2008) nostaa artikkelissaan esiin riskin, että yksittäisillä tiimin jäsenillä saattaa olla kriittistä osaamista, jota muut eivät hallitse. Jos tällainen jäsen poistuu tiimistä, osaamisen tason palautuminen voi viedä kuukausia. Lisäksi tiimien välinen kommunikaatio voi olla puutteellista, mikä saattaa johtaa tuplatyöhön ja resurssien hukkaamiseen.

Scrumin tapahtumat voivat olla viitekehyksen vahvuuden lisäksi myös sen heikkous. Chon (2008) tutkimukseen osallistuneet henkilöt kokivat, että sprinttien suunnittelu ja katselmointi olivat tehottomia ja päivittäispalaverien aikarajoissa ei pysytty, koska niissä keskusteltiin palaveriin kuulumattomista aiheista. Samankaltaisia havaintoja tehtiin myös Akifin ja Majeedin (2012) tutkimuksessa, jossa monet pitivät kokouksia ajanhukkana, jos ne eivät liittyneet heidän työtehtäviinsä. Tehokkuuden parantamiseksi kokouksiin tulisi kutsua vain ne henkilöt, jotka ovat suoraan osallisena käsiteltävässä aiheessa. Muut tiimin jäsenet voivat saada kokouksessa käsitellyt tiedot tiivistettynä jälkikäteen.

Sprintin kesto, dokumentaatio ja asiakkaiden osallistuminen vaikuttavat tuotteen laatuun. Sprintin keston määrittäminen on keskeistä Scrumin tehokkuuden kannalta. Sopiva kesto riippuu vaatimusten muuttumisnopeudesta ja tiimin kyvykkyydestä tuottaa toiminnallisuuksia. Liian lyhyet sprintit voivat lisätä ongelmia, kun taas liian pitkät sprintit voivat vaikeuttaa tiimin edistymisen seuranta. (Akif & Majeed 2012.) Scrumin lyhyet sprintit ja tahtotila julkaista nopeasti voivat vaikuttaa myös tuotteen laatuun, sillä testaamiselle ja laadun varmistamiselle ei varata tarpeeksi aikaa sprintin suunnittelussa. Lisäksi Scrumin vähäinen dokumentaatio on haaste uusille tai kokemattomille työntekijöille, sillä asioiden selvittäminen vie paljon aikaa, jos kunnollinen dokumentaatio puuttuu. Scrumin teorian mukaan asiakkaiden tulisi osallistua aktiivisesti projektin kaikkiin vaiheisiin. Käytännössä tutkimukset kuitenkin osoittavat, ettei asiakasosallistuminen aina toteudu odotetusti. Esimerkiksi tuotteelle asetetut vaatimukset voivat jäädä puutteellisiksi, tai asiakkaalta ei saada riittävästi palautetta projektin aikana. (Akif & Majeed 2012; Cho 2008.)

4 TUTKIMUKSEN TOTEUTUS

4.1 Tutkimuksen tavoite ja tutkimuskysymykset

Tämän opinnäytetyön tilaajana toimi Ponsse Oyj. Opinnäytetyö on saanut alkunsa omasta kiinnostuksestani Scrum-viitekehystä kohtaan, sillä olen työskennellyt Ponsella Scrum Masterina. Scrum on myös keskeinen osa Ponssen ohjelmistokehitystä. Ponsella Scrumia on toteutettu vuodesta 2022 lähtien ja sen käyttöönotto on edennyt tiimeissä eri tahtiin.

Tutkimuksen tavoitteena on toimintatutkimuksen avulla selvittää, miten hyvin Ponssen digitaalisten palveluiden kehitystiimien jäsenet tuntevat Scrumin teorian: pilarit ja arvot, sekä miten nämä toteutuvat tiimien työskentelyssä. Toisena tavoitteena on selvittää, voidaanko kolmen sprintin mittaisen intervention avulla vahvistaa yhden tiimin Scrum-osaamista. Lisäksi tutkimuksen tavoitteena on kartoittaa, miten kehitystiimit kokevat Ponssen nykyisten Scrum-käytäntöjen tukevan tiimien työskentelyä sekä mitkä ovat nykyisten käytäntöjen vahvuudet ja heikkoudet.

Tutkimus ei ainoastaan arvioi nykytilaa, vaan se pyrkii myös tarjoamaan kehitysehdotuksia, joita voidaan hyödyntää Ponssen Scrum-toimintamallin ja tiimien työskentelytapojen parantamisessa. Tämä tekee tutkimuksesta paitsi ajankohtaisen myös käytännönläheisen ja organisaation kehittämiseen tähtäävän kokonaisuuden.

Tutkimuskysymykset:

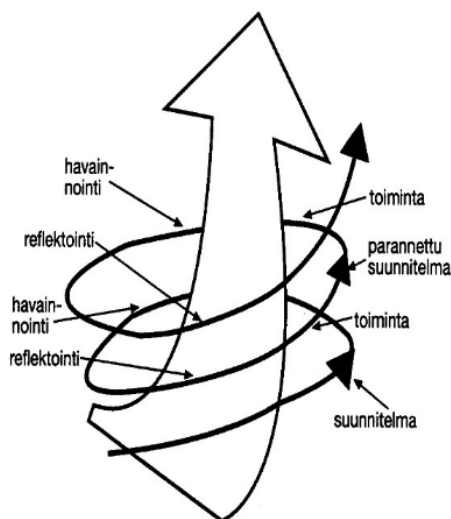
1. Miten hyvin Ponssen kehitystiimit tuntevat Scrumin teorian ja miten Scrumin pilarit ja arvot toteutuvat tiimien arjessa?
2. Voidaanko lyhyellä interventiolla vahvistaa yhden kehitystiimin jäsenten Scrum-osaamista?
3. Miten kehitystiimin jäsenet kokevat Ponssen nykyisten Scrum-käytäntöjen tukevan heidän työtään?

4.2 Toimintatutkimus tutkimusmenetelmänä

Tämä tutkimus toteutettiin laadullisena tutkimuksena, jossa yhdistyivät sekä tapaustutkimuksen että toimintatutkimuksen piirteet. Tapaustutkimuksessa pyritään muodostamaan mahdollisimman laaja kuva tutkimuskohteesta tutustumalla siihen kokonaisvaltaisesti. Tutkimuskohteena voi olla esimerkiksi organisaatio tai ryhmä, kuten työpaikka tai oppilaitos. Tapauksena voi olla myös esimerkiksi prosessi tai yksittäinen henkilö. (Vuori 2021.) Tässä tutkimuksessa tapauksena voidaan pitää Scrum-viitekehystä. Tutkimuksen tavoitteena oli ymmärtää Scrum-viitekehityksen nykyistä tilannetta Ponssen kehitystiimeissä ja miten Scrum tukee tiimien työskentelyä. Tutkimuksessa oli myös toimintatutkimuksen piirteitä, sillä toimin itse tutkijana ja toteutin intervention yhdelle kehitystiimeistä samalla, kun työskentelin Ponsella.

Toimintatutkimukselle ei ole yhtä selkeää määritelmää, mutta ominaista sille on tutkimuksen ja käytännön samanaikaisuus. Toimintatutkimuksen käytännön näkökulmasta keskeistä on uuden oppiminen ja oivaltaminen, jotka johtavat muutokseen toiminnassa. Näiden muutosten myötä myös tutkimuksen kohteena olevan teoreettisen ymmärryksen tulisi syventyä. (Juuti & Puusa 2020.) Toiminta-

tutkimusta hyödynnetään monilla tutkimus- ja tieteenaloilla, erityisesti sitä on käytetty kasvatustieteissä ja työelämäntutkimuksessa, sosiaalityön ja hoitotieteen sekä yhdyskuntien tutkimisessa. (Jyrkämä 2021). Lewin on kehittänyt toimintatutkimuksen perusmallin (kuva 6), jossa tutkimus etenee spiraalinomaisena prosessina. Tutkimus alkaa ongelman havaitsemisesta, tunnistamisesta ja kartoittamisesta. Ongelman pohjalta tehdään tutkimus- ja muutossuunnitelma, jota lähdetään toteuttamaan. Suunnitelman toteutumista arvioidaan ja tämän pohjalta suunnitelmaa muutetaan ja täsmennetään. Tämän jälkeen tarkennettua suunnitelmaa toteutetaan, seurataan ja arvioidaan edelleen, kunnes tavoitellut muutokset on saavutettu tai todetaan, että haluttuun lopputulokseen ei päästä. (Jyrkämä 2021.)



Kuva 6. Toimintatutkimuksen perusmalli (Jyrkämä 2021)

Tässä tutkimuksessa sovellettiin toimintatutkimuksen piirteitä, sillä yhtenä tavoitteena oli syventää kehitystiimin asiantuntemusta Scrumista kolmen sprintin mittaisen intervention aikana. Tutkijana osallistuin havainnoijan roolissa tiimin sprintin tapahtumiin, kuten päivittäiskatselmukseen ja sprintin suunnitteluun. Näiden havaintojen perusteella suunnittelin ja toteutin tiimille koulutuskokonaisuuden. Tätä koulutuskokonaisuutta kutsutaan tässä tutkimuksessa myös interventioksi. Interventio koostui kolmesta retrospektiivistä, joissa käytiin läpi Scrumin teoria ja tapahtumat sekä keskusteltiin tiimin toiminnasta. Jokaisen retrospektiivin jälkeen havainnoin taas tiimiä ja havaintojen perusteella muokasin koulutussuunnitelman sisältöä tiimille paremmin soveltuvaksi. Koska tämä tutkimus oli opinnäytetyö, intervention pituudeksi sovittiin kolme sprinttiä, muuten tutkimuksen laajuus olisi kasvanut liian suureksi.

4.3 Tutkimusaineiston keruu: kyselyt ja interventio

Toimintatutkimuksessa käytetään monenlaisia aineistoja, kuten dokumentteja, havainnointia, haastatteluja ja kyselyjä (Jyrkämä 2021). Tämän tutkimuksen aineiston keruu tapahtui kolmessa vaiheessa ja menetelminä käytin kyselylomaketta, havainnointia ja interventiota. Tutkimuksen ensimmäisessä vaiheessa loin kyselylomakkeen Webropol.fi-sivustolla. Kyselylomakkeen alussa oli taustakysymyksiä vastaajasta ja tämän jälkeen Likert-asteikollisia kysymyksiä Scrumin pilareista, arvoista ja tapahtumista. Kyselylomakkeen lopussa oli vielä kaksi avointa kysymystä Ponsen nykyisistä Scrum-käytännöistä. Lähetin saatekirjeen tutkimuksesta ja kyselylomakkeen linkin kaikille digi-

taalisten palveluiden kehitystiimien jäsenille, jotka käyttävät Scrum-viitekehystä arjessaan. Sähköpostilistat sain tiimien Scrum Mastereilta ja lähetin kyselylomakkeen yhteensä viidelle eri tiimille. Kyselylomake oli avoinna 13.–31.1.2025 välisen ajan.

Tutkimuksen toisessa vaiheessa toteutin kolmen sprintin mittaisen intervention yhdelle kehitystiimeistä. Koulutusjakson tavoitteena oli vahvistaa tiimin Scrum-osaamista. Toteutin intervention vuoden 2025 tammi-maaliskuun aikana Teams-yhteyden välityksellä. Koulutusmateriaalit (liite 4) jaoin tiimin kanssa Miro.com-sivustolla, joka on virtuaalinen yhteistyöalusta. Intervention aikana osallistuin itse havainnoijan roolissa tiimin Scrum-tapahtumiin, kuten päivittäiskatselmukseen, sprintin suunnitteluun ja katselmointiin. Havainnoinnin avulla sain tietoa tiimin nykyisistä käytännöistä, ja näiden havaintojen ja kyselylomakkeen perusteella suunnittelin koulutuskokonaisuuden, jossa käsiteltiin sekä Scrum-viitekehityksen teoreettista pohjaa että tiimin käytäntöjä. Tutkimuksen kolmannessa vaiheessa interventioon osallistuneet henkilöt täyttivät vielä loppukyselyn. Loppukysely oli muutoin sama kuin tutkimuksen aluksi täytetty kysely, mutta siinä oli myös muutama lisäkysymys liittyen koulutusjaksoon.

4.4 Aineiston käsittely ja analyysi

Tutkimusaineiston analyysitapa valitaan sen perusteella, millaista aineistoa tutkimuksessa on kerätty ja mitä tietoa siitä halutaan saada esille. Kun ensimmäisen kyselylomakkeen vastausaika oli sulkeutunut, aloin tutustua kerättyyn aineistoon ja pohtia, miten aineistoa tulisi käsitellä. Ensimmäiseen kyselyyn vastasi 28 henkilöä (N=43), eli vastausprosentti oli 65 %.

Analyysin aluksi ryhmittelin vastaajat viiteen tiimiin heidän ilmoittamansa kehitystiimin perusteella ja annoin tiimeille korvaavat nimet vastaajien anonymiteetin parantamiseksi. Raportissa tiimit ovat nimetty seuraavasti: Tiimi A, Tiimi B, Tiimi C, Tiimi D ja Tiimi E. Tässä tutkimuksessa Tiimi A osallistui interventioon, joten tuloksissa käytän nimeä ”Tiimi A ennen” ja ”Tiimi A jälkeen” kuvaamaan tiimin tuloksia ennen ja jälkeen intervention.

Kyselylomakkeessa oli sekä Likert-asteikollisia että avoimia kysymyksiä, minkä vuoksi päädyin käyttämään sekä määrällistä että laadullista analyysiä. Likert-asteikko on järjestysasteikko, mutta se usein mielletään välimatka-asteikoksi, koska vastausvaihtoehtojen väliset etäisyydet tulkitaan samansuuruisiksi. Välimatka-asteikko tarjoaa tutkijalle enemmän tilastollisia analyysimenetelmiä kuin järjestysasteikolliset muuttujat. (Tähtinen, Laakkonen & Broberg 2020, 32–33.) Tutkimusaineisto oli sen verran pieni, että tilastollisten testien tekeminen, kuten t-testin tai varianssianalyysin, tekeminen ei olisi ollut luotettavaa. Tästä syystä päädyin kuvailevaan analyysiin sekä tunnuslukujen ja visuaalisten esitystapojen hyödyntämiseen tulosten esittämisessä. Erilaisten kuvaajien tekeminen ja tunnuslukujen laskeminen Webropol.fi-sivustolla oli helppoa ja myös kahden eri kyselyn tulosten yhdistäminen olisi ollut mahdollista. Itse käytin kuitenkin Exceliä aineistojen yhdistämiseen.

Kyselylomakkeen avoimille kysymyksille tein laadullista sisällönanalyysiä. Aineistolähtöisen sisällönanalyysin ensimmäisessä vaiheessa aineistolle esitetään tutkimuskysymys ja poimitaan aineistosta kaikki analyysiyksikön mukaiset ilmaisut. Tässä vaiheessa aineistoa myös pelkistetään, mutta on tärkeää varmistaa, ettei sisältö muutu tai tutkija tee omia tulkintoja. (Elo, Kajula, Tohmola & Kärräinen 2022, 119–121.) Aloitin analyysin kokoamalla kaikki avoimet vastaukset Ponssin nykyisten Scrum-käytäntöjen vahvuuksista ja heikkouksista kahteen tiedostoon. Poistin aineistosta myös tyhjät vastaukset tai muut kommentit, jotka eivät liittyneet itse tutkimuskysymykseen.

