

Lotta Kallonen

**KETTERYYDEN OMAKSUMISEN ARVIOINTI
JULKISHALLINNOLLISSA ORGANISAATIOSSA**



JYVÄSKYLÄN YLIOPISTO
INFORMAATIOTEKNOLOGIAN TIEDEKUNTA
2022

TIIVISTELMÄ

Kallonen, Lotta

Ketteryyden omaksumisen arviointi julkishallinnollisessa organisaatiossa

Jyväskylä: Jyväskylän yliopisto, 2022, 137 s.

Tietojärjestelmätiede, pro gradu -tutkielma

Ohjaajat: Abrahamsson, Pekka; Kultanen, Joni; Laakkonen Sari

Moni organisaatio käyttää ohjelmistokehityksessä ketteriä menetelmiä tuottaakseen asiakkaalleen asiakasarvoa mahdollisimman usein. Ketterien menetelmien taustalla toimii ketterät arvot ja periaatteet, jotka ohjaavat erilaisia ketterien menetelmien viitekehyskiä, mutta viitekehysten käyttöönotossa on omat haasteensa ja organisaatio päätyykin omaksumaan viitekehysten eikä ketteryyden ideologiaa. Tässä pro gradu -tutkielmassa arvioidaan kuinka hyvin julkishallinnollinen organisaatio, Kela, on onnistunut omaksumaan ketteryyden arvot ja periaatteet. Tutkielman tavoitteena oli tunnistaa myös osa-alueet, jotka vaativat kehittämistä ja myös kolmantena tutkittavana asiana oli tunnistaa, sopiiko kypsyystasomalli julkishallinnolliseen organisaation ketteryyden kypsyystason arvioimiseen.

Tutkimuksessa hyödynnettiin laadullista tutkimusmenetelmää, jossa tutkimusstrategiana käytettiin tapaustutkimusta. Empiirinen aineisto kerättiin teemahaastattelulla. Tutkimusaineisto koostuu 8 haastattelusta ja haastateltaviksi valittiin henkilöitä, joiden katsottiin ymmärtävän, minkälaisia onnistumisia ja haasteita Kelalla on ketteryyden suhteen.

Tutkimuksen tulokset osoittavat, että Kela on onnistunut omaksumaan joitakin ketteriä arvoja ja periaatteita. Kypsyystasomallin mukaisesti Kela on tasolla 1 tai 2 asteikon ollessa 1-5. Organisaation osa-alueista eniten kehittämistä vaativat asiakkaan tunnistaminen, asiakkaan ymmärtäminen sekä asiakasarvon luominen mahdollisimman usein. Myös yksilöiden ja tiimien valmentaminen kohti ketteryyttä tunnistettiin tärkeänä kehittämiskohteena. Tutkielman puitteissa tunnistettiin myös ketteryyden omaksumista estäviä kulttuurillisia tekijöitä, joita ovat Kelassa johtaminen, asenteet toisia työntekijöitä kohtaan sekä työympäristössä vallitsevat arvot. Tutkielman perusteella tunnistettiin myös kypsyystasomallin sopivuus ketteryyden kypsyystason arviointiin, vaikka mallissa tunnistettiin muutamia epäkohtia.

Asiasanat: ketteruus, ketterät menetelmät, organisaatiokulttuuri, SAFe, julkishallinnollinen organisaatio

ABSTRACT

Kallonen, Lotta

Assessing the adoption level of Agile in a Public Sector Organization in Finland
Jyväskylä: University of Jyväskylä, 2022, 137 pp.

Information Systems, Master's Thesis

Supervisors: Abrahamsson, Pekka; Kultanen, Joni; Laakkonen Sari

Many organizations use agile methods in software development to create customer value for their customers as often as possible. Agile methods are based on agile values and principles that guide the different frameworks of agile methods, but the implementation of frameworks has its own challenges, and the organization ends up adopting the ideology of the framework rather than agility. This master's thesis assesses how well a public administration organization, Kela, has succeeded in embracing the values and principles of agility. The aim of the study was also to identify the areas that need to be developed, and the third issue to be examined was to determine whether the maturity level model is suitable for assessing the maturity level of a public organization.

The study follows qualitative research method and case study was utilized as its primary research method. Empirical material was collected through thematic interviews. The research material consists of 8 interviews. People who were selected to be interviewed were considered to understand the successes and challenges that Kela has in terms of agile.

The results of the study show that Kela has succeeded in adopting some agile values and principles. According to the maturity level model, Kela is at level 1 or 2 with a scale of 1-5. Of the areas of the organization that require the most development is identifying the customer, understanding the customer and creating customer value as often as possible require development. Coaching individuals and teams towards agility was also identified as an important area for development. The study also identified cultural factors that prevent the adoption of agility, such as leadership, attitudes towards other employees and basic underlying assumptions. Based on the study, the suitability of the maturity level model for assessing the maturity level of agility was also identified, although a few drawbacks were identified in the model.

Keywords: agile, agile methodologies, organizational culture, SAFe, public sector organization

KUVIOT

KUVIO 1 Ketteryys ajattelutapa (Measey 2015, s.11)	14
KUVIO 2 Perinteinen ja ketterä rautakolmio (Measey, 2015, 18-19).....	15
KUVIO 3 Ketterän kulttuurin tasot (Tolfo ym. 2009, s. 430)	17
KUVIO 4 Kuvaus SAFe viitekehyksestä (<i>Portfolio SAFe</i> , Scaled Agile Inc., 2022)	21
KUVIO 5 Käytännön tarkkuustasot (Jacobson ym., 2017).....	23
KUVIO 6 Kelan kehittämismalli (Kehittämismalliryhmä, 2022)	27
KUVIO 7 SAFe Maturity Model: tasot, periaatteet ja käytännöt (Turetken ym., 2016).....	29

TAULUKOT

TAULUKKO 1 Ketterän ohjelmistokehityksen julistuksesta löytyvät painokohdat.....	12
TAULUKKO 2 Kiinteän ja ketterän ajattelutavan erot (Measey, 2015)	14
TAULUKKO 3 Ketterien menetelmien omaksumisen haasteen julkishallinnollisessa organisaatiossa (Nuottila, Aaltonen & Kujala, 2016).....	23
TAULUKKO 4 Kypsyystason käytänteiden kuvaukset (Turetken ym., 2016)	29
TAULUKKO 5 Ohjelmoijan osaamistasot (Cockburn, 2002; Boehm & Turner, 2003) 35	
TAULUKKO 6 Aineiston kautta nousseet empiirisen johtopäätökset	48
TAULUKKO 7 Asiakasarvon tuottamisen empiirisen johtopäätökset.....	57
TAULUKKO 8 Suunnittelun ja jatkuvan tuottamisen empiiriset johtopäätökset	74
TAULUKKO 9 Ihmiskeskeisyyden empiirisen johtopäätökset	87
TAULUKKO 10 Teknisen huippuosaamisen empiirisen johtopäätökset	103
TAULUKKO 11 Asiakasyhteistyön empiirisen johtopäätökset.....	114
TAULUKKO 12 Ensisijaiset empiirisen johtopäätökset.....	115
TAULUKKO 13 Organisaation ketteryyden kypsyys: Taso 1	117
TAULUKKO 14 Organisaation ketteryyden tulokset ensimmäisen kypsyystason osalta.....	117
TAULUKKO 15 Organisaation ketteryyden kypsyys: Taso 2	117
TAULUKKO 16 Organisaation ketteryyden tulokset toisen kypsyystason osalta 118	
TAULUKKO 17 Organisaation ketteryyden kypsyys: Taso 3	118
TAULUKKO 18 Organisaation ketteryyden tulokset kolmannen kypsyystason osalta 119	
TAULUKKO 19 Organisaation ketteryyden kypsyys: Taso 4	119
TAULUKKO 20 Organisaation ketteryyden tulokset neljännen kypsyystason osalta 120	
TAULUKKO 21 Organisaation ketteryyden kypsyys: Taso 5	120

TAULUKKO 22	Organisaation ketteryyden tulokset viidennen kypsyystason osalta	121
TAULUKKO 23	Organisaation ketteryyden tulokset.....	121

SISÄLLYS

TIIVISTELMÄ

ABSTRACT

KUVIOT JA TAULUKOT

SISÄLLYS

1	JOHDANTO.....	9
2	KETTERYYS JA KETTERÄT MENETELMÄT	11
2.1	Ketteryys ohjelmistokehityksessä	11
2.2	Ketteryys ajattelutapana ja organisaatiokulttuurillisena ilmiönä	13
2.2.1	Perinteisen ja ketterän projektinhallinnan erot.....	15
2.2.2	Ketterät menetelmät ja organisaatiokulttuuri.....	16
2.3	Laajamittaiset ketterät menetelmät	17
2.3.1	Laajamittaisen ketteryyden omaksumisen haasteet ja menestystekijät	18
2.3.2	SAFe	20
2.3.3	Essence	21
2.4	Julkishallinnon erityispiirteet ja haasteet ketterissä menetelmissä	23
2.5	Julkishallinnollisen organisaation kehittämismalli	26
2.6	SAFe MM	27
2.6.1	Ketteryyden kypsyytasomalli.....	27
2.6.2	SAFe MM tasot, periaatteet ja käytännöt.....	28
2.6.3	Cockburnin henkilöstötasot.....	35
2.6.4	SAFe MM mallin käyttäminen	36
2.6.5	Millä tapaa SAFe MM mallia on sovellettu aikaisemmassa kirjallisuudessa?	37
3	TUTKIMUSMENETELMÄ	38
3.1	Tutkimuksen tavoite ja tutkimusmenetelmä.....	38
3.2	Kohdeorganisaation esittely.....	39
3.3	Tiedonkeruumenetelmä.....	39
3.3.1	Haastateltavien valinta ja haastateltavien taustatiedot	41
3.3.2	Haastattelujen suunnittelu ja toteutus	42
3.4	Aineiston analysointi.....	43
4	EMPIIRISEN TUTKIMUKSEN TULOKSET.....	45
4.1	Organisaatiokulttuuri ja toimintatavat ketteryyden omaksumisen esteenä	45
4.2	Asiakasarvon tuottaminen	48
4.2.1	Prosessin reflektointi ja parantaminen.....	50
4.2.2	Vaatimusten evolutiivisuus	51
4.2.3	Pienemmät, useammat julkaisut	52