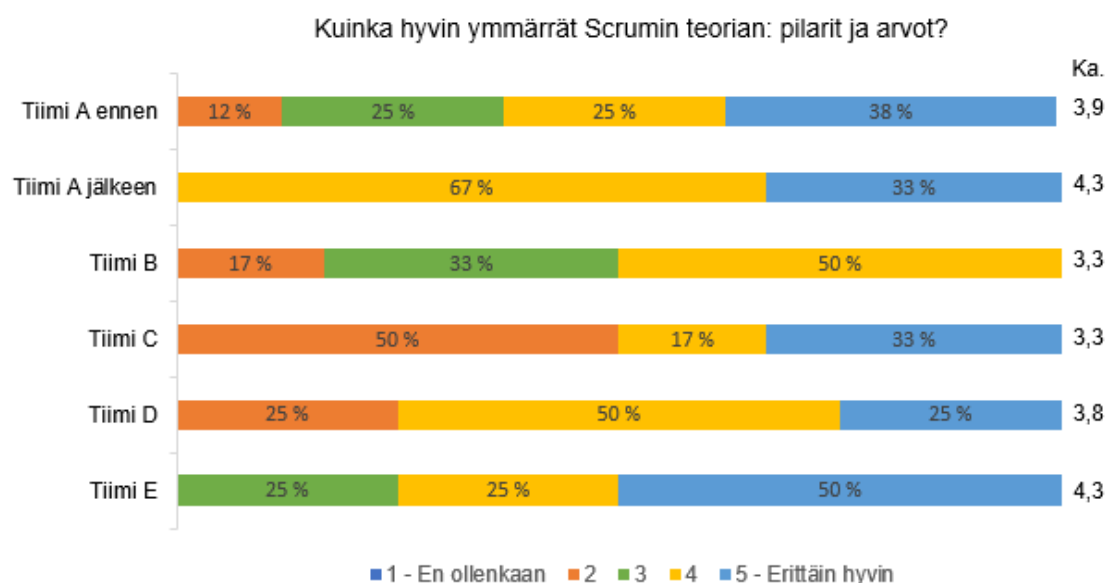
Kun aineisto on pelkistetty, seuraava vaihe on sen koodaus, jossa vastauksia jäsenellään esimerkiksi alleviivauksin, värikoodein tai muilla merkinnöillä. Sisällönanalyysin viimeisessä vaiheessa aineisto luokitellaan ja ryhmitellään etsimällä samankaltaisuuksia ja eroavaisuuksia. Näitä yhdistämällä muodostetaan alaluokkia, jotka edelleen voidaan yhdistää yläluokiksi ja lopulta pääluokiksi. (Elo ym. 2022, 119–121.) Tässä tutkimuksessa avointen vastausten määrä (n=28) oli melko pieni, joten pelkistetyn aineiston luokittelu oli melko nopeaa. Aineistosta alkoi hahmottua erilaisia luokkia, joita yhdisti samanlaiset kokemukset ja näkökulmat. Päädyin analyysissä kaksitasoiseen luokitteluun, joiden avulla pystyin kuvaamaan hyvin Ponssen nykyisten Scrum-käytäntöjen vahvuuksia ja heikkouksia.

5 TUTKIMUKSEN TULOKSET

5.1 Scrum-viitekehys Ponsen kehitystiimeissä

Kyselyyn vastasi 28 henkilöä viidestä kehitystiimistä (N=43), mikä vastaa 65 %:n vastausprosenttia. Vastaajien työurat vaihtelivat yhdestä vuodesta 27 vuoteen, tyypillinen työuran pituus oli kymmenen vuotta. Kokemus Scrum-viitekehityksen mukaan työskentelystä vaihteli kuudesta kuukaudesta 14 vuoteen, ja tyypillisin kokemus oli 4,25 vuotta. Puolella vastaajista (n=14) oli kokemusta myös muista viitekehityksistä, kuten Kanban ja SAFe tai hybridimallin mukaisesta ohjelmistokehityksestä. Ensimmäiseen kyselyyn mennessä vastaajista 61 % (n=17) oli osallistunut koulutukseen Scrumista. Useat olivat osallistuneet kansainvälisesti tunnustettuihin sertifiointikoulutuksiin, kuten Certified Scrum Master (CSM) ja Certified Scrum Product Owner (CSPO). Lisäksi monet vastaajista olivat osallistuneet Ponsen sisäisiin koulutuksiin tai muiden koulutuspalveluita tarjoavien organisaatioiden järjestämiin koulutuksiin.

Seuraavaksi käydään läpi, miten eri kehitystiimit A, B, C, D ja E tuntevat Scrum-viitekehityksen peruspilarit ja arvot, sekä miten nämä toteutuvat kehitystiimien arjessa. Alla olevissa kuvissa (kuvat 7–11) ja taulukoissa (taulukot 1–4) on esitetty jokaisen kehitystiimin tulokset sekä lisäksi interventioon osallistuneen Tiimi A:n intervention jälkeiset tulokset. Intervention vaikutusta Tiimin A:n tuloksiin tarkastellaan sekä tässä luvussa että seuraavassa luvussa *5.2 Intervention vaikutus kehitystiimin Scrum-osaamiseen*.



Kuva 7. Scrum-viitekehityksen pilarien ja arvojen tuntemus kehitystiimeissä

Kyselyssä vastaajat arvioivat ymmärrystään Scrumin pilareista ja arvoista asteikolla 1–5, jossa 1 tarkoitti "en ollenkaan" ja 5 "erittäin hyvin" (kuva 7). Tiimi A:n tilanne ennen interventiota osoitti, että ymmärrys Scrum-teoriasta oli melko hyvä ja intervention jälkeen osaaminen nousi selvästi. Ennen koulutusta Tiimin A:n keskiarvo oli 3,9 ja koulutuksen jälkeen se nousi arvoon 4,3. Suurin muutos tapahtui asteikon arvoissa 2 ja 3. Ennen koulutusta 12 % tiimiläisistä arvioi ymmärryksensä tasolle

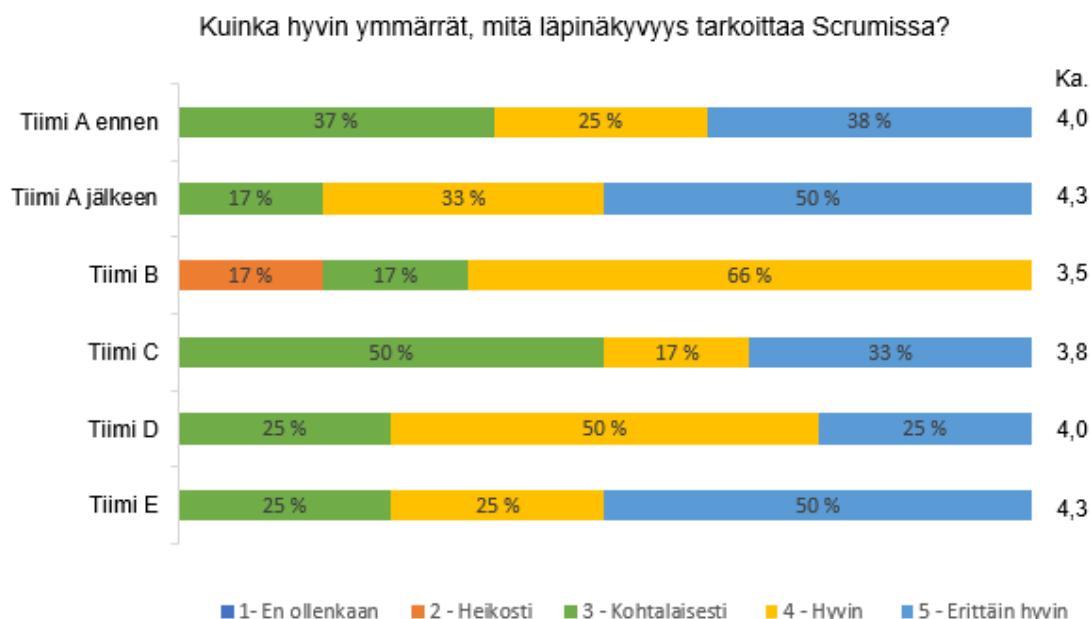
2, mikä viittaa heikkoon osaamiseen Scrumin teoriasta. Koulutuksen jälkeen tämä osuus katosi kokonaan ja myös asteikon arvon 3 antaneet henkilöt arvioivat loppukyselyssä osaamisensa lisääntyneen.

Muiden tiimien arviot Scrumin teorian osaamisesta vaihtelivat. Tiimi B ja Tiimi C saavuttivat molemmat keskiarvon 3,3, mikä osoittaa, että ymmärryksen taso oli kohtalaista. Tiimi D:n keskiarvo oli 3,8, mikä on keskitasoa korkeampi ja Tiimi E saavutti korkeimman keskiarvon 4,3, mikä osoittaa, että tässä tiimissä Scrumin teorian ymmärrys oli hyvä. Koska kaikkien tiimien keskiarvot olivat yli 3, voidaan todeta, että Scrum-teorian perusasiat olivat suhteellisen hyvin hallussa, mutta tiimin sisällä yksittäisten vastaajien osaaminen vaihteli arvoasteikon 2–5 välillä. Tämä osoittaa, että Scrumin teorian ymmärtämisessä on laajaa vaihtelua yksittäisten vastaajien välillä.

Läpinäkyvyys

Kuva 8 esittää kehitystiimien ymmärrystä läpinäkyvyyden käsitteestä Scrum-viitekehityksessä. Kyselyyn osallistuneet tiimit arvioivat omaa käsitystään asteikolla 1–5, jossa 1 tarkoittaa ”en ollenkaan” ja 5 ”erittäin hyvin”. Tulosten perusteella voidaan havaita merkittäviä eroja tiimien ja yksittäisten vastaajien välillä, sekä intervention positiivinen vaikutus Tiimi A:n tuloksiin.

Tiimi A:n tulokset osoittivat, että ennen interventiota tiimin keskiarvo läpinäkyvyyden käsitteestä oli 4,0 ja intervention jälkeen se nousi arvoon 4,3. Intervention jälkeen 50 % tiimiläisistä arvioi ymmärryksensä olevan erittäin hyvä, kun ennen koulutusta vastaajista 38 % valitsi vaihtoehdon 5. Muiden tiimien osalta vaihtelua esiintyi läpinäkyvyyden ymmärtämisessä. Matalin keskiarvo oli Tiimi B:llä (ka. 3,5), ja heidän joukossaan oli myös heikoin osaaminen läpinäkyvyyden pilarista, sillä 17 % vastaajista antoi arvion 2 ”heikosti”. Tiimi C ja Tiimi D saivat keskiarvot 3,8 ja 4,0, ja heidän tuloksissaan korostui kohtalainen tai hyvä ymmärrys. Tiimi E saavutti korkeimman tuloksen (ka. 4,3), ja heidän joukostaan 50 % arvioi ymmärryksensä läpinäkyvyydestä olevan erittäin hyvä.



Kuva 8. Läpinäkyvyyden ymmärtäminen kehitystiimeissä

Taulukossa 1 on esitetty tiimien arviot kolmesta kysymyksestä, jotka liittyvät läpinäkyvyyden toteutumiseen tai näkymiseen kehitystiimien arjessa. Arvioinnissa käytettiin samaa Likert-asteikkoa arvoasteikolla 1–5, jossa 1 tarkoittaa "ei ollenkaan" ja 5 "erittäin hyvin". Tiimien tulokset on esitetty keskiarvoina taulukossa. Tiimien käytännöt, kuten päivittäispalaverit ja sprintin katselmuksot tukivat läpinäkyvyyttä kohtalaisesti tai hyvin (ka. 3,3–4,5) kaikissa tiimeissä. Intervention vaikutus Tiimi A:n keskiarvoon oli positiivinen ja se nousi 3,9:stä 4,2:een. Tämä viittaa siihen, että tiimi paransi käytäntöjään ja ymmärsi niiden vaikutuksen läpinäkyvyyteen entistä paremmin.

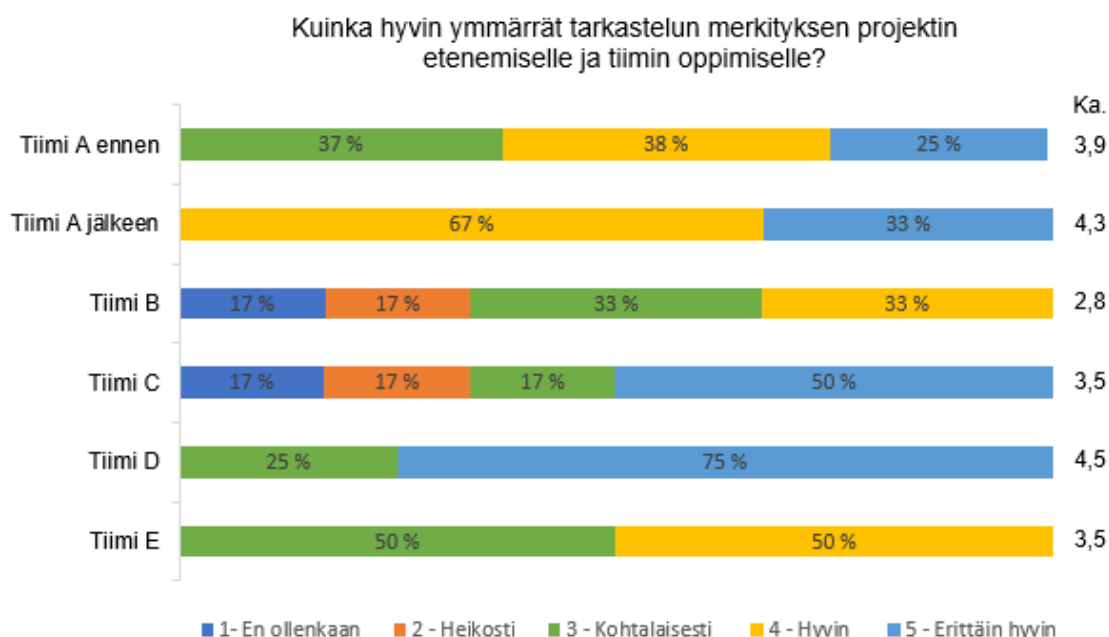
Sen sijaan yrityksen vision ja strategian ymmärtämistä (ka. 2,8–3,8) sekä sidosryhmien välistä viestintää (ka. 2,8–3,8) koskevat keskiarvot jäivät matalammiksi lähes kaikissa tiimeissä. Lisäksi intervention jälkeen tehdyssä kyselyssä Tiimi A:n keskiarvot näillä osa-alueilla laskivat. Tulokset viittavat siihen, että organisaation ja sidosryhmien välinen viestintä kaipaa edelleen kehittämistä. Jotta läpinäkyvyys vahvistuisi koko yrityksen tasolla, on tärkeää panostaa laadukkaaseen viestintään ja tiedon jakamiseen eri sidosryhmien välillä.

Taulukko 1. Läpinäkyvyyden toteutuminen kehitystiimeissä (keskiarvot Likert-asteikolla 1–5, jossa 1 tarkoittaa "ei ollenkaan" ja 5 "erittäin hyvin")

	Tiimi A ennen	Tiimi A jälkeen	Tiimi B	Tiimi C	Tiimi D	Tiimi E
Tukevatko tiiminne käytännöt, esim. päivittäispalaverit, sprinttikatselmuksot, läpinäkyvyyttä tehokkaasti?	3,9	4,2	3,3	3,8	4,5	3,3
Kuinka selkeästi ymmärrät yrityksen vision ja strategian?	3,8	3,5	2,8	3,0	2,5	3,3
Koetko, että tiimisi ja sidosryhmien välinen viestintä on avointa, selkeää ja sujuvaa?	3,8	3,3	3,2	3,0	2,8	3,3

Tarkastelu

Kehitystiimien ymmärrys tarkastelun merkityksestä projektin etenemiselle ja tiimin oppimiselle (kuva 9) vaihteli suuresti tiimien välillä. Tiimi D saavutti korkeimman keskiarvon 4,5, kun taas Tiimi B:n keskiarvo jäi heikon ja kohtalaisen välille (ka. 2,8). Tiimi B:ssä ja Tiimi C:ssä myös 17 % vastaajista koki, ettei ymmärtänyt lainkaan tarkastelun merkitystä. Tiimi A:n tulokset ennen ja jälkeen intervention osoittavat selkeää kehitystä tarkastelun ymmärtämisessä. Ennen koulutusta suurin osa tiimiläisistä arvioi ymmärryksensä kohtalaiseksi (38 %) tai hyväksi (25 %). Koulutuksen jälkeen jakauma muuttui huomattavasti: 67 % arvioi ymmärtävänsä tarkastelun merkityksen hyvin ja 33 % erittäin hyvin, mikä osoittaa tiimin osaamisen kehittymistä.