4.2.4	Vaatimusten löytäminen	53
4.2.5	Säännöllinen reflektointi ja sopeutuminen.....	53
4.2.6	Asiakaslähtöiset iteraatiot	54
4.2.7	Jatkuva asiakastyytyväisyyspalaute.....	55
4.2.8	Leanin vaatimukset skaalautuvassa mittakaavassa.....	56
4.2.9	Matala prosessiseremonia	56
4.2.10	Jatkuva SAFe kyvykkyyksien parantaminen.....	57
4.3	Suunnittelu ja jatkuva tuottaminen.....	58
4.3.1	Yhteissuunnittelu	58
4.3.2	Jatkuva tuottaminen	59
4.3.3	Kahden tason suunnittelu ja seuraaminen	60
4.3.4	Ketteryyden arviointi ja kiertonopeus	60
4.3.5	Julkaisun suunnittelu.....	62
4.3.6	Riskilähtöiset iteraatiot.....	63
4.3.7	Ominaisuuksien, eikä tehtävien suunnittelu	64
4.3.8	Tiekartta.....	65
4.3.9	Iteraation hallitseminen.....	66
4.3.10	Kanbanin hyödyntäminen	68
4.3.11	PSI / julkaisu	69
4.3.12	Julkaisujuna.....	70
4.3.13	Pienemmät ja useammin toistuvat julkaisut	71
4.3.14	Adaptiivinen suunnittelu.....	72
4.3.15	Liiketoiminnan tehokkuuden mittaaminen	73
4.3.16	Ketterien projektien arviointi	73
4.4	Ihmiskeskeisyys	75
4.4.1	Yhteistyötä tekevät tiimit	75
4.4.2	Voimaantuneet ja motivoituneet tiimit	77
4.4.3	Määrittely/Ohjelmointi/testaus -tiimi	80
4.4.4	Itseohjautuvat tiimit.....	81
4.4.5	Säännöllinen henkilökohtainen vuorovaikutus.....	82
4.4.6	Scrumien scrum	83
4.4.7	Hajautettujen tiimien hallinta.....	84
4.4.8	Ihanteellinen fyysinen ympäristö ketterälle kehittämiselle	84
4.4.9	Organisaation muuttaminen	85
4.5	Tekninen huippuosaaminen	88
4.5.1	Koodausstandardit.....	88
4.5.2	Tiedon jakaminen	89
4.5.3	Vapaaehtoisuus tehtävissä.....	89
4.5.4	Hyväksymistestaus	90
4.5.5	Sovelluskokoonpanon hallinta.....	91
4.5.6	Automaatiotestaus	91
4.5.7	Iteraation edistymisen seuranta	93
4.5.8	Ei isoa suunnittelua etukäteen	93
4.5.9	Tuotteen kehitysjono	95
4.5.10	Jatkuva integraatio.....	96

4.5.11	Jatkuva parantaminen (refaktorointi)	96
4.5.12	Yksikkötestit.....	97
4.5.13	30% tason 2 ja tason 3 ihmisiä	98
4.5.14	Tarkoituksenmukainen arkkitehtuuri.....	99
4.5.15	Päivittäiset edistymisen seurantakokoukset.....	100
4.5.16	Testilähtöinen kehittäminen.....	101
4.5.17	Tason 1B tai -1 ihmisiä vähän tai ei ollenkaan tiimissä	102
4.5.18	Rinnakkainen testaaminen	102
4.6	Asiakasyhteistyö	104
4.6.1	Käyttäjätarinat	107
4.6.2	Asiakas sitoutuu työskentelemään kehitystiimin kanssa.....	109
4.6.3	Asiakassopimus heijastaa evolutiivista kehitystä	110
4.6.4	DevOps	110
4.6.5	Visio, ominaisuudet	111
4.6.6	Vaikutus asiakkaisiin ja toimintoihin.....	112
4.6.7	Asiakas on heti käytettävissä.....	113
4.6.8	Asiakasyhteys pyörii yhteistyöhön sitoutumisen ympärillä...113	
4.6.9	Toistuva kasvokkain tapahtuva vuorovaikutus kehittäjien ja käyttäjien välillä	114
4.7	Yhteenveto	115
5	TULOSTEN TULKINTA JA POHDINTA	116
5.1	Kelan kypsyystasomallin tulokset.....	116
5.2	SAFe MM kypsyysmallin arviointi	123
5.3	Tutkimuksen rajoitteet	124
5.4	Jatkotutkimusaiheita	125
6	YHTEENVETO	127
	LÄHTEET	129
	LIITE 1 SAFE MM VIITTAUKSET	131
	LIITE 2 SAFE MM -MALLI: TASOT, PERIAATTEET JA KÄYTÄNNÖT (TURETKEN YM., 2016)	135
	LIITE 3 TIETOA TUTKIMUKSESTA HAASTATELTAVILLE	136
	LIITE 4 SUOSTUMUSLOMAKE HAASTATELTAVILLE.....	137