Kuva 9. Tarkastelun ymmärtäminen kehitystiimeissä

Tarkastelun toteutumista kehitystiimeissä arvioitiin kolmella kysymyksellä (taulukko 2): tiimin käytännöt tarkastelun tukena, edistymisen arviointi ja ongelmien tunnistaminen sekä reagointikyky havaittuihin ongelmiin ja parannusehdotuksiin. Tulokset osoittavat, että interventio vaikutti selvästi Tiimi A:n toimintaan kaikissa kolmessa kohdassa. Ennen koulutusta Tiimi A:n käytännöt tukivat tarkastelua kohtalaisesti (ka. 3,0), mutta koulutuksen jälkeen keskiarvo nousi 3,8:aan, mikä lähestyy "hyvin" arvosanaa. Edistymisen arvioinnissa ja ongelmien tunnistamisessa keskiarvo parani 3,4:stä 3,8:aan, ja reagointikyky havaittuihin ongelmiin nousi 3,6:sta 4,0:aan.

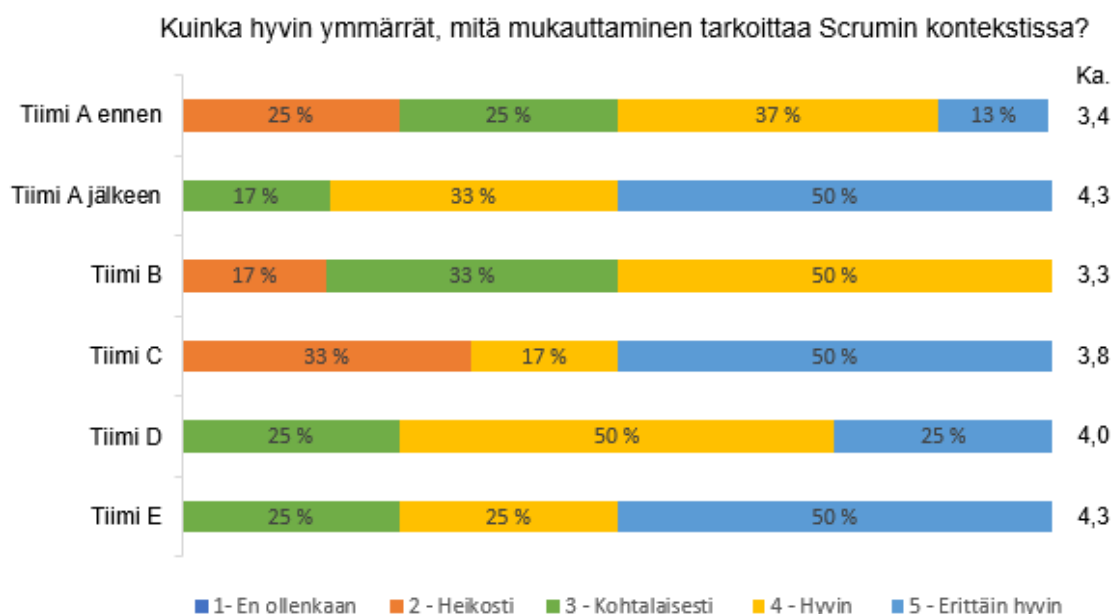
Tiimi B:n ja Tiimi E:n alhaisimmat keskiarvot kaikissa kolmessa kysymyksessä osoittavat, että tarkastelun toteutuminen on tiimeissä heikkoa tai kohtalaista. Tämä voi tarkoittaa sitä, että tiimit eivät ole omaksuneet systemaattisia tarkastelukäytäntöjä tai että käytössä olevat menetelmät eivät tue edistymisen arviointia ja ongelmiin reagoimista riittävän hyvin. Tiimi D erottui edukseen sekä tarkastelun ymmärtämisessä että toteutumisessa, mikä näkyy korkeina keskiarvoina.

Taulukko 2. Tarkastelun toteutuminen kehitystiimeissä (keskiarvot Likert-asteikolla 1–5, jossa 1 tarkoittaa "ei ollenkaan" ja 5 "erittäin hyvin")

	Tiimi A ennen	Tiimi A jälkeen	Tiimi B	Tiimi C	Tiimi D	Tiimi E
Tukevatko tiiminne käytännöt järjestelmällistä tarkastelua?	3,0	3,8	2,5	3,3	4,5	2,8
Kuinka hyvin tiiminne arvioi edistymistään ja tunnistaa ongelmia?	3,4	3,8	3,3	4,0	4,3	3,3
Kuinka nopeasti ja tehokkaasti tiiminne reagoi tarkastelussa havaittuihin ongelmiin tai parannusalueisiin?	3,6	4,0	3,2	3,8	3,8	3,0

Mukauttaminen

Mukauttamisen ymmärtäminen Scrumin kontekstissa (kuva 10) vaihteli eri tiimien välillä samalla tavoin kuin läpinäkyvyyden ja tarkastelun ymmärtäminen. Matalimmat keskiarvot saivat Tiimi A ennen interventiota (ka. 3,4) ja Tiimi B (ka. 3,3), joissa 50 % vastaajista ymmärsi mukauttamisen merkityksen heikosti tai kohtalaisesti. Parhaiten mukauttaminen ymmärrettiin tiimeissä D (ka. 4,0) ja E (ka. 4,3), joissa 75 % vastaajista koki ymmärtävänsä mukauttamisen hyvin tai erittäin hyvin. Interventiolla oli merkittävä vaikutus Tiimi A:n osaamiseen, sillä koulutuksen jälkeen keskiarvo nousi 4,3:een. Lisäksi ”heikosti” vastanneiden osuus katosi kokonaan ja myös ”kohtalaisesti” vastanneiden määrä väheni.



Kuva 10. Mukauttamisen ymmärtäminen kehitystiimeissä

Kehitystiimien arjessa mukauttamisen toteutumista arvioitiin neljän kysymyksen avulla (taulukko 3). Tulokset osoittavat, että tiimien käytännöt tukivat mukauttamista ja toiminnan parantamista kohtalaisesti tai lähes hyvin. Parhaiten tiimin käytäntöjen koettiin tukevan mukauttamista Tiimissä D (ka. 3,8), kun taas Tiimi C:n ja Tiimi B:n tulokset olivat heikoimpia. Tämä viittaa siihen, että joissain tiimeissä tarkastelua seuraavat toimenpiteet ovat vielä epäselviä tai toteutus ei ole riittävän systemaattista.

Intervention vaikutus näkyy erityisesti Tiimi A:n osalta siinä, kuinka hyvin retrospektiiveissä esiin tuodut parannusehdotukset johtivat konkreettisiin muutoksiin. Ennen koulutusta tiimin arvio oli ”kohtalainen” (ka. 3,0), mutta intervention jälkeen keskiarvo nousi 3,7:ään, mikä voi kertoa siitä, että tiimi on hyödyntänyt retrospektiivejä paremmin koulutuksen jälkeen. Tämä on tärkeä muutos, sillä retrospektiivien arvo on rajallinen, jos niiden pohjalta ei tehdä konkreettisia parannuksia. Tiimi D (ka. 4,0) erottui tuloksista positiivisesti, tämä osoittaa, että tiimi osaa hyödyntää retrospektiivien tarjoaman mahdollisuuden tarkasteluun ja mukauttamiseen.

Tiimien autonomia työskentelytapojen ja prosessien kehittämiseen oli suhteellisen tasainen kaikissa tiimeissä (ka. 3,3–3,9), mutta mielenkiintoista on, että Tiimi A koki autonomiansa hieman vähentyneen koulutuksen jälkeen. Tämä voi viitata siihen, että koulutuksen myötä tiimi on alkanut nähdä organisaation tai muiden sidosryhmien vaikuttavan enemmän tiimiin kuin aiemmin oletettiin. Organisaation tuen merkitys korostuu erityisesti silloin, kun tiimi haluaa tehdä nopeita muutoksia. Tulokset osoittavat, että organisaation tuki on yleisesti ottaen kohtalaista (ka. 3,0–3,5). Tiimi D ja Tiimi E kokivat saavansa eniten tukea, kun taas Tiimi B ja Tiimi C arvioivat tuen olevan heikompaa. Tämä voi tarkoittaa, että organisaation prosessit eivät aina mahdollista nopeaa reagointia tai että päätöksentekoon liittyy viivettä.

Taulukko 3. Mukauttamisen toteutuminen kehitystiimeissä (keskiarvot Likert-asteikolla 1–5, jossa 1 tarkoittaa "ei ollenkaan" ja 5 "erittäin hyvin")

	Tiimi A ennen	Tiimi A jälkeen	Tiimi B	Tiimi C	Tiimi D	Tiimi E
Tukevatko tiiminne käytännöt järjestelmällistä mukauttamista tai parantamista tarkastelun perusteella?	3,4	3,7	3,3	3,2	3,8	3,5
Kuinka hyvin retrospektiiveissä esiin tuodut parannusehdotukset johtavat konkreettisiin muutoksiin?	3,0	3,7	3,0	3,7	4,0	3,3
Koetko, että tiimillänne on riittävästi autonomiaa tehdä tarvittavia muutoksia työskentelytapoihin tai prosesseihin?	3,9	3,7	3,3	3,3	3,3	3,8
Kuinka hyvin organisaationne tukee tiimiänne tekemään nopeita muutoksia, kun mukauttamista tarvitaan?	3,3	3,3	3,0	3,0	3,5	3,5

Arvot

Scrumissa arvot - sitoutuminen, rohkeus, keskittyminen, avoimuus ja kunnioitus - ovat keskeinen osa toimintamallia, koska ne luovat pohjan ja kulttuurin, joka tukee tiimien yhteistyötä ja jatkuvaa kehittymistä. Arvoja ja niiden toteutumista kehitystiimien arjessa arvioitiin kymmenen kysymyksen avulla (kuva 11; taulukko 4) ja arvoasteikkona käytettiin samaa Likert-asteikkoa 1–5: 1 – ei lainkaan, 2 – huonosti, 3 – kohtalaisesti, 4 – hyvin, 5 – erittäin hyvin. Tuloksista voidaan todeta, että kaikissa tiimeissä arvot toteutuivat vähintään kohtalaisesti, alin yksittäinen keskiarvo oli 3,2.

Sitoutumista arvioitiin sen perusteella, kuinka hyvin tiimin jäsenet kokivat pysyvänsä sprintin tavoitteissa ja kuinka tärkeänä tavoitteisiin sitoutumista pidettiin. Vastaajat arvioivat, että muut tiimin jäsenet ovat sitoutuneet tiimin tavoitteisiin melko hyvin, keskiarvot vaihtelivat 3,5 ja 4,5 välillä. Mielenkiintoista kuitenkin oli, että Tiimi B arvioi, että tiimin sitoutuminen sprintin tavoitteisiin on vain kohtalaisen tärkeää (ka. 3,2). Toinen mielenkiintoinen havainto oli, että Tiimi A:n keskiarvo sitoutumisen molemmissa kysymyksissä laski intervention jälkeen. Havainnot voivat viitata tarpeeseen keskustelusta tiimin kanssa, jotta varmistetaan, että sprintin tavoitteet ja niihin sitoutuminen olisi kaikille selkeää ja merkityksellistä. Rohkeuden kohdalla arvioitiin, kuinka rohkeasti tiimien jäsenet kokivat pystyvänsä ilmaisemaan mielipiteensä ja toimitaanko tiimeissä rohkaisevasti. Tiimien vastaukset vaihtelivat kohtalaisesta melkein erittäin hyvään. Tiimi A:ssa rohkeus parani hieman tutkimusjakson aikana, mikä voi viitata turvallisemman keskustelukulttuurin kehittämiseen.

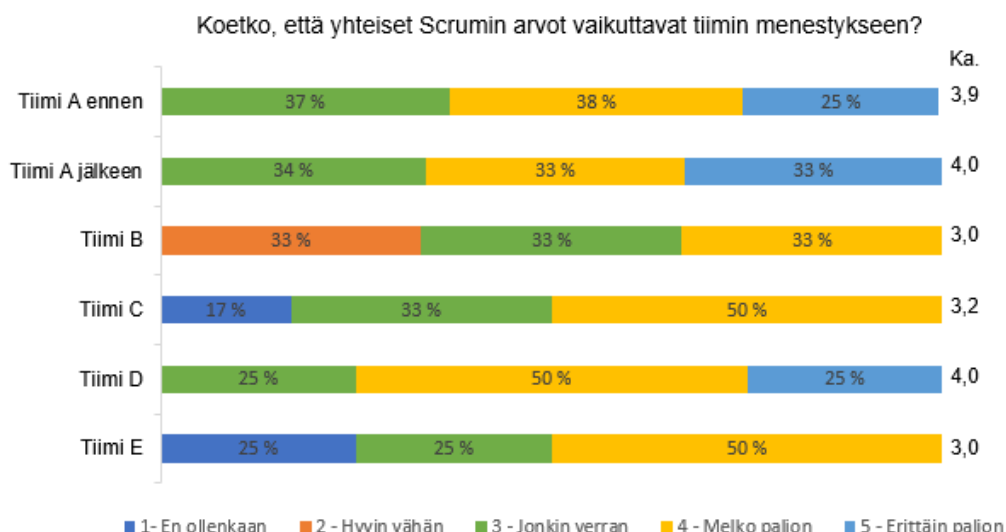
Keskittyminen sai heikoimmat arvioinnit kehitystiimien kesken ja vaikka useimmilla tiimeillä oli melko selkeä käsitys siitä, mihin sprintillä tulee keskittyä (ka. 3,2–4,0), keskeytyi tiimin työskentely ulkoisten

häiriötekijöiden tai lisätehtävien vuoksi melko usein tai usein. Tämä voi kertoa epäselvistä prioriteeteista ja työn vaatimuksista tai epävarmasta sprintin suunnittelusta. Avoimuus on tärkeä osa tiimien toimintakulttuuria, ja tutkimuksen perusteella tiimit kokivat viestintänsä ja tiedonjakonsa olevan avointa. Erityisesti Tiimi D erottui korkealla arvioinnillaan (4,8). Tiimi A:ssa avoimuuden kokemus parani merkittävästi tutkimusjakson aikana (3,6 → 4,5), mikä viittaa myönteiseen kehitykseen tiimin sisäisessä vuorovaikutuksessa ja läpinäkyvyydessä. Parhaimmat keskiarvot kaikista arvoista saavutettiin kunnioituksessa (ka. 4,2–5,0), mikä viittaa vahvaan keskinäiseen arvostukseen ja luottamukseen tiimien sisällä.

Taulukko 4. Arvojen toteutuminen kehitystiimeissä (keskiarvot Likert-asteikolla 1–5, jossa 1 tarkoittaa "ei lainkaan" ja 5 "erittäin hyvin")

Arvot		Tiimi A ennen	Tiimi A jälkeen	Tiimi B	Tiimi C	Tiimi D	Tiimi E
Sitoutuminen	Koetko, että kaikki tiimin jäsenet ovat sitoutuneet sprintin aikana sovittuihin tavoitteisiin?	3,9	3,5	4,0	3,8	4,3	4,0
	Kuinka tärkeänä pidät sprinttitaavoitteisiin sitoutumista?	3,9	3,8	3,2	3,8	4,0	3,8
Rohkeus	Kuinka rohkeasti jaat mielipiteesi ja parannusehdotuksesi tiimillesi?	4,1	4,2	3,3	3,8	4,8	3,8
	Koetko, että tiimisi rohkaisee ongelmien käsittelyyn ja uusien ratkaisujen ehdottamiseen?	3,6	3,8	3,5	4,2	4,0	4,3
Keskittyminen	Onko tiimilläsi selkeä käsitys siitä, mihin sprintin aikana tulee keskittyä?	3,9	3,7	3,5	3,2	4,0	4,0
	Kuinka usein tiimisi työskentely keskeytyy lisätehtävien tai odottamattomien häiriöiden vuoksi? (1 - ei koskaan, 5 - hyvin usein)	3,9	3,5	3,8	3,2	3,5	3,8
Avoimuus	Kuinka avoimena pidät tiimisi viestintää ja tiedonjakamista?	3,6	4,5	3,8	4,5	4,8	4,0
	Kuinka avoimesti tiimisi keskustelee epäonnistumisista ja oppii niistä yhdessä?	3,6	3,7	3,2	4,2	4,5	4,0
Kunnioitus	Osoittavatko tiimin jäsenet kunnioitusta toistensa työtä ja mielipiteitä kohtaan?	4,6	4,7	4,2	4,8	5,0	4,5

Tiimeiltä kysyttiin myös, miten merkityksellisinä he kokivat yhteiset Scrum-arvot tiimin menestyksen kannalta (kuva 11). Tiimien keskiarvot vaihtelivat välillä 3 - jonkin verran ja 4 - melko paljon, mikä osoittaa, että suurin osa vastaajista koki arvojen vaikuttavan tiimin toimintaan positiivisesti. Kuitenkin osa vastaajista koki, että jaetuilla arvoilla ei ollut lainkaan tai vain vähäinen vaikutus tiimin menestykseen. Tämä voi viitata siihen, ettei arvojen merkitys ole kaikille yhtä selkeää tai että niiden vaikutusta ei tunnisteta tiimin arjessa. Tulosten perusteella voidaan todeta, että Scrum-arvot luovat puitteet tehokkaalle yhteistyölle, mutta niiden ymmärtäminen ja käytännön soveltaminen vaihtelee tiimeittäin.



Kuva 11. Yhteisten arvojen merkitys tiimin menestykselle

5.2 Intervention vaikutus kehitystiimin Scrum-osaamiseen

Osana tutkimusta järjestettiin Tiimille A interventio Scrum-viitekehyksen teoriasta ja tapahtumista. Interventio koostui kolmesta koulutuskerrasta aikavälillä 3.1.–17.2.2025 ja aiheet olivat: 1. Scrumin pilarit, 2. Scrumin arvot ja 3. Scrumin tapahtumat. Lähes koko tiimi pystyi osallistumaan kaikkiin koulutuskertoihin, mutta yksittäisiä poissaoloja myös esiintyi. Intervention vaikutuksia Tiimi A:n Scrum-viitekehyksen osaamiseen sekä sen käytännön soveltamiseen on käsitelty aiemmassa luvussa. Seuraavaksi kuitenkin vielä tiivistetty kooste intervention keskeisistä vaikutuksista.

Loppukyselyn perusteella Scrumin pilarit ja arvot olivat tiimin mielestä hyödyllisimmät koulutuskerrat ja puolet vastaajista koki, että he olivat huomanneet muutoksia työssään intervention jälkeen. Esimerkiksi tiimin kommunikaation koettiin lisääntyneen ja sprintin suunnittelun ja tavoitteiden asettamisen sekä seuraamisen kehittyneen. Myös Scrum-tapahtumat koettiin merkityksellisemmiksi. Myös ymmärrys siitä, että Scrumin tulisi palvella tiimiä ja sen toimintaa, eikä tiimin tulisi pelkästään noudattaa Scrumin tapahtumia, syveni.

Intervention vaikutus tiimin osaamiseen Scrumin teoriasta oli selvästi positiivinen. Ennen interventiota tiimin keskiarvo Scrumin pilarien ja arvojen ymmärtämisessä (kuva 7) oli 3,9, mikä jo osoitti hyvää osaamista. Koulutuksen jälkeen keskiarvo nousi 4,3:een, mikä viittaa siihen, että ymmärrys parani huomattavasti. Erityisesti ne tiimiläiset, jotka aiemmin arvioivat osaamisensa heikoksi tai kohtalaiseksi kokivat merkittävää kehitystä. Tämä osoittaa, että koulutus auttoi tasoittamaan osaamiseroja ja nostamaan koko tiimin ymmärrystä Scrumin teoriasta.

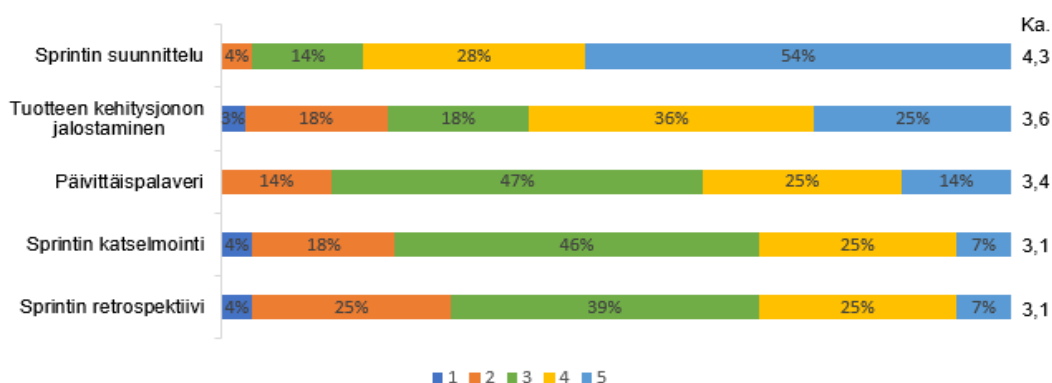
Läpinäkyvyyden käsitteen ymmärtämisessä (kuva 8) tiimin keskiarvo nousi intervention myötä. Myös käytäntöjen tasolla tiimin Scrum-tapahtumat tukivat läpinäkyvyyttä paremmin intervention jälkeen. Tämä kertoo siitä, että tiimi kehitti käytäntöjään ja ymmärsi niiden vaikutuksen läpinäkyvyyteen paremmin. Sen sijaan yrityksen vision ja strategian ymmärtäminen sekä sidosryhmien välisen viestinnän avoimuus laskivat hieman. Tämä saattaa tarkoittaa, että vaikka tiimin sisäinen läpinäkyvyys parani, organisaation ja sidosryhmien välinen viestintä koettiin edelleen haasteelliseksi. Tarkastelun merkityksen ymmärtämisessä (kuva 9) tiimi koki myös selkeää kehitystä. Tarkastelun toteutumisen

käytännöissä interventio vaikutti merkittävästi. Tiimin käytännöt tukivat tarkastelua aiemmin vain kohtalaisesti, mutta koulutuksen jälkeen tiimi arvioi tarkastelun toteutuvan hyvin. Myös tiimin edistymisen arviointi ja ongelmien tunnistaminen paranivat sekä reagointikyky havaittuihin ongelmiin ja parannusehdotuksiin nousi. Tämä kertoo siitä, että tiimi omaksui tarkastelun osaksi toimintaansa ja pystyi hyödyntämään sitä tiimin kehityksessä paremmin.

Intervention vaikutus Tiimi A:n mukauttamisen ymmärtämiseen (kuva 10) oli merkittävä ja positiivista kehitystä tapahtui tässä Scrumin pilarissa eniten. Koulutuksen jälkeen retrospektiiveissä esiin tuodut parannusehdotukset johtivat konkreettisiin muutoksiin paremmin. Kuitenkin tiimi koki autonomiansa vähentyneen koulutuksen jälkeen. Tämä voi viitata siihen, että koulutuksen myötä tiimi on alkanut havaita organisaation tai muiden sidosryhmien vaikuttavan muutoksiin enemmän kuin aiemmin oletettiin. Scrum-arvojen toteutuminen tai näkyminen arjessa (taulukko 4) ei kokenut suuria muutoksia intervention myötä. Mielenkiintoinen havainto kuitenkin oli, että tiimin sitoutuminen ja keskittyminen laskivat hieman intervention jälkeen. Rohkeus ja avoimuus paranivat koulutusjakson aikana, mikä viittaa kehittyvään keskustelukulttuuriin ja tiimin yhteistyön vahvistumiseen.

5.3 Scrum-käytäntöjen vahvuudet ja heikkoudet

Tutkimuksen vastaajista 86 % (n=28) koki, että Ponsen nykyiset Scrum-käytännöt tukivat heidän työtään ja arvioitaessa yksittäisten Scrum-tapahtumien hyödyllisyyttä (kuva 12) sprintin suunnittelu nousi selvästi hyödyllisimmäksi tapahtumaksi. Lähes 80 % vastaajista arvioi sprintin suunnittelun joko erittäin hyödylliseksi tai välttämättömäksi. Tämä viittaa siihen, että hyvin toteutettu suunnittelu- vaihe on keskeinen tiimin tehokkaan työskentelyn kannalta.



1 = Ei lainkaan hyödyllinen – Tapahtuma ei tuo mitään arvoa työhön tai tiimin suorituskykyyn, 2= Vain hieman hyödyllinen, 3 = Kohtalaisen hyödyllinen – Tapahtuma on hyödyllinen joissakin tilanteissa, mutta sen vaikutus ei ole johdonmukaisesti merkittävä, 4= Erittäin hyödyllinen, 5 = Välttämätön – Tapahtuma on kriittinen ja erittäin hyödyllinen päivittäisessä työssä tai tiimin suorituskyvyn kannalta.

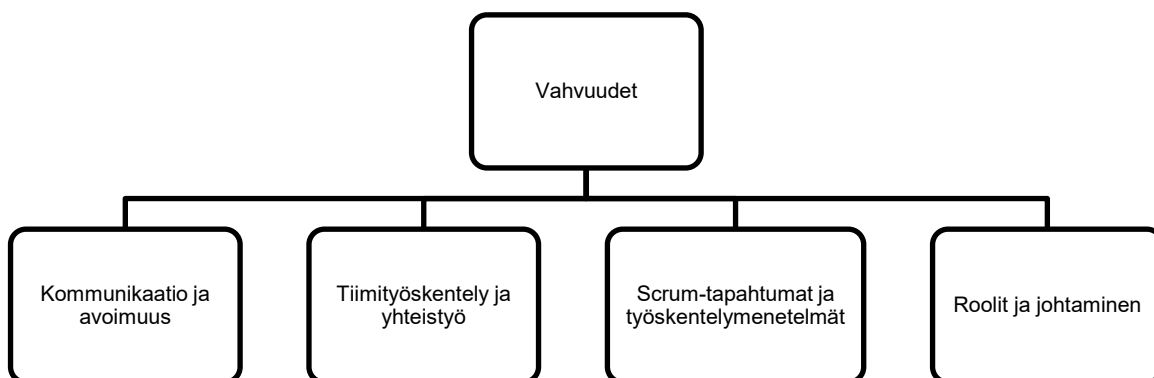
Kuva 12. Scrum-tapahtumien hyödyllisyys

Seuraavaksi hyödyllisimpänä tapahtumana koettiin tuotteen kehitysjonon jalostaminen. Vastaajista 36 % piti sitä erittäin hyödyllisenä, ja 25 % arvioi sen olevan välttämätön tiimin toiminnan kannalta, mikä korostaa sen merkitystä työtehtävien selkeyttämisessä ja suunnittelussa. Päivittäispalaverin hyödyllisyys koettiin hieman kohtalaista paremmaksi, mikä osoittaa vaihtelevia näkemyksiä sen merkityksestä. Vaikka 39 % vastaajista koki päivittäispalaverin erittäin hyödylliseksi tai välttämättömäksi, lähes 14 % koki sen vain hieman hyödylliseksi. Samoin sprintin katselmointi ja retrospektiivi jakoivat mielipiteitä. Molemmat tapaukset nähtiin kohtalaisen hyödyllisinä, mutta eivät välttämättä kriittisinä

kehitystiimin onnistumisen kannalta. Vähiten hyödyllisenä Scrum-tapahtumana koettiin sprintin retrospektiivi. Vastaajista 29 % piti sitä ei lainkaan tai vain hieman hyödyllisenä.

Yhteenvedona voidaan todeta, että Scrum-tapahtumien hyödyllisyys vaihteli tapahtumasta toiseen. Sprintin suunnittelu koettiin selvästi tärkeimmäksi, kun taas päivittäispalaverin, katselmoinnin ja retrospektiivin merkitystä ei nähty yhtä vahvasti. Tuotteen kehitysjonon jalostaminen sai keskimäärin positiivisia arvioita. Kyselylomakkeen avoimien kysymysten avulla saatiin tarkempaa tietoa Scrum-käytäntöjen vahvuuksista ja heikkouksista, ja nämä selittävät hyvin Scrum-tapahtumien arvosanojen vaihtelut eri tiimien välillä. Sama Scrum-tapahtuma saatettiin mainita joko vahvuutena tai heikkoutena. Tutkimukseen osallistuneet viisi kehitystiimiä toteuttavat Scrumia omilla tavoillaan, mikä osaltaan selittää kokemusten vaihtelua. Lisäksi yksilölliset kokemukset vaikuttavat siihen, miten hyödylliseksi eri tapahtumat koettiin.

Vahvuuksina Ponssen nykyisissä Scrum-käytännöissä havaittiin neljt keskeistä teemaa: kommunikaatio ja avoimuus, tiimityöskentely ja yhteistyö, Scrum-tapahtumat ja työskentelymenetelmät sekä roolit ja johtaminen.



Kuva 13. Ponssen Scrum-käytäntöjen vahvuudet

Kommunikaatio ja avoimuus

Tutkimuksen vastaajat pitivät vahvuutena toimivaa kommunikaatiota. Viestintä tiimeissä oli avointa ja tehokasta. Tiimeissä jaettiin tietoa ja ajatuksia avoimesti ja tiimin jäsenet tarjosivat tukea toisilleen. Myös sidosryhmien välinen viestintä mainittiin toimivaksi. Hyvä viestintä ja avoimuus lisäsivät työn läpinäkyvyyttä, auttoi tunnistamaan aiemmin huomaamatta jääneitä riippuvuuksia ja vähensi hiljaisen tiedon kertymistä tiimin sisällä.

Tiimityöskentely ja yhteistyö

Tiimin yhteistyön todettiin olevan sujuvaa ja kaikki kunnioittivat toistensa osaamista. Vastaajat kokivat, että tiimeissä ymmärrettiin omien työtehtävien ja toisten työn välinen riippuvuus. Tukea oli tarjolla sekä oman tiimin sisällä että yli tiimirajojen. Tiimeissä vallitsi hyvä yhteishenki ja kannustava ilmapiiri. Tiimit pystyivät myös sopeutumaan helposti yllättävissä tilanteissa

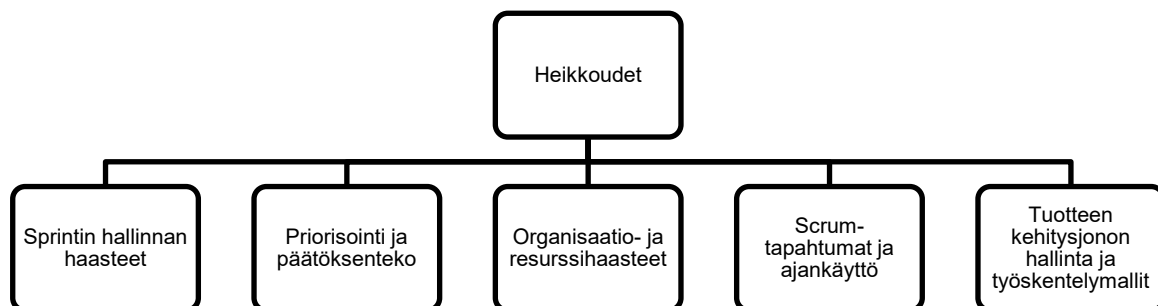
Scrum-tapahtumat ja työskentelymenetelmät

Scrum-viitekehys oli käytössä hyvin ainakin osassa tiimejä, ja kaikki tapahtumat järjestettiin näissä tiimeissä säännöllisesti. Sprintin tapahtumat olivat sopivan mittaisia ja tukivat työn tekemistä. Sprintille suunnitellut työmäärät olivat pääosin hallittavissa, ja kahden viikon sprinttisyklin koettiin olevan useimmiten sopiva työn suunnitteluun. Kvartaalisuunnittelu loi pidemmän aikavälin näkymän, mutta sitä tulisi paremmin hyödyntää myös sprinttisuunnittelussa. Päivittäispalaverit lisäsivät läpinäkyvyyttä tiimissä ja tukivat avointa ja rehellistä kommunikaatiota. Sprintin retrospektiivit saivat erityisesti kiitosta sosiaalisempana ja rennompana tapahtumana muihin Scrumin tapahtumiin verrattuna. Retrospektiivit tukivat myös tiimin avoimuutta. Tuotteen kehitysjonon jalostaminen auttoi tehtävien selkeyttämisessä, vaikka ne saattoivat tuntua välillä työläiltä tapahtumilta. Lisäksi Jira-alustan käyttö koettiin toimivaksi työtehtävien hallinnassa ja se lisäsi työn näkyvyyttä sekä mitattavuutta.