1 JOHDANTO

Moni organisaatio on omaksunut käyttöönsä SAFen eli Scaled Agile Frameworkin, jonka avulla isojen organisaatioiden on mahdollista kehittää, johtaa sekä organisoida toiminnan kehittämistä ketterillä menetelmillä (Alqudah & Razali, 2016). Menetelmän käyttöönoton jälkeen on kuitenkin epäselvää, kuinka hyvin organisaatio on omaksunut uuden mallin käyttöönsä. Tähän ongelmaan vastaa kypsyysmalli, jonka avulla mitataan organisaation skaalautuvan ketteryyden kypsyyttä (Turetken, Stojanov & Trienekens, 2016).

Viitekehyksen onnistunut käyttöönotto ja omaksuminen vaikuttaa laajasti niin organisaation tuottavuuteen, tuotteiden laatuun, työhyvinvointiin kuin kulujen minimointiin (Putta, Paasivaara, & Lassenius, 2018). Varsinkin isoissa julkishallinnollisissa organisaatioissa viitekehyksen onnistuneella omaksumisella on laajoja vaikutuksia organisaation lisäksi myös koko yhteiskuntaan, kun palveluita pystytään tuottamaan kansalaisille nopeammin ja tehokkaammin. Pro gradu -tutkielmassa etsitään vastausta seuraaviin tutkimuskysymyksiin:

- Kuinka hyvin Kela on onnistunut omaksumaan ketterien menetelmien arvot ja periaatteet Turetkenin ym. (2016) kypsyystasomallin perusteella?
- Mitkä organisaation osa-alueet vaativat kehittämistä kypsyystasomallin perusteella?
- Soveltuuko Turetkenin ym. (2016) SAFe MM -malli ketteryyden kypsyystason mittaamiseen?

Tutkimus keskittyy tunnistamaan, kuinka hyvin Kela on onnistunut omaksumaan ketterien menetelmien arvot ja periaatteet. Toisena tutkielman tavoitteena on tutkia sitä, kuinka hyvin Turetkenin ym. (2016) SAFe MM kypsyystasomalli soveltuu julkishallinnollisen organisaation kypsyystason mittaamiseen eli kasvattaa ymmärrystä teoreettisen viitekehyksen sopivuudesta empiriaan.

Monesti ketteryyttä ja ketteriä menetelmiä otetaan käyttöön työkalujen ja prosessien kautta, jolloin ketteryys jäädä saavuttamatta, kun ketteryyttä toteutetaan. Ketteryys tarkoittaa ennen kaikkea tietynlaista ajattelutapaa. (Measey, 2015) Ketterään ajattelutapaan kuuluu muutoksen nopea aikaansaaminen, ennakoiminen tai reagoiminen muutokseen sekä muutoksesta oppiminen samalla asiakasarvoa luoden (Conboy, 2009)

Kirjallisuuskatsauksessa hyödynnettiin Google Scholaria tunnistamaan kypsyystasomalleja, joilla organisaation kypsyyttä voisi mitata. Hakusanoina käytettiin avainsanoja "Agile", "SAFe" ja "Maturity Model". Näiden hakusanojen perusteella löytyi Turetkenin ym. (2016) SAFe MM -malli, jolla organisaation ketteryyden kypsyystasoa on mahdollista arvioida. Tutkielman empiirinen aineisto kerättiin teemahaastattelujen avulla. Haastatteluihin osallistui yhteensä 8 henkilöä. Haastateltavien valinnassa hyödynnettiin lumipallo-otantaa, jossa aluksi tunnistettiin ketteryyden avaintekijöitä (Tuomi & Sarajärvi, 2009). Aineiston analysoinnissa käytettiin suunnattua sisällönanalyysia, jossa sisällönanalyysissa lähtökohtana toimii teoreettinen viitekehys, joka ohjaa koodien muodostamista ja lähestymistavan tavoitteena on validoida tai laajentaa teoreettista viitekehystä tai teoriaa (Hsieh & Shannon, 2005). Tässä teoreettisena viitekehysenä aineiston analysoinnissa käytettiin Turetkenin ym. (2016) SAFe MM -mallia.

Tutkimus keskittyy organisaation IT-liitännäiseen tekemiseen ja teknisen näkökulman sijaan tutkimuksen ensisijaisena tutkittavan aiheena on ketteryys organisatorisena ja kulttuurisena ilmiönä. Lisäksi skaalautuvissa ketterissä menetelmissä keskitytään erityisesti niihin viitekehyksiin, joita hyödynnetään kohdeorganisaatiossa. Tutkimuksen ja esimerkiksi kirjallisuuskatsauksen ulkopuolelle jätetään ne viitekehukset, joita kohdeorganisaatio ei ole hyödyntänyt kehittäessään omaa kehittämismallia.