Roolit ja johtaminen

Tuoteomistajan ja tuotepäällikön roolit tunnistettiin tärkeiksi onnistuneen Scrum-prosessin kannalta. Tuoteomistajan todettiin johtavan tuotteen kehitysjonoa hyvin ja hallitsevan tuotteen kokonaisuutena. Tuotepäällikkö toimi vahvana tukena ja oli jatkuvasti tuoteomistajan saatavilla.

Heikkoudet ja mahdolliset kehityskohteet Ponsen nykyisissä Scrum-käytännöissä voidaan jaotella viiteen keskeiseen teemaan: sprintinhallinnan haasteet, priorisointi ja päätöksenteko, organisaatio- ja resurssihaasteet, Scrum-tapahtumat ja ajankäyttö sekä tuotteen kehitysjonon hallinta ja työskentelymallit.



Kuva 14. Ponsen Scrum-käytäntöjen heikkoudet

Sprintin hallinnan haasteet

Tutkimuksen perusteella yksi merkittävimmistä haasteista liittyi sprintin hallintaan ja siitä johtuviin ongelmiin. Monet vastaajat kokivat, että sprintin sisältö muuttui suunnitellusta ja paljon odottamattomia asioita, kuten asiakkaiden löytämiä virheitä, tapahtui sprintin aikana. Odottamattomat muutokset vaikuttivat sprintin onnistumiseen. Lisäksi koettiin, että sprintin tavoitteisiin sitoutuminen oli hajanaisista, ja monet kehittäjät keskittyivät ensisijaisesti omiin tehtäviinsä yhteisen sprintin tavoitteen sijaan. Tämä johti tilanteeseen, jossa sprintin aikana aloitettiin useita tehtäviä, mutta vain harvat niistä valmistuivat suunnitellussa aikataulussa.

Priorisointi ja päätöksenteko

Priorisointiin ja päätöksentekoon liittyvät haasteet nousivat esille useissa vastauksissa. Epäselvät riippuvuudet muihin tiimeihin ja päätöksenteon hajanaisuus vaikeuttivat tiimien tehokasta työskentelyä. Lisäksi korkeamman tason ”tiekartan” puute koettiin ongelmaksi, sillä tiimit keskittyivät usein yksittäisiin sprintteihin ilman laajempaa ymmärrystä niiden liiketoiminnallisesta arvosta. Myös äkilliset prioriteettimuutokset ja puutteelliset vaatimusmäärittelyt, aiheuttivat tiimeissä jatkuvaa kiirettä ja lisäsivät sprinttien epävarmuutta.

Organisaatio- ja resurssihaasteet

Organisaatiotason haasteet vaikuttivat merkittävästi tiimien kykyyn toteuttaa Scrumia tehokkaasti. Esimerkiksi päällekkäiset kokoukset muiden tiimien kanssa ja jaetut resurssit eri projektien välillä vaikuttivat tiimien keskittymiskykyyn heikentävästi. Lisäksi useat vastaajat kokivat, että tiimeiltä puuttui riittävä autonomia päätöksenteossa, mikä heikensi kehittäjien motivaatiota ja sitoutumista. Myös selkeyttä koko organisaation toimintaan kaivattiin sekä enemmän yhteistyötä eri projektien välillä.

Scrum-tapahtumat ja ajankäyttö

Scrumin tapahtumat eivät aina tuottaneet tiimeille lisäarvoa, ja niitä järjestettiin vain tiimin ulkopuolelta tulevien vaatimusten vuoksi. Esimerkiksi retrospektiivit koettiin hyödyllisiksi vain, jos niissä käsiteltiin konkreettisia kehityskohteita, jotka johtivat todellisiin muutoksiin. Osa vastaajista koki päivittäispalaverien olevan enemmän valvontaa kuin yhteistyötä edistävä tapahtuma, mikä lisäsi stressiä ja vähensi niistä koettua hyötyä. Lisäksi vastauksissa tuli esille toiveita sprinttien pituudesta. Kahden viikon sprinttien koettiin olevan liian lyhyitä, sillä ne eivät mahdollistaneet riittävää keskittymistä kehitystyöhön. Osa vastaajista ehdotti joustavampaa mallia; tiimi voisi itse päättää, mikä olisi heille sopiva sprintin pituus ja miten usein esimerkiksi päivittäispalaverit ja retrospektiivit järjestetään, jotta nämä palvelisivat paremmin tiimin tarpeita.

Tuotteen kehitysjonon hallinta ja työskentelymallit

Tuotteen kehitysjonon hallinta koettiin haasteelliseksi erityisesti sen pituuden vuoksi. Tämän takia tärkeimpien työtehtävien tunnistaminen oli vaikeaa. Haasteeksi koettiin myös se, että tuotteen kehitysjonon jalostamiseen osallistui yksittäisiä tiimin jäseniä sen sijaan, että koko tiimi osallistui aktiivisesti työskentelyyn. Lisäksi vastaajat toivat esille, että testauksen näkyvyys ja tehtävien liittyminen laajempaan kokonaisuuteen ei ollut aina selkeää. Esimerkiksi ominaisuuksien epic-tason tarkastelu puuttui, mikä vaikeutti tehtävien etenemisen hahmottamista ja priorisointia. Myös Jira-tikettien kuvaukset vaaditusta työstä, saattoivat olla epäselviä tai liian abstrakteja, mikä lisäsi tiimin epävarmuutta tehtävien sisällöstä.

6 POHDINTA

6.1 Tutkimustulosten tarkastelu

Opinnäytetyön ensimmäisenä tavoitteena oli tutkia, miten hyvin Ponsen kehitystiimit tuntevat Scrum-viitekehyksen teorian ja miten Scrumin pilarit ja arvot näkyvät tiimien arjessa. Tulokset osoittavat, että kehitystiimien ja yksittäisten vastaajien välillä oli eroja Scrumin peruseräkkeiden ymmärtämisessä. Tulosten perusteella näyttää siltä, että tiimeissä D ja E Scrum-osaaminen oli korkeimmalla tasolla, kun taas tiimeissä B ja C ilmeni enemmän hajontaa. Tämä kertoo siitä, että Scrumin teorian ymmärtäminen ei ole selvää kaikille kehitystiimeille, vaan se vaatii koulutusta ja käytäntöjen tarkastelua, jotta Scrum-viitekehyksen hyödyt voidaan saavuttaa.

Läpinäkyvyyden pilarin osalta tutkimus osoitti, että tiimit yleisesti kokivat ymmärtävänsä sen merkityksen hyvin ja tiimien käytännöt, kuten päivittäispalaverit ja sprintin katselmukset, tukivat läpinäkyvyyttä. Sen sijaan yrityksen vision ja strategian ymmärtäminen sekä sidosryhmien välinen viestintä saivat heikompia arvioita. Tämä viittaa siihen, että organisaatiotasolla tulisi panostaa avoimempaan viestintään ja tiedon jakamiseen, jotta kehitystiimit voisivat aidosti hyödyntää Scrumin tarjoamaa läpinäkyvyyttä.

Tarkastelun merkityksen ymmärtämisessä ja sen toteutumisessa arjessa oli tiimikohtaisia eroja. Tiimi D arvioi tiimin käytäntöjen tukevan tarkastelua hyvin, ja tiimi arvioi omaa toimintaansa ja reagoi nopeasti tarkastelun tuloksiin. Tiimi B:n arvioissa puolestaan tarkastelun toteutuminen jäi heikolle tai kohtalaiselle tasolle. Mukauttamisen osalta tulokset olivat samansuuntaisia kuin tarkastelussa ja läpinäkyvyydessä; osassa tiimeistä osaaminen oli vahvempaa, kun taas osalla tiimeistä oli enemmän kehitettävää mukauttamisen pilarin ymmärtämisessä ja sen toteuttamisessa arjessa. Tämä voi viitata siihen, että kaikilla tiimeillä ei ole selkeitä rakenteita tarkasteluun ja sen tuloksien hyödyntämiseen. Mukauttamisen onnistuminen edellyttää tiimeiltä kykyä tunnistaa kehitystarpeita sekä tukea organisaatiolta, jotta tiimi voi toteuttaa tarvitsemansa muutokset.

Scrumin arvojen - sitoutuminen, rohkeus, keskittyminen, avoimuus ja kunnioitus - toteutuminen kehitystiimeissä vaihteli sekin tiimeittäin, mutta yleisesti ottaen ne näyttivät toteutuvat kohtalaisesti tai hyvin. Arvoista kunnioitus toteutui parhaiten. Tiimien jäsenet arvostivat toistensa mielipiteitä ja työtä, mikä luo positiivisen ja avoimen työilmapiirin. Keskittymisen heikommat arvosanat kertoivat siitä, että tiimeillä oli haasteita pysyä fokuksittuneina sprintin tavoitteisiin ja työhön. Tämä voi johtua esimerkiksi epäselvistä prioriteeteista, heikosta suunnittelusta tai yllättävistä työtehtävistä, jotka heikentävät tiimien kykyä keskittyä olennaiseen ja saavuttaa asetettuja tavoitteita.

Mielenkiintoista kuitenkin oli, että suurin osa tiimeistä koki yhteisten arvojen vaikuttavan tiimin menestykseen vain kohtalaisesti. Tämä voi tarkoittaa, että vaikka arvot ovat tärkeitä ja niitä toteutetaan, niiden merkitystä ja vaikutusta tiimin toimintaan ei ole täysin sisäistetty tai ymmärretty kaikissa tiimeissä. Tiimien tulisi käyttää aikaa arvoista keskusteluun, sillä se voi auttaa tiimiä ymmärtämään, että arvot eivät ole pelkästään abstrakteja käsitteitä, vaan niillä on konkreettinen vaikutus tiimin toimintaan ja menestykseen.

Opinnäytetyön toisena tavoitteena oli tutkia, voidaanko lyhyellä interventiolla vaikuttaa yhden kehitystiimin Scrum-osaamiseen. Koulutusjaksolla oli selvästi myönteinen vaikutus Tiimin A tuloksiin.

Koulutuksessa käsitellyt Scrumin periaatteet, arvot ja tapahtumat auttoivat tiimiä syventämään ymmärrystään ja parantamaan käytäntöjään. Loppukyselyn mukaan tiimi koki muutoksia kommunikaatiossa, suunnittelussa ja tavoitteiden asettamisessa sekä Scrum-tapahtumien merkityksen ymmärtämisessä. Erityisesti Scrumin pilareiden teoreettinen ymmärtäminen parani intervention myötä.

Intervention vaikutus kokemuksiin Scrum-tapahtumien hyödyllisyydestä oli myös selkeä. Päivittäispalaverit, sprinttien katselmoinnit ja retrospektiivit koettiin hyödyllisemmiksi, mikä kertoo, että tiimi sai paremman käsityksen näiden tapahtumien roolista ja merkityksestä Scrum-viitekehityksessä. Tiimi A:n arvoihin intervention ei ollut kovin suuri, mutta erityisesti rohkeuden ja avoimuuden parantaminen viittaa siihen, että tiimi on onnistunut kehittämään keskustelukulttuuriaan ja yhteistyötään. Koulutuksen ja oppimisen myötä tiimi voi hyödyntää Scrum-viitekehystä entistä tehokkaammin arjessaan ja parantaa suorituskykyään.

Tutkimuksen perusteella voidaan todeta, että koulutuksella oli selkeä positiivinen vaikutus Scrum-osaamiseen ja käytäntöihin. Pelkkä lyhyt interventio ei välttämättä kuitenkaan riitä juurruttamaan uusia toimintamalleja pysyväksi osaksi tiimin arkeen, vaan oppimista ja soveltamista tulisi seurata ja tukea myös jatkossa. Lisäksi organisaatiotasolla on tärkeää varmistaa, että Scrumin periaatteet eivät jää vain kehitystiimien vastuulle, vaan niitä tuetaan johdon ja sidosryhmien toimesta.

Opinnäytetyön kolmantena tavoitteena oli tutkia, miten Ponssin nykyiset Scrum-käytännöt tukevat tiimien työskentelyä. Tulosten perusteella voidaan todeta, että Scrum-viitekehityksen käyttö kehitystiimeissä toi mukanaan sekä vahvuuksia että haasteita. Vastaajista 86 % koki, että nykyiset Scrum-käytännöt tukivat tiimien työskentelyä, mutta yksittäisten Scrum-tapahtumien hyödyllisyys vaihteli. Tärkeimmiksi Scrum-tapahtumiksi nousivat sprintin suunnittelu ja tuotteen kehitysjonon jalostus, kun taas retrospektiivit ja sprinttikatselmuksot koettiin vähiten hyödyllisiksi.

Scrumin soveltaminen ja muokkaaminen tiimin tarpeisiin on tärkeää, sillä samat teemat nousivat esiin sekä vahvuuksina että haasteina. Vahvuuksina Ponssin nykyisissä käytännöissä nousivat avoin kommunikaatio, tiimityöskentely ja selkeät roolit. Kehitystiimien sisällä oli hyvä yhteishenki ja tiimityöskentely oli sujuvaa. Lisäksi Scrum-tapahtumat tarjosivat mahdollisuuksia tarkastella tuotteen etenemistä ja tiimin työskentelyä. Päivittäispalaverit tukivat läpinäkyvyyttä ja avoimuutta tiimin sisällä, samoin kuin retrospektiivien ja tuotteen kehitysjonon jalostamisen koettiin lisäävän selkeyttä työhön.

Haasteita Ponssin nykyisissä Scrum-käytännöissä kuitenkin ilmeni erityisesti sprintinhallinnan, priorisoinnin ja päätöksenteon osalta. Yksi keskeinen ongelma oli sprintin sisällön muuttuminen kesken sprintin, mikä vaikeutti sprintin tavoitteiden saavuttamista ja tiimin työskentelyä. Muuttuvat prioriteetit sekä liian pitkä tai vaikeasti hallittava tuotteen kehitysjohto hankaloittivat myös tiimien työskentelyä. Lisäksi päätöksenteon hajanaisuus ja epäselvät riippuvuudet tiimien välillä aiheuttivat haasteita. Osa vastaajista myös koki, että Scrum-tapahtumat eivät tuoneet tiimin työskentelyyn lisäarvoa, vaan ne suoritettiin ulkoisen vaatimusten vuoksi. Erityisesti retrospektiivit ja päivittäispalaverit eivät tuntuneet hyödyllisiltä.

Kokonaisuudessaan tutkimuksen perusteella voidaan todeta, että Scrum-viitekehitys ei ole yksi ja sama kaikille tiimeille, vaan sen soveltaminen ja muokkaaminen tiimin tarpeisiin on tärkeää. Tulokset tuovat esille sen, että Scrum toteutuu käytännössä eri tavoin tiimeissä ja yksittäiset vastaajat kokevat Ponssin nykyisten Scrum-käytäntöjen hyödyllisyyden eri tavoin. Voidaankin todeta, että Scrum

voi tuoda merkittäviä hyötyjä kehitystiimeille, mutta sen onnistunut toteutus vaatii jatkuvaa arviointia, mukauttamista ja yhteistyötä sekä tiimien sisällä että organisaation tasolla. Myös koulutusta tarvitaan, sillä kehitystiimien jäsenillä tulee olla myös tarvittava teoreettinen tietämys Scrum-viitekehiksestä, jotta Scrumin käyttäminen tiimin työskentelyssä on mielekästä ja tehokasta. Tutkimuksen perusteella voidaan ehdottaa useita kehitysideoita, jotka voisivat parantaa Ponsen kehitystiimien ketteryyttä ja Scrum-prosessien tehokkuutta.

Sprintinhallinnan osalta tiimien tulisi määritellä sprintille yhteinen tavoite tarkemmin ja varmistaa, että tavoite ei muutu sprintin aikana. Scrum Masterin tulisi suojella tiimiä paremmin ulkoisilta muutoksilta ja käydä tuoteomistajan ja tiimin kanssa keskusteluja työn kiireellisyydestä ennen kuin työ otetaan sprintille mukaan. Kesken sprintin tuleviin muutoksiin voisi varautua paremmin, esimerkiksi varaumalla jo sprintin suunnittelussa kapasiteettia odottamattomille töille, kuten kiireellisille virheenkorjauksille. Tuotteen kehitysjonon jalostamisessa voisi tarkemmin tarkastella, onko työt tarpeeksi hyvin määritelty ja täyttävätkö ne DOR-kriteerit (Definition of Ready), jotta kehittäjät voivat ottaa ne seuraavalle sprintille työn alle. Olisi tärkeää, että tuoteomistaja osallistuu suunnitteluun, jotta varmistetaan, että työtehtävät ovat linjassa tuotteen vision kanssa, ja että tiimi ymmärtää selkeästi, mitä sprintissä tulee saavuttaa. Lisäksi olisi tärkeää käyttää työmääräarvioita tarkasti ja huomioida ne sprintin suunnittelussa, jotta työmäärä on realistinen suhteessa tiimin kapasiteettiin. Tiimin kapasiteettia tulisi seurata tarkastelemalla edellisten sprinttien suorituksia, jotta varmistetaan, että seuraavan sprintin suunniteltu työmäärä on realistinen ja toteutettavissa.

Priorisoinnin ja päätöksenteon selkeyttäminen on myös tärkeää. Tiimien väliset riippuvuudet tulisi tuoda paremmin näkyville, esimerkiksi lisäämällä tiimien välistä kommunikointia ja suunnittelua. Lisäksi riippuvuuksien hallintaa voisi parantaa visuaalisilla työkaluilla, jotka tekisivät riippuvuuksista kaikille tiimeille läpinäkyviksi. Pitkän aikavälin tiekartan luominen ja tuotteen vision vahvistaminen tiimitasolla auttaisi myös sprinttisuunnittelussa ja tukisi kehitystiimejä ymmärtämään, mikä arvo heidän tekemällään työllä on suhteessa organisaation strategisiin tavoitteisiin. Tiekartan tulisi olla elävä asiakirja, jota päivitetään ja tarkastellaan tiimien kanssa säännöllisesti.

Scrum-tapahtumien osalta on mahdollista kehittää niiden rakennetta tiimin tarpeiden mukaan. Jokaisen tiimin tulisi yhdessä keskustella siitä, mikä on kunkin tapahtuman tavoite ja miten nykyiset käytännöt tukevat tiimin toimintaa. Retrospektiivit ja sprintin katselmoinnit koettiin vähiten hyödyllisiksi, joten erityisesti näiden tapahtumien merkitystä tiimeille tulisi pohtia ja rohkeasti kokeilla erilaisia toteutustapoja. Scrum-viitekehiksen tulisi palvella tiimiä, joten jokaisen tiimin täytyy itse löytää heille sopivat käytännöt, sillä yhtenäiset ohjeet eivät välttämättä toimi kaikille. Ponsse voi tarjota "kehikset" Scrumin käytölle, mutta tiimin vastuulla on niiden toteuttaminen ja jatkuva kehittäminen omien tarpeidensa mukaisesti. Tällainen lähestymistapa mahdollistaa joustavuuden ja tiimien tarpeisiin räätälöidyn prosessin, joka tukee tehokasta kehitystyötä ja parantaa tiimien suoriutumista.

6.2 Tutkimuksen eettisyys ja luotettavuus

Laadullisen tutkimuksen luotettavuuden tarkasteluun ei ole olemassa yksiselitteisiä ohjeita, mutta Tuomi ja Sarajärvi (2018) ovat kirjassaan listanneet yhdeksän kohtaa, joilla tutkimuksen luotettavuutta voidaan arvioida. Tässä opinnäytetyön raportissa olen kuvannut *tutkimuksen kohteen ja tarkoituksen* mahdollisimman hyvin, jotta opinnäytetyön lukijalla olisi selkeä käsitys siitä, mitä tutkimus

käsittelee. Lisäksi varmistin saatekirjeellä ja tietosuojailmoituksella, että kyselyyn vastanneet ja interventioon osallistuneet saivat jo tutkimuksen alussa kattavan kuvauksen sen tavoitteista sekä omista oikeuksistaan. Koska työskentelin Ponsella Scrum Masterina, on syytä pohtia tutkimuksen luotettavuuden kannalta *omaa sitoumustani ja tutkija-tiedonantaja suhdetta*. Opinnäytetyön aihe oli itselleni ammatillisesti merkityksellinen, ja tutkimusprosessi tuki ammatillista kehittymistäni, joten motivaatio ja sitoutuminen tutkimuksen tekemiseen oli korkea. On kuitenkin hyvä huomioida, että tutkimustuloksiin saattoi vaikuttaa se, että osa osallistujista tunsikin minut jo entuudestaan työni kautta.

Tuomi ja Sarajärvi (2018) tuovat esille, että *aineistonkeruu ja menetelmät* voivat vaikuttaa tutkimuksen luotettavuuteen. Tässä raportissa olen pyrkinyt kuvaamaan aineistonkeruun etenemisen mahdollisimman tarkasti, jotta lukija saa käsityksen tutkimuksen kulusta ja pystyy tekemään omat johtopäätökset aineistonkeruun luotettavuudesta. Tutkimuksen aineisto on kerätty kyselylomakkeen avulla, mutta myös havainnointi ja koulutusjakso olivat osa tutkimuksen toteuttamista. Usean aineistonkeruumenetelmän yhdistäminen antoi laajemman kuvan tutkittavasta tapauksesta, mikä paransi luotettavuutta. Sähköisen kyselylomakkeen heikkoutena oli kuitenkin se, että tutkimuksen osallistujia ei voinut tehdä tarkentavia kysymyksiä lomakkeesta, jolloin kysymysten väärinymmärtäminen oli mahdollista. Kyselylomake oli myös melko pitkä, mikä saattoi heikentää vastaajien motivaatiota ja vaikuttaa vastausten laatuun.

Tutkimuksen luotettavuutta voidaan arvioida myös sen *keston, analyysiin ja raportointiin* liittyvien tekijöiden kautta (Tuomi & Sarajärvi, 2018). Minulla oli hyvin aikaa opinnäytetyön tekemiseen, sillä aloitin opinnäytetyöprosessin töiden ohella loppuvuonna 2024 ja alkuvuoden 2025 keskityin vain opinnäytetyön tekemiseen. Minulla oli hyvin aikaa perehtyä teoreettiseen viitekehukseen ja kyselylomakkeen suunnitteluun. Tutkimuksen kyselylomake oli auki 13.1.–31.1.2025 välisen ajan ja siihen saatiin vastauksia 28 kappaletta (N=43). Kyselyn vastausprosentti oli 65 %, mitä voidaan pitää melko hyvänä. Näin ollen voidaan olettaa, että vastaukset kuvastavat kehitystiimien Scrum-osaaamista ja käytäntöjä hyvin. Tutkimuksen analyysin olen pyrkinyt kuvaamaan tähän raporttiin tarkasti, jotta lukija voi itse tehdä päätelmät analyysin luotettavuudesta. Lisäksi määrällisen aineiston analyysissä käytin apuna Webropolin omia analysointi- ja raportointityökaluja, joten aineiston käsittelyssä ei ole tapahtunut laskennallisia virheitä. Olen pyrkinyt tässä raportissa luotettavuuteen myös sillä, että työn teoreettinen viitekehys antaisi tutkimuksen aiheesta perehtymättömälle riittävät pohjatiedot tutkimuksen ymmärtämiseksi. Lisäksi käyttämällä tutkimustulosten raportoinnissa erilaisia graafisia esityksiä ja tunnuslukuja, olen pyrkinyt varmistamaan, että tulokset ovat helposti ymmärrettävissä ja havainnollistavat selkeästi keskeiset tulokset.

Eettinen tarkastelu on olennainen osa tutkimusta, sillä se varmistaa tutkimuksen toteuttamisen tavalla, joka kunnioittaa osallistujien oikeuksia, yksityisyyttä ja hyvinvointia. Eettinen tarkastelu edistää myös rehellisyyttä ja luotettavuutta tutkimusprosessissa. Tutkimuseettinen neuvottelukunta (TENK) on antanut ohjeet ihmiseen kohdistuvan tutkimuksen eettisistä periaatteista. Osallistuessaan tutkimukseen tutkittavalla henkilöllä on oikeus osallistua vapaaehtoisesti ja oikeus myös kieltäytyä osallistumisesta. Tutkittavalla on oikeus keskeyttää osallistumisensa ja peruuttaa suostumuksensa. Lisäksi tutkimukseen osallistuvalla on oikeus saada tietoa tutkimuksen toteutuksesta, tutkimuksen tavoitteista, henkilötietojen käsittelystä ja mahdollisista riskeistä liittyen tutkimukseen osallistumiseen. (TENK 2019.)

Keräsin tutkimuksen aineiston kyselylomakkeella (liite 1), jonka alussa oli saateviesti tutkimuksen aiheesta ja tavoitteesta. Kyselylomakkeen liitteenä oli myös tietosuojailmoitus (liite 2) henkilötietojen käsittelyyn liittyen. Lisäksi kysyin interventioon osallistuneelta tiimiltä vuoden 2024 loppupuolella alustavaa kiinnostusta tutkimukseeni osallistumisesta. Tammikuussa, ennen intervention alkamista, keräsin osallistujilta erillisen suostumuslomakkeen (liite 3) interventioon osallistumisesta. Näin interventioon osallistuneilla oli riittävästi aikaa päättää, haluavatko he osallistua tutkimukseen. Lisäksi omat yhteystietoni löytyivät kyselylomakkeesta, tietosuojailmoituksesta ja suostumuslomakkeesta, jolloin tutkimukseen osallistuneet pystyivät ottamaan minuun yhteyttä tarvittaessa.

Tutkimuseettinen neuvottelukunta (2019) antaa ohjeet myös yksityisyyden suojasta. Yleisenä periaatteena on, että tutkimukseen osallistuneiden henkilöiden yksityisyyttä suojellaan, henkilönimiä julkaistaessa tulee huomioida tekijänoikeudet, eikä osallistujille tule luvata täyttä tunnistamattomuutta, jos tämä ei ole saavutettavissa. (TENK 2019.) Tässä tutkimuksesta en kerännyt tutkimukseen osallistuneilta suoria henkilötietoja. Mahdolliset yksilöivät tunnisteet, kuten kehitystiimi, pseudonymisoin. Annoin kehitystiimeille korvaavat tunnisteet heti tutkimusaineiston käsittelyn alussa ja käytin näitä tutkimuksen analyysissä ja raportoinnissa.

6.3 Jatkotutkimusehdotukset ja opinnäytetyön merkitys

Opinnäytetyön aikana nousi esiin yksi erityisen kiinnostava jatkotutkimusaihe, joka voisi syventää ymmärrystä tiimien ketteryydestä ja Scrumin käytöstä Ponssella. Yksi mahdollinen tutkimus olisi toteuttaa pidempikestoinen interventio, jossa hyödynnettäisiin vielä laajemmin erilaisia aineistonkeruumenetelmiä ja mittareita kehityksen ja toimintatapojen seurantaan. Esimerkiksi Jirasta löytyviä raportointityökaluja voitaisiin käyttää analysoimaan, onko koulutuksella todellista vaikutusta tiimin toimintaan. Tällaiset mittarit voivat tarjota tietoa esimerkiksi siitä, paraneeko tiimien sprinttisuunnittelu ja tavoitteiden saavuttaminen koulutuksen myötä.

Tämän tutkimuksen aineisto kerättiin kyselylomakkeella ja aineisto koostui viiden eri tiimin vastauksista. Tulokset tarjosivat yleiskuvan Ponssen kehitystiimeistä, mutta tulokset jäivät silti vielä melko pintapuoliseksi. Pidempikestoisessa interventiossa voitaisiin hyödyntää aineistonkeruussa myös yksilö- tai ryhmähaastatteluja. Haastattelut voisivat tuoda esiin syvällisempiä näkökulmia, sillä se mahdollistaisi tutkijan esittää tarkentavia kysymyksiä ja tutkimukseen osallistujien perustella omia kokemuksiaan ja ajatuksiaan. Mikäli jatkotutkimus keskittyisi yhteen tiimiin, olisi mahdollista tarkastella syvällisemmin myös tiimin sisäistä vuorovaikutusta, vahvuuksia ja haasteita. Tällainen lähestymistapa mahdollistaisi tarkempien johtopäätösten tekemisen ja kehitysideoiden esittämisen tiimin toiminnan ja Scrum-viitekehityksen vahvistamiseksi.

Opinnäytetyön tekeminen on ollut merkittävä kokemus ammatilliselle kehitykselleni, sillä se on syventänyt osaamistani ketterästä kehityksestä ja erityisesti Scrum-viitekehityksestä. Tämä on tukenut ammatillista rooliani Scrum Masterina, sillä ymmärrän nyt paremmin, miten Scrum voi auttaa tiimiä toimimaan ketterästi, edistämään jatkuvaa oppimista, vahvistamaan tiimin yhteistyötä ja tukemaan itseohjautuvuutta. Osaan jatkossa paremmin perustella Scrum-viitekehityksen vahvuuksia ja tukea tiimin toimintaa. Tutkimustulokset ovat auttaneet minua hahmottamaan kehitystiimien osaamista, asenteita ja nykyisiä käytäntöjä, minkä ansiosta voin jatkossa keskittyä työssäni erityisesti tutkimuksessa esiin nousseisiin haasteisiin. Lisäksi voin hyödyntää interventioita varten luotua materiaalia myöhemmin työskennellessäni muiden kehitystiimien kanssa.

Ponsselle tämä opinnäytetyö tarjoaa arvokasta tietoa kehitystiimien Scrum-osaamisesta ja käytännöistä. Tutkimus on auttanut tunnistamaan tiimien vahvuuksia ja heikkouksia sekä alueita, joissa ketteryyttä ja Scrumia voidaan edelleen parantaa. Tulosten pohjalta on syntynyt konkreettisia kehitysehdotuksia, jotka voisivat parantaa tiimien työskentelyä. Nämä parannukset voisivat osaltaan vaikuttaa jopa lopputuotteen laatuun. Tutkimuksen tuloksia voidaan hyödyntää myös tulevien koulutustarpeiden tunnistamisessa ja suunnittelussa.

LÄHTEET

Työssä on käytetty tekoälyä seuraavasti: ChatGPT 2024. OpenAI. GPT-4o. Käytetty opinnäytetyön lähdemateriaalin etsimiseen ja teoreettisen viitekehyksen kielentarkistukseen, joulukuu 2024.
<https://chat.openai.com>

Agile Alliance 2001. Ketterän ohjelmistokehityksen julistus. Verkkojulkaisu. <https://agilemanifesto.org/iso/fi/manifesto.html>. Viitattu 9.12.2024.