Tutkielma on jaoteltuna kuuteen eri lukuun. Luku 2 pitää sisällään kirjallisuuskatsauksen, jossa määritellään tarkemmin ketteryys ja ketterän ohjelmistokehityksen arvot ja periaatteet, ketterien menetelmien viitekehyksiä, julkishallinnollisen organisaation erityispiirteet, kohdeorganisaatiossa käytetty kehittämisen malli sekä SAFe MM -malli.

Luvussa 3 kuvataan tarkemmin käytetty tutkimusmenetelmä, tiedonkeruumenetelmä sekä aineiston analysointimenetelmä, joita hyödynnettiin empiirisessä osiossa. Tässä luvussa kerrotaan myös haastateltavien valinnasta, haastattelujen suunnittelusta ja toteutuksesta.

Luvussa 4 pitää sisällään empiiriset tutkimustulokset, jotka on jaoteltuna tarkemmin teoriapohjalta löytyneisiin teemoihin. Luvussa 5 on tulosten tulkinta sekä pohdinta, vastaukset tutkimuskysymyksiin sekä jatkotutkimusaiheet. Viimeisenä lukuna on luku 6, jossa tutkielman sisältö on tiivistetty yhteen. Viimeisenä tutkielmassa löytyvät lähdeluettelo ja liitteet 1, 2, 3 ja 4, jotka sisältävät SAFe MM viittaukset, SAFe MM -mallin tasot, periaatteet ja käytännöt, tietoa tutkimuksesta haastateltaville sekä suostumuslomake haastateltaville.

2 KETTERYYS JA KETTERÄT MENETELMÄT

Tässä luvussa käsitellään, mitä ketteryydellä, ketterillä menetelmillä sekä skaalautuvilla ketterillä menetelmillä tarkoitetaan tässä tutkielman kontekstissa. Luvussa esitellään julkishallinnollisen organisaation erityispiirteet, Kelassa käytetty kehittämisen malli sekä SAFe MM -malli.

2.1 Ketteryys ohjelmistokehityksessä

Käsitteinä ketteryys ja ketterät menetelmät ovat laajasti käytössä, mutta käsitteille ei löydy yhtä oikeaa määrittelyä. Käsitteet ovat monitahoisia, minkä vuoksi määritelmiä käytettäessä saatetaan viitata eri asioihin (Conboy, 2009). Eri ketterät menetelmät keskittyvät eri ohjelmistokehitysvaiheiden tukemiseen ja eri laajuudessaan, mikä osaltaan vaikeuttaa yhtenäisen määritelmän löytämistä (Abrahamsson ym., 2002).

Conboyn (2009) mukaan ketteryydeksi voidaan määritellä muutoksen nopea aikaansaaminen, ennakoiminen tai reagoiminen muutokseen, sekä muutoksesta oppiminen samalla asiakasarvoa (taloudellisuutta, laatua ja yksinkertaisuutta) luoden. Tämän lisäksi määritelmässä korostetaan sitä, että ketteryys voi tapahtua vuorovaikutuksessa ympäristönsä muiden järjestelmien kanssa ja oleellista on, että ketteryyttä pitää tarkastella koko yrityksen laajuisessa kontekstissa, eli pelkkien organisaation osien ketteröittäminen ei riitä, jotta ketteryyden määritelmä täyttyisi. (Conboy, 2009)

Ketterille menetelmille ei löydy aikaisemmasta kirjallisuudesta yhtä ainoaa määritelmää, mutta se mikä näitä määrittelyjä yhdistää on ketterän ohjelmistokehityksen julistus (engl. *Manifesto for Agile Software Development*). Ketterä ohjelmistokehityksen julistus on julkaistu vuonna 2001 ohjelmistokehittäjien toimesta ja se koostuu neljästä eri arvosta ja kahdestatoista periaatteesta (Beck ym., 2001). Nämä arvot ovat:

- Yksilöitä ja kanssakäymistä enemmän kuin menetelmiä ja työkaluja

- Toimivaa ohjelmistoa enemmän kuin kattavaa dokumentointia
- Asiakasyhteistyötä enemmän kuin sopimusneuvotteluja
- Vastaamista muutokseen enemmän kuin pitäytymistä suunnitelmassa (Beck ym., 2001)

Laanti, Similä ja Abrahamsson (2013) ovat analysoineet, mitä ketterän ohjelmistokehityksen julistuksen kahdessatoista periaatteissa korostetaan. Nämä painokohdat löytyvät alla olevasta taulukosta (TAULUKKO 1) Kohtia löytyy yhteensä 22, mutta koska asiakastyytyväisyys ja asiakasetu ja aikainen tuottaminen ja tiheä tuottaminen ovat hyvin lähellä toisiaan, nämä ovat yhdistetty seuraavanlaisesti: asiakastyytyväisyys/asiakasetu ja aikainen tuottaminen/tiheä tuottaminen. Yhdistelyn jälkeen kohtia löytyy 20, jotka ovat:

TAULUKKO 1 Ketterän ohjelmistokehityksen julistuksesta löytyvät painokohdat

Periaatteen painotus

1. asiakastyytyväisyys/ asiakasetu
 2. jatkuva tuottaminen
 3. arvontuotto
 4. aikainen tuottaminen, tiheä tuottaminen
 5. mukautumiskyky
 6. kilpailukyky
 7. yhteistyö
 8. motivoituneet yksilöt
 9. hyvä ympäristö
 10. tuki
 11. luottamus
 12. tehokkuus
 13. kommunikointi
 14. tuotteen edistymisen mittaaminen
 15. kestävyys (engl. *sustainability*)
 16. ihmiset
 17. keskittyminen tekniseen erinomaisuuteen
 18. yksinkertaisuus
 19. työn optimointi
 20. sisäänrakennettu tehokkuuden ja käyttäytymisen parantaminen (Laanti, Similä & Abrahamsson, 2013)
-

Abrahamssonin, Salon, Ronkaisen ja Warstan (2017) mukaan ohjelmistokehityksen eri ketteriä menetelmiä yhdistävät seuraavat piirteet: inkrementaalisuus, yhteistyökykyisyys, suoraviivaisuus sekä adaptiivisuus. Inkrementaalisudella tarkoitetaan sitä, että ohjelmistoa tehdään pienissä sykleissä. Yhteistyökykyisyydessä on tärkeää, että asiakas sekä kehittäjät toimivat jatkuvassa tiiviissä yhteistyössä. Suoraviivaisuudella viitataan siihen, että ketterän menetelmän oppiminen ei ole monimutkaista ja menetelmän

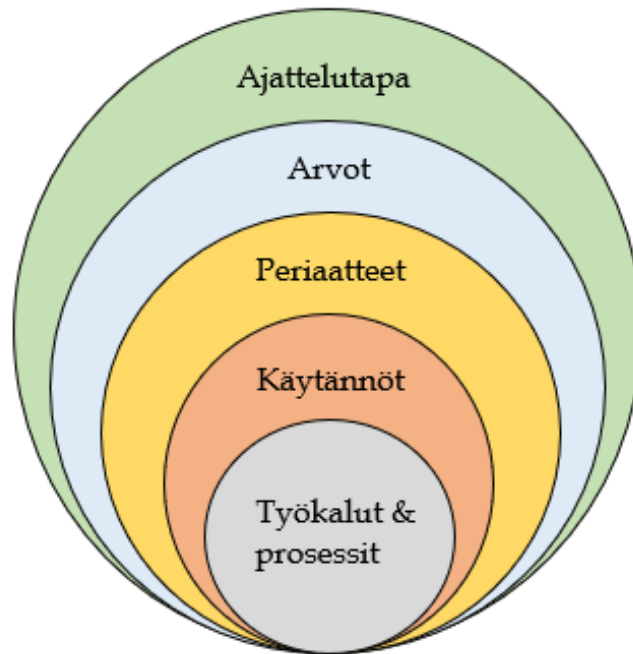
soveltaminen on mahdollista pienellä vaivalla. Adaptiivisuudella tarkoitetaan kykyä tehdä viimehetken muutoksia. (Abrahamsson ym., 2002)

2.2 Ketteryys ajattelutapana ja organisaatiokulttuurillisena ilmiönä

Measeyn (2015) on kuvannut ketteryyden tasot sipulimaisena kuvana, jossa sipulin sisimmästä osasta ulospäin ovat työkalut ja prosessit, käytännöt, periaatteet, arvot sekä ajattelutapa. Ketteryys tarkoittaaakin ennemmin ketterällä ajattelutavalla asioiden lähestymistä kuin ketteryyden toteuttamista esimerkiksi työkalujen ja prosessien avulla. (Measey, 2015).

Ketteryydessä työkalun ja prosessin avulla pyritään saamaan avoin näkymä nykyiseen tilanteeseen: yhdellä näkymällä pitäisi saada käsitys siitä, miten asiat edistyvät ja mikä toimituksen tila on. Yleisin periaate ketterissä viitekehyksissä on visuaalisten taulujen käyttö. Visuaalinen taulu voi olla esimerkiksi fyysinen valkotaulu tai virtuaalinen Kanban-tili. Visuaalisilla tauluilla pyritään toteuttamaan läpinäkyvyyden periaatetta eli taulun kautta pitäisi pystyä näkemään, mikä on tekemisen nykytilanne: kuinka lähellä valmistumista tehtävä on. Muutakin tietoa, kuten mikä tiimi tai kuka työskentelee tehtävän parissa, voi sisällyttää näkymään. Läpinäkyvyyden periaate tukee puolestaan ketteryyden "vastaaminen muutoksiin" -arvoa eli avoimuudella pystytään saamaan muutostarve ja siitä aiheutuvat muutokset näkyväksi. Kaikki nämä edellä mainitut tasot kuuluvat ketterään ajattelutapaan. (Measey, 2015)

Mitä kuva (KUVIO 1) yrittää visualisoida on se, että vaikka yksilö, tiimi tai organisaatio omaksuisi esimerkiksi ketteryyteen liittyvät työkalut ja prosessit, ei se tee toiminnasta ketterää, ellei asioita lähestytä myös ketterällä ajattelutavalla. Ketteryys on ennen kaikkea organisaation muutosmatka kohti ketterää ajattelutapaa, joka lähtee ketteryyden arvojen ja periaatteiden ymmärtämisestä. (Measey, 2015).