Agile Alliance n.d. Agile resources. A Short History of Agile. Verkkojulkaisu. <https://www.agilealliance.org/a-short-history-of-agile/>. Viitattu 9.12.2024.

Ahmad, M. O., Markkula, J. & Oivo, M. 2013. Kanban in software development: A systematic literature review. 39th Euromicro Conference on Software Engineering and Advanced Applications (SEAA). <https://doi.org/10.1109/SEAA.2013.28>. Viitattu 13.12.2024.

Akif, R. & Majeed, H. 2012. Issues and Challenges in Scrum Implementation. International Journal of Scientific & Engineering Research 3(8), 1–4. https://www.researchgate.net/publication/303632370_Issues_and_Challenges_In_Scrum_Implementation. Viitattu 31.12.2024.

Akour, I. A., Alshurideh, M. T., Al Kurdi, B. & Hamadneh, S. 2024. Exploring the Impact of Scrum Framework on Project Effectiveness: A Quantitative Analysis of Agile Implementation Challenges and Benefits. International Journal of Theory of Organization and Practice (IJTOP) 4(2), 161–174. <https://www.journals.gafim.com/index.php/ijtop/article/view/407>. Viitattu 30.12.2024.

Bancroft-Connors, J. n.d. What Is Product Backlog Refinement? Verkkojulkaisu. <https://resources.scrumalliance.org/Article/product-backlog-refinement>. Viitattu 9.12.2024.

Beck, K. & Andres, C. 2004. Extreme Programming Explained: Embrace Change, Second Edition. E-kirja. Boston: Addison-Wesley Professional. <https://learning.oreilly.com/library/view/extreme-programming-explained/0321278658/>. Viitattu 10.12.2024.

Bhatt, T. 2023. Benefits of Software Development Life Cycle (SDLC). Verkkojulkaisu. Päivitetty 25.10.2023. <https://www.intelivita.com/blog/benefits-of-software-development-life-cycle/>. Viitattu 1.12.2024.

Bluestein, N. 2023. What is the Software Development Lifecycle (SDLC)? Verkkojulkaisu. Päivitetty 29.8.2023. <https://www.harness.io/harness-devops-academy/what-is-the-software-development-lifecycle-sdlc>. Viitattu 1.12.2024.

Carvalho, B. & Mello, C. 2011. Scrum agile product development method - literature review, analysis and classification. Product: Management & Development 9(1), 39–49. <http://dx.doi.org/10.4322/pmd.2011.005>. Viitattu 31.12.2024.

Cho, J. 2008. Issues And Challenges of Agile Software Development with Scrum. Issues in Information Systems 9(2), 188–195. https://doi.org/10.48009/2_iis_2008_188-195. Viitattu 31.12.2024.

Dwivedi, N., Katiyar, D. & Goael, G. 2022. A Comparative Study of Various Software Development Life Cycle (SDLC) Models. International Journal of Research in Engineering, Science and Management 5(3), 141–144. <https://journal.ijresm.com/index.php/ijresm/article/view/1881>. Viitattu 8.12.2024.

Dooley, J. 2011. Software Development and Professional Practice. E-kirja. New York: Apress. <https://learning.oreilly.com/library/view/software-development-and/9781430238010/>. Viitattu 30.11.2024.

- Ekechi, C. C., Okeke, C. D. & Adama H. E. 2024. Enhancing agile product development with scrum methodologies: A detailed exploration of implementation practices and benefits. *Engineering Science & Technology Journal* 5(5), 1542–1570. <https://doi.org/10.51594/estj.v5i5.1108>. Viitattu 30.12.2024.
- Elo, S., Kajula, O., Tohmola, A. & Kääriäinen, M. 2022. Laadullisen sisällönanalyysin vaiheet ja eteneminen. *Hoitotiede* 34(4), 215–225. <https://journal.fi/hoitotiede/article/view/128987/78028>. Viitattu 22.1.2025.
- Foster, E. C. 2011. *Software Engineering: A Methodical Approach*. E-kirja. New York: Apress. <https://www.oreilly.com/library/view/software-engineering-a/9781484208472/>. Viitattu 30.11.2024.
- Futurum Technology 2024. Ohjelmistokehityksen elinkaaren mallit – Kaikki mitä tarvitset tietää. Verkkojulkaisu. Päivitetty 11.4.2024. <https://www.futurum.tech/blog/index.php/2024/04/11/ohjelmistokehityksen/>. Viitattu 1.12.2024.
- Ghosh, A. 2024. Advantages and Disadvantages of SDLC (Software Development Life Cycle). Verkkojulkaisu. Päivitetty 20.6.2024. <https://ellow.io/advantages-and-disadvantages-of-sdlc/>. Viitattu 1.12.2024.
- Gupta, A., Rawal, A. & Barge, Y. 2021. Comparative Study of Different SDLC Models. *International Journal for Research in Applied Science & Engineering Technology* 9 (XI), 73–80. <https://doi.org/10.22214/ijraset.2021.38736>. Viitattu 1.12.2024.
- Hoffmann, J. n.d. The Three Pillars of Scrum. Verkkojulkaisu. <https://resources.scrumalliance.org/Article/three-pillars-of-scrum>. Viitattu 20.12.2024.
- Hossain, M. I. 2023. Software Development Life Cycle (SDLC) Methodologies for Information Systems Project Management. *International Journal for Multidisciplinary Research (IJFMR)* 5(5), 1–36. https://www.researchgate.net/publication/373800862_Software_Development_Life_Cycle_SDLC_Methodologies_for_Information_Systems_Project_Management. Viitattu 8.12.2024.
- Juuti, P. & Puusa, A. 2020. Toimintatutkimus. Sekä toimintaa että tutkimusta. Teoksessa Puusa, A. & Juuti, P. (toim.) *Laadullisen tutkimuksen näkökulmat ja menetelmät*. Helsinki: Gaudeamus, 268–281.
- Jyrkämä, J. 2021. Toimintatutkimus. Teoksessa Vuori, J. (toim.) *Laadullisen tutkimuksen verkkokäsikirja*. Tampere: Yhteiskuntatieteellinen tietoarkisto. <https://www.fsd.tuni.fi/fi/palvelut/metelmaopeetus/>. Viitattu 2.1.2025.
- Layton, M. C., Ostermiller, S. J. & Kynaston, D. J. 2022. *Scrum for Dummies*. E-kirja. Hoboken, New Jersey: John Wiley & Sons, Inc. Viitattu 20.12.2024.
- Novriansyah, N. 2024. Understanding the Scrum Framework: What, Why, How, and Best Practices. Verkkojulkaisu. Päivitetty 16.8.2024. <https://medium.com/novai-devops-101/understanding-the-scrum-framework-what-why-how-and-best-practices-b8d83a4c8c9f>. Viitattu 27.12.2024.
- Pargaonkar, S. 2023. A Comprehensive Research Analysis of Software Development Life Cycle (SDLC) Agile & Waterfall Model Advantages, Disadvantages, and Application Suitability in Software. *International Journal of Scientific and Research Publications* 13(8), 120–124. <http://dx.doi.org/10.29322/IJSRP.13.08.2023.p14015>. Viitattu 8.12.2024.
- Poppendieck M. & Poppendieck T. 2003. *Lean Software Development: An Agile Toolkit*. Upper Saddle River, NJ: Addison-Wesley.
- Poppendieck M. & Poppendieck T. 2008. *Implementing Lean Software Development. From Concept to Cash*. Upper Saddle River, NJ: Addison-Wesley.

- Rodríguez, P., Mäntylä, M., Oivo, M., Lwakatare, L. E., Seppänen, P. & Kuvaja P. 2019. Advances in Using Agile and Lean Processes for Software Development. Teoksessa Memon, A. (toim.) Advances in Computer, Volume 113. Cambridge: Academic Press, An imprint of Elsevier, 135–224. <https://doi.org/10.1016/bs.adcom.2018.03.014>. Viitattu 13.12.2024.
- Royce, W. W. 1970. Managing the Development of Large-Scale Software Systems. Proceedings of IEEE WESCON 26, 328–388. <https://dl.acm.org/doi/pdf/10.5555/41765.41801>. Viitattu 8.12.2024.
- Schwaber, K. & Sutherland, J. 2020. Scrum-opas. Scrumin määritelmä ja pelisäännöt. Verkkojulkaisu. <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Finnish.pdf>. Viitattu 18.12.2024.
- Scrum Alliance n.d. What is Scrum? Verkkojulkaisu. <https://www.scrumalliance.org/about-scrum?> Viitattu 18.12.2024.
- Sommerville, I. 2019. Engineering Software Products: An Introduction to Modern Software Engineering. E-kirja. Boston: Pearson. <https://learning.oreilly.com/library/view/engineering-software-products/9780137586721/>. Viitattu 10.12.2024.
- Sommerville, I. 2011. Software Engineering, 9th edition. Boston: Pearson.
- Stephens, R. 2015. Beginning Software Engineering. E-kirja. Indianapolis: John Wiley & Sons, Inc. <https://learning.oreilly.com/library/view/beginning-software-engineering/9781118969175/>. Viitattu 7.12.2024.
- Tsui, F., Karam, O. & Bernai, B. 2022. Essentials of Software Engineering, 5th Edition. E-kirja. Burlington: Jones & Bartlett Learning. <https://learning.oreilly.com/library/view/essentials-of-software/9781284229004/>. Viitattu 30.11.2024.
- Tuomi, J. & Sarajärvi, A. 2018. Laadullinen tutkimus ja sisällönanalyysi. Uudistettu laitos. Helsinki: Kustannusosakeyhtiö Tammi.
- Tutkimuseettinen neuvottelukunta. 2019. Ihmiseen kohdistuvan tutkimuksen eettiset periaatteet ja ihmistieteiden eettinen ennakoarviointi Suomessa. Tutkimuseettisen neuvottelukunnan ohje 2019. Tutkimuseettisen neuvottelukunnan julkaisuja 3/2019. Helsinki: TENK. https://tenk.fi/sites/default/files/2021-01/Ihmistieteiden_eettisen_ennakoarvioinnin_ohje_2020.pdf. Viitattu 3.1.2025.
- Tutorials Point n.d. SDLC – Quick Guide. Verkkojulkaisu. https://www.tutorialspoint.com/sdlc/sdlc_quick_guide.htm. Viitattu 1.12.2024.
- Tähtinen, J., Laakkonen, E. & Broberg, M. 2020. Tilastollisen aineiston käsittelyn ja tulkinnan perusteita. Turku: Turun yliopiston kasvatustieteiden laitos ja Opettajankoulutuslaitos. <https://www.utu.fi/handle/10024/149687>. Viitattu 22.1.2025.
- Vuori, J. 2021. Tapaustutkimus. Teoksessa Vuori, J. (toim.) Laadullisen tutkimuksen verkkokäsikirja. Tampere: Yhteiskuntatieteellinen tietoaarkisto. <https://www.fsd.tuni.fi/fi/palvelut/menetelmaopetus/>. Viitattu 2.1.2025.
- Washizaki, H. 2024 Guide to the Software Engineering Body of Knowledge (SWEBOK Guide), Version 4.0. Verkkojulkaisu. IEEE Computer Society. www.swebok.org. Viitattu 30.11.2023.
- Wonohardjo, E. P., Sunaryo, R. F., Sudiyono, Y., Surantha, N & Suharjito. 2019. A Systematic Review of SCRUM in Software Development. International Journal on Informatics Visualization 3(2), 108-112. https://www.researchgate.net/publication/373149699_A_Systematic_Review_of_SCRUM_in_Software_Development. Viitattu 30.12.2024.

LIITE 1: KYSELYLOMAKE

Background questions

1. How long have you been working in the IT industry or in software development?

Years:

2. Do you have experience with agile methods other than Scrum?

No

Yes. If yes, which ones?

3. How long have you been working within the Scrum framework?

Years and months:

4. Have you attended any training on Scrum? For example, training aimed at certification or training organized by your company?

No

Yes. If yes, which one?

5. How well do you understand the core pillars of Scrum (transparency, inspection, adaptation) and the values (commitment, courage, focus, openness, respect)?

(1 = Not at all, 5 = Very well)

6. Which development team do you work with at Ponsse?

Team name:

The three pillars of Scrum

Transparency

7. How well do you understand what transparency means in Scrum?

- 1 Not at all – I don't know what transparency means in the Scrum context.
- 2 Poorly – I partially understand transparency in Scrum, but its application is unclear.
- 3 Moderately – I understand transparency in Scrum but not its full significance.
- 4 Well – I understand transparency and its importance in the project.
- 5 Very well – I understand and can apply transparency effectively in the team.

8. Do your team's practices (e.g., daily stand-ups, sprint reviews) support transparency effectively?

- 1 Not at all – Team practices don't support transparency.
- 2 Poorly – Transparency is partially supported but needs improvement.
- 3 Moderately – Practices moderately support transparency.
- 4 Well – Practices strongly support transparency and effective information sharing.
- 5 Very well – Practices fully support transparency and make all information easily accessible

9. How clearly do you understand the company's vision and strategy, especially concerning product development?

- 1 Not at all clear – I don't know the company's vision or strategy.
- 2 Poorly – I have little understanding of the vision and strategy.
- 3 Moderately – I understand the basics but lack clarity on details.
- 4 Well – I have a solid understanding of the vision and strategy.
- 5 Very well – I have a detailed understanding of the vision and strategy.

10. Do you feel that communication between your team and stakeholders is open, clear, and smooth?

- 1 Not at all – Communication is unclear, difficult, and ineffective.
- 2 Poorly – Communication is weak and hinders collaboration.
- 3 Moderately – Communication works to some extent but needs improvement.
- 4 Well – Communication is mostly clear, open, and smooth.
- 5 Very well – Communication is very open, clear, and smooth.

Inspection

11. How well do you understand the importance of inspection in supporting project progress and team learning?

- 1 Not at all – I don't understand the importance of inspection.
- 2 Poorly – I partially understand the importance of inspection, but I am unclear about its role.
- 3 Moderately – I understand how inspection supports project progress and team learning, but not its full role.
- 4 Well – I understand inspection well and its importance for project progress and team learning.
- 5 Very well – I understand inspection very well and can apply it effectively to support progress and learning.

12. Do your team's practices support systematic inspection?

- 1 Not at all – Team practices don't support inspection.
- 2 Poorly – Practices support limited inspection, which is not systematic.
- 3 Moderately – Practices moderately support systematic inspection.
- 4 Well – Practices support systematic inspection well.
- 5 Very well – Practices strongly support systematic inspection.

13. How well does your team evaluate its progress and identify issues, for example, in sprint reviews and daily stand-ups?

- 1 Not at all – The team cannot evaluate progress or identify issues.
- 2 Poorly – The team evaluates progress and identifies issues poorly, delaying action.
- 3 Moderately – The team evaluates progress and identifies issues but occasionally faces delays or lacks clarity.
- 4 Well – The team evaluates progress effectively and identifies issues in a timely manner.
- 5 Very well – The team evaluates progress thoroughly and identifies issues promptly, ensuring smooth project flow.

14. How quickly and effectively does your team respond to problems or areas for improvement identified during inspections?

- 1 Not at all – The team does not respond to problems or areas for improvement.
- 2 Poorly – The team responds slowly and ineffectively, impacting project progress.
- 3 Moderately – The team responds reasonably well, but occasionally experiences delays.
- 4 Well – The team responds quickly and effectively to identified issues, improving project outcomes.
- 5 Very well – The team responds extremely quickly and effectively, proactively solving problems.