KUVIO 1 Ketteryys ajattelutapa (Measey 2015, s.11)

Muutosmatkaan saattaa aluksi kuulua käytäntöjen tai prosessien mekaaninen suorittaminen, mutta muutoksen edetessä organisaatiossa pitäisi alkaa näkyä muutosta niin käyttäytymisessä kuin kulttuurissa. Ketterän prosessit, metodit, ja käytännöt on luotu sitä varten, että organisaatio pystyisi omaksumaan ketterän ajattelutavan. Jos organisaatio ei sitoudu ketterän ajattelutavan muutokseen, ketteriä prosesseja ja metodeja suoritetaan jatkossakin vain mekaanisesti. Organisaatioiden tulisi siis ymmärtää, että ketteryyttä ei voi saavuttaa ilman ketterää käyttäytymistä, mikä edellyttää ketteryyden arvojen ja periaatteiden omaksumista. Muutosmatka tarkoittaa kulttuurimuutosta, jossa ihmisten käyttäytymisessä tapahtuu muutos vastauksena organisaation arvojen ja oletusten muutokseen. (Moreira, 2017)

Muutosmatkoilla, joissa ketteryyden käytännöt, periaatteet ja arvot omaksutaan vain osittaisella menestyksellä johtaa tilanteeseen, missä tiimit ymmärtävät, mitä tehdä, mutta eivät ymmärrä miksi. Tällöin tiimi palaa todennäköisesti aiempiin työskentelytapoihin, ja muutos kohti ketteryyttä keskeytyy. (Measey, 2015) On siis olennaista että, yksilö, tiimit ja organisaatio jaksavat sitoutua ketterän ajattelutavan omaksumiseen ja toisaalta organisaation tulisi ymmärtää, että muutosmatka kohti ketteryyttä on kulttuurimuutos. (Moreira, 2017) Taulukossa (TAULUKKO 2) on kuvattuna kiinteän ja ketterän ajattelutapojen eroja. Muutos kiinteästä ajattelutavasta ketterään ajattelutapaan on iso, sillä maailmaa katsotaan melkein jopa päinvastaisista näkökulmista.

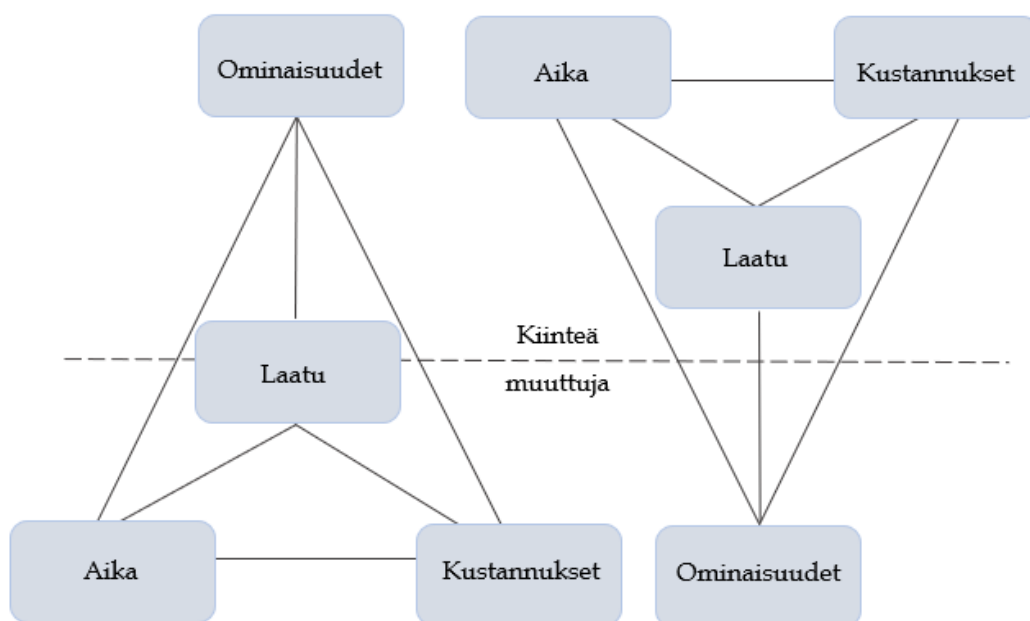
TAULUKKO 2 Kiinteän ja ketterän ajattelutavan erot (Measey, 2015)

Kiinteä ajattelutapa	Ketterä ajattelutapa
----------------------	----------------------

Kyvykkyys	Staattinen, kuten korkeus	Voi kasvaa, kuten lihas
Tavoite	Näyttää hyvältä	Oppia
Haaste	Välttää	Omaksua
Epäonnistuminen	Määrittelee identiteetin	Tarjoaa informaatiota
Vaivannäkö	Koskee niitä, joilla ei ole lahjakkuutta	Polku mestaruuteen
Reaktio haasteeseen	Avuttomuus	Resilienssi, sinnikkyys

2.2.1 Perinteisen ja ketterän projektinhallinnan erot

Perinteisessä projektinhallinnassa käytetään usein rautakolmiota (engl. *Iron triangle*) kuvaamaan projektin rajoitteita: mitä ominaisuuksia pitää saada aikaan, missä ajassa ja millä budjetilla? Measey lisäsi rautakolmioon vielä yhden osa-alueen: teknisen laadun. Kuvassa (KUVIO 2) on kuvattuna perinteisen ja ketterän projektinhallinnan erot.