Adaptation

15. How well do you understand what adaptation means in the Scrum context?

- 1 Not at all – I don't understand what adaptation means in Scrum.
- 2 Poorly – I partially understand adaptation but am unclear about its application.
- 3 Moderately – I understand adaptation and its connection to team processes, but lack clarity on practical usage.
- 4 Well – I clearly understand adaptation in Scrum and can apply it in practice.
- 5 Very well – I can explain and apply adaptation in Scrum and recognize its importance for continuous improvement.

16. Do your team's practices support systematic adaptation or improvement based on inspection?

- 1 Not at all – Team practices don't support adaptation or improvement.
- 2 Poorly – Practices support limited adaptation or improvement, and it is not systematic.
- 3 Moderately – Practices moderately support systematic adaptation or improvement.
- 4 Well – Practices support systematic adaptation or improvement well.
- 5 Very well – Practices strongly support systematic adaptation or continuous improvement.

17. How well do ideas for improvement raised in retrospectives lead to concrete changes?

- 1 Not at all – Ideas raised in retrospectives do not lead to changes.
- 2 Poorly – Improvement ideas are rarely implemented.
- 3 Moderately – Improvement ideas sometimes lead to changes but not consistently.
- 4 Well – Improvement ideas usually lead to concrete changes.
- 5 Very well – Improvement ideas always lead to meaningful and implemented changes.

18. Do you feel your team has sufficient autonomy to make necessary changes to working methods or processes?

- 1 Not at all – The team lacks autonomy to make changes.
- 2 Poorly – The team has minimal autonomy.
- 3 Moderately – The team has some autonomy, but changes often require higher-level approval.
- 4 Well – The team has enough autonomy to make necessary changes independently.
- 5 Very well – The team is fully autonomous and can make changes without restrictions.

19. How well does your organization support your team in making quick changes when adaptation is required?

- 1 Not at all – The organization doesn't support quick changes.
- 2 Poorly – Organizational support is limited, making quick changes difficult.
- 3 Moderately – The organization provides moderate support for quick changes.
- 4 Well – The organization supports the team well in making quick changes.
- 5 Very well – The organization fully supports and facilitates quick changes.

The Scrum values

Commitment

20. Do you feel that all members of your team are committed to the goals agreed upon during the sprint?

- 1 Not at all – Most team members are not committed to the goals.
- 2 Poorly – Only a few team members are committed to the goals.
- 3 Moderately -- Most of the team is committed, but there is a lack of commitment.
- 4 Well – Almost all team members are committed to the goals.
- 5 Very well – All team members are fully committed to the goals.

21. How important do you consider commitment to the sprint goals set by the team?

- 1 Not important – I don't think commitment matters much.
- 2 Somewhat unimportant – Commitment is only slightly significant.
- 3 Moderately important – Commitment is important but not critical.
- 4 Very important – Commitment is highly important for the team's success.
- 5 Extremely important – Commitment is absolutely critical for the team's success.

Courage

22. How confidently do you share your opinions and suggestions for improvement with your team?

- 1 Not at all – I do not feel confident sharing my views.
- 2 Poorly – I rarely share my views or suggestions.
- 3 Moderately – I share my views occasionally, but not always.
- 4 Well – I frequently share my views and suggestions.
- 5 Very well – I always openly share my views and suggestions with the team.

23. Do you feel your team encourages courage in addressing problems or proposing new solutions?

- 1 Not at all – My team does not encourage courage.
- 2 Poorly – Courage is minimally supported in the team.
- 3 Moderately – The team encourages courage to some extent.
- 4 Well – The team actively encourages courage.
- 5 Very well – The team fully supports and encourages courageous behavior.

Focus

24. Does your team have a clear understanding of what to focus on during the sprint?

- 1 Not at all – The team lacks clarity about sprint priorities.
- 2 Poorly – The team has a vague idea of priorities, but it is unclear.
- 3 Moderately – The team generally understands priorities, but clarity could improve.
- 4 Well – The team has a clear understanding of sprint priorities.
- 5 Very well – The team is fully focused and clear about sprint priorities.

25. How often is your team's workflow disrupted by additional tasks or unexpected interruptions?

- 1 Never – There are no interruptions, and work proceeds smoothly.
- 2 Rarely – Interruptions are infrequent and minimally disruptive.
- 3 Moderately – Interruptions occur but do not significantly affect work.
- 4 Often – Interruptions occur frequently but are manageable.
- 5 Very often – Interruptions are constant and disruptive.

Openness

26. How open do you consider communication and information sharing within your team?

- 1 Not at all open – Communication and information sharing are very poor.
- 2 Poorly open – There is limited openness in communication.
- 3 Moderately open – Communication and information sharing are somewhat open but need improvement.
- 4 Well open – Communication is mostly open, and information sharing is effective.
- 5 Very well open – Communication and information sharing are fully open and seamless.

27. How openly does your team discuss failures and learn from them together?

- 1 Not at all – The team does not discuss failures openly.
- 2 Poorly – Failures are rarely discussed openly.
- 3 Moderately – Failures are discussed but conversations lack depth.
- 4 Well – Failures are openly discussed and used constructively.
- 5 Very well – The team openly discusses and learns from failures collaboratively.

Respect

28. Do team members show respect for each other's work and opinions?

- 1 Not at all – There is no respect among team members.
- 2 Poorly – Respect is rarely shown among team members.
- 3 Moderately – Team members generally show respect, but it is inconsistent.
- 4 Well – Team members mostly show respect for each other.
- 5 Very well – Respect among team members is consistent and genuine.

29. Do you feel that shared Scrum values influence the success of the team?

1 Not at all – Scrum values do not influence the team's success.

2 Very little – The influence of Scrum values on the team's success is minimal.

3 Somewhat – Scrum values influence the team's success to some extent but are not a significant factor.

4 Quite a lot – Scrum values have a clear and important impact on the team's success.

5 Very much – Scrum values play a central role in the team's success.

Current Scrum practices

30. Do you feel that Ponsse's current Scrum practices support your work?

No

Yes

31. How useful do you find the following Scrum events in your work?

Sprint planning, Daily scrum meetings, Sprint review, Sprint retrospective, Backlog refinement

1 = Not useful at all – The event adds no value to your work or the team's performance.

2 = Slightly useful – The event adds minimal value.

3 = Moderately useful – The event is useful in some situations but not consistently impactful.

4 = Very useful – The event adds significant value to your work and team performance.

5 = Essential – The event is crucial and highly beneficial for daily work or team performance.

32. What do you consider the biggest challenges in the current Scrum practices?

33. What do you think works well in the current Scrum practices?

Information for a person participating in a study

You are taking part in a study project related to a thesis that is being prepared at the Savonia University of Applied Sciences. This privacy statement describes how your personal data will be processed during the study.

Participation in the study is voluntary. No negative consequences will ensue from your refusal to participate in the study or if you later withdraw from the study. If you withdraw from participating in the study, the material compiled prior to your refusal can, however, be used in the study. Section 14 of this statement provides more detailed information on your rights and how you can influence the processing of your data.

1. Data controller of the study

Name: Emilia Hakomäki

Telephone: 040 5687225

E-mail address: emilia.hakomaki@edu.savonia.fi

2. Description of the study project and the purpose of processing personal data

The aim of the study is to examine the Scrum framework within Ponsse's Digital Services development teams. The research questions are:

1. How well do Ponsse's development teams understand Scrum theory: its pillars and values?
2. How do developers perceive the current Scrum practices at Ponsse in supporting their work?
3. How did the training period provided to the development team change developers' views on Scrum and working within the Scrum framework?

The study consists of three parts:

1. **Initial survey** for all Digital Services development teams at Ponsse.
2. **Intervention** with a selected development team, during which the team will review Scrum's theoretical foundation and their current Scrum practices.
3. **Final survey** for the development team that participated in the intervention.

The data for the study will be collected using a questionnaire implemented via the Webropol service. A general link to the survey will be distributed, and respondents' email addresses will not be collected. The collected data will be stored on the researcher's computer; it will not be saved in a cloud service or on an external hard drive. During the analysis phase, any identifiable information will be removed, and in the reporting, individual respondents cannot be identified. The collected data will be destroyed after the thesis has been assessed, published, and appeal periods have expired.

The researcher will also participate in the Scrum events of the team involved in the intervention, making observations about the team's activities. However, these observations will not be included in the research data; instead, they will serve as support for planning retrospectives. The Miro.com digital collaboration platform will be used as a tool during the retrospectives, and any data potentially collected from it will be included in the study. Separate consent forms have been obtained from the individuals participating in the intervention for this purpose.

3. Persons performing the study

The study has only one researcher, Emilia Hakomäki.

4. The title and the running time of the study

Title of the study: Scrum Competence and Practices in Ponsse's Development Teams

The duration of processing personal data: The study will be conducted in January-May 2025.

5. Legal basis for processing personal data

Personal data will be processed based on the grounds as stated in the Article 6(1) of the General Data Protection Regulation.

In this study, the processing of personal data is based on the following grounds:

- Consent given by the research subject (data subject)
- Scientific or historical research or compilation of statistics

6. Personal data contained in the study material

The study collects only data relevant to the research. As personal data, indirect identifiers such as work experience and the development team in which the respondent works are collected about the participants. During the analysis and reporting phases, the respondents' data will be anonymized, and identifiers, such as the names of development teams, will be replaced with pseudonyms (e.g., "Team A") to ensure that individual respondents cannot be identified. The data will be stored on the researcher's personal computer. The collected data and survey form will be deleted from the Webropol platform once the thesis has been evaluated, published, and the statute of limitations has expired.

7. Special categories of personal data (sensitive personal data)

Personal data belonging to special categories (sensitive personal data) will not be processed in this study.

8. Sources that are used for the collection of personal data

Personal data is collected from the data subject.

9. Transfer or onward disclosure of the personal data

Personal data will not be transferred or disclosed further.

10. Transfer or disclosure of personal data outside the EU or the European Economic Area

Personal data will not be transferred or disclosed outside the EU/EEA.

11. Principles of protecting personal data

- Personal data is confidential

Processing of direct identifiers:

- Direct identifiers will be removed in the analysis phase
- The material will be analysed containing direct identifiers

12. Processing of personal data after the completion of the study

- The study register will be destroyed once the thesis has been evaluated, published, and the appeal period has expired.
- The study register will be filed:
 - without identifying data
 - with identifying data

The place where the material will be filed and the duration of filing:

13. Contact information of the Data Protection Officer (DPO) of the Savonia University of Applied Sciences

The Data Protection Officer of the Savonia University of Applied Sciences is Administrative Coordinator Mervi Häätinen. You can contact her by e-mail at tietosuojavastaava@savonia.fi.

14. Your rights and derogations from the rights

The person mentioned in the Section 1 of this Privacy Statement is the contact person in matters concerning the rights of the data subject.

Withdrawal of consent (Article 7 of the GDPR)

You have the right to withdraw your consent if the processing of personal data is based on consent. Withdrawal of consent does not affect the lawfulness of the processing based on consent prior to its withdrawal.

Right of access by the data subject (Article 15 of the GDPR)

You have the right to obtain confirmation as to whether your personal data is being processed in the study or not, and which categories of your personal data will be processed. If you wish, you may ask the controller to provide you with a copy of your personal data undergoing processing.

Right to the rectification of personal data (Article 16 of the GDPR)

If your personal data undergoing processing is inaccurate, you have the right to ask that the data to be rectified or completed.

Right to the erasure of personal data ('right to be forgotten') (Article 17 of the GDPR)

You have the right to require your personal data to be erased where one of the following grounds applies:

- a) the personal data are no longer necessary in relation to the purposes for which they were collected or otherwise processed;
- b) you withdraw the consent upon which the processing is based, and where there is no other legal ground for the processing;
- c) you object to the processing (the description of the right to object is given further below) and there are no overriding legitimate grounds for the processing;
- d) the personal data have been unlawfully processed; or
- e) the personal data have to be erased for compliance with a legal obligation in Union or Member State law to which the controller is subject.

The right to erasure shall not apply to the extent that the erasure of personal data is likely to render impossible or seriously impair the achievement of the objectives of processing the data for scientific research purposes.

Right to the restriction of processing (Article 18 of the GDPR)

You have the right to require the restriction of processing your personal data where one of the following applies:

- a) you contest the accuracy of the personal data, whereupon the processing will be restricted for a period enabling the author of the study to verify the accuracy of the data in question;
- b) the processing is unlawful, and you oppose the erasure of the personal data and request the restriction of their use instead;
- c) the author of the study no longer needs the particular personal data for the purposes of the processing, but you need them for the establishment, exercise or defence of legal claims; or
- d) you have objected to the processing of your personal data (see more specifically below) pending the verification whether the legitimate grounds of the controller override those of the data subject.

Right to data portability (Article 20 of the GDPR)

You have the right to obtain the personal data you have provided to the author of the study in a structured, commonly used and machine-readable format and you have the right to transmit the data to another controller, where the processing is based on consent or on a contract, and where the processing is carried out by automated means.

In exercising your right to data portability, you shall have the right to have your personal data transmitted directly from one controller to another, where technically feasible.

Right to object (Article 21 of the GDPR)

You have the right to object to the processing of your personal data, if the processing is based on general public interest or for the purposes of legitimate interests. The author of the study is no longer allowed to process your personal data unless he or she demonstrates compelling legitimate grounds for the processing which override the interests, rights and freedoms of the data subject or for the establishment, exercise or defence of legal claims. The author of the study can proceed with processing your personal data also where it is necessary for the performance of a task carried out in the public interest.

Derogations of rights

Derogations of the rights described in this section may be allowed in certain individual cases on the grounds laid down by the General Data Protection Regulation and Finnish Data Protection Act in cases where these rights render impossible or seriously impair the achievement of the objectives in the scientific or historical research purposes or statistical purposes. The necessity for derogations of rights will always be assessed case-specifically.

Right of appeal

You have the right to file in a complaint at the Office of the Data Protection Ombudsman if you find that the current data protection legislation has been violated in the processing of your personal data.

Contact information:

Office of the Data Protection Ombudsman

Visiting address: Lintulahdenkuja 4, 00530 Helsinki

Postal address: PL 800, 00531 Helsinki

Switchboard: 029 566 6700

Telefax: 029 566 6768

E-mail: tietosuoja(at)om.fi

LIITE 3: SUOSTUMUSLOMAKE INTERVENTIOON OSALLISTUMISESTA

Consent Form for Participation in Research

Research Title: Scrum Competence and Practices in Ponsse's Development Teams

Researcher: Emilia Hakomäki, emilia.hakomaki@edu.savonia.fi

The data for the study will be collected using a questionnaire implemented via the Webropol service. A general link to the survey will be distributed, and respondents' email addresses will not be collected. The researcher will also participate in the Scrum events of the team involved in the intervention, making observations about the team's activities. However, these observations will not be included in the research data; instead, they will serve as support for planning retrospectives.

The [Miro.com](https://miro.com) digital collaboration platform will be used as a tool during the retrospectives. It is possible that materials produced during the retrospectives will be used as part of the research data. This Forms survey serves as a separate consent for the use of Miro materials.

The collected data will be stored on the researcher's computer; it will not be saved in a cloud service or on an external hard drive. During the analysis phase, any identifiable information will be removed, and in the reporting, individual respondents cannot be identified. The collected data will be destroyed after the thesis has been assessed, published, and appeal periods have expired.

With my signature, I confirm that I am aware of the objectives of the study and the ways in which the material is processed. I agree to participate in the research under the above conditions.

* Required

1. I authorize the collection of research material from the [Miro.com](https://miro.com) digital collaborative platform in relation to this research. *

Yes

No

2. Name and Date *

Enter your answer

Submit

Never give out your password. [Report abuse](#)

 Microsoft 365

This content is created by the owner of the form. The data you submit will be sent to the form owner. Microsoft is not responsible for the privacy or security practices of its customers, including those of this form owner. Never give out your password.

Microsoft Forms | AI-Powered surveys, quizzes and polls [Create my own form](#)

[Privacy and cookies](#) | [Terms of use](#)

LIITE 4: MIRO-TAULUT