KUVIO 2 Perinteinen ja ketterä rautakolmio (Measey, 2015, 18-19)

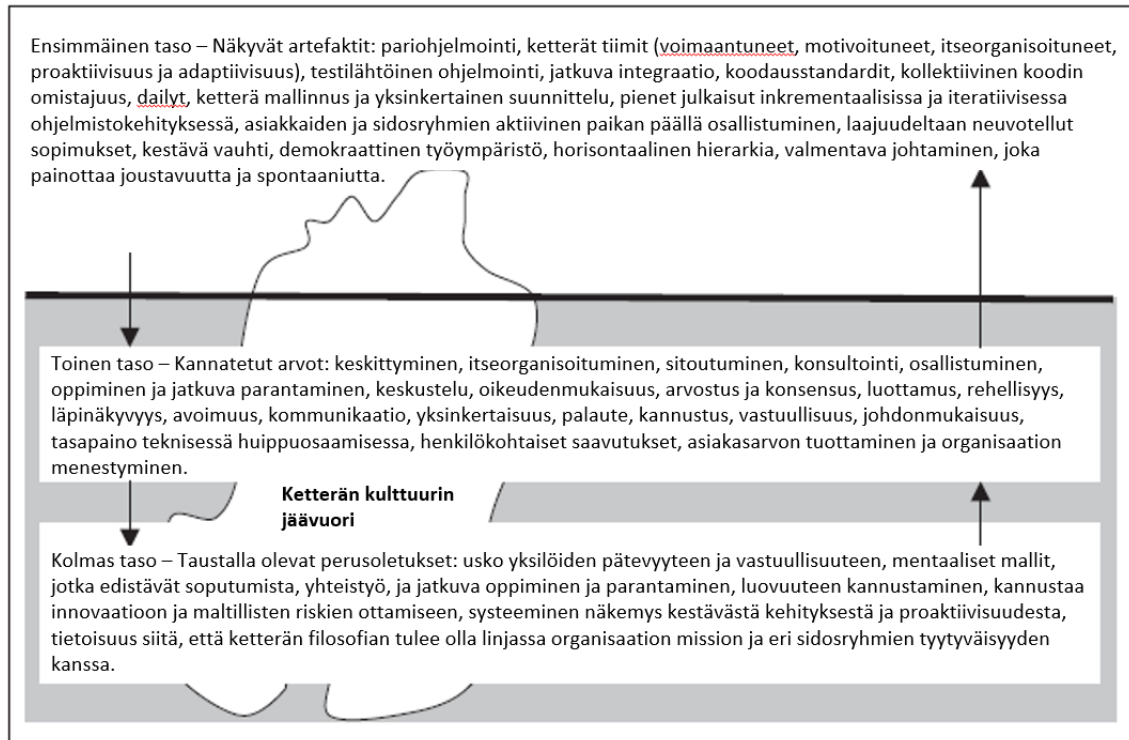
Kuvan (KUVIO 2) vasemmalla puolella on kuvattuna perinteisen projektinhallinnan lähestymistapa, jossa kaikki ominaisuudet on määritelty etukäteen ja projektin tavoitteena on tuottaa kaikki etukäteen määritellyt ominaisuudet. Projektin rajoitteista aika, laatu sekä kustannukset voivat muuttua. Jos asiakas päättäisi muuttaa ominaisuuksia, projekti pitäisi määritellä uudelleen, mikä todennäköisesti tarkoittaisi sitä, että projekti venyy ja kustannukset kasvavat huomattavasti. Hidastunutta projektin etenemistä

saatetaan paikata uusilla työntekijöillä, mikä hidastuttaa edelleen projektin etenemistä sekä nostaa projektin kustannuksia, kun olemassa olevan tiimin aikaa kuluu uuden tai uusien työntekijöiden perehdyttämiseen. Perinteiseen projektihallintaan liittyy myös toinen riski: jos ominaisuudet muuttuvat, mutta aikataulusta halutaan yhä pitää kiinni, tekninen laatu todennäköisesti kärsii. Esimerkiksi testausaika saattaa jäädä lyhyemmäksi tai testaus jää priorisoimatta, arkkitehtuurit ja suunnitelmat saattavat vaarantua ja lopulta ohjelmiston saaminen toimintakuntoon millä tahansa laadulla johtaa tekniseen velkaan. (Measey, 2015)

Kuvan (KUVIO 2) oikealla puolella on kuvattuna ketterän projektinhallinnan lähestymistapa, josta löytyvät kaikki samat osiot kuin perinteisessäkin projektinhallinnassa, erona se, että rautakolmio on käännetty ylösalaisin. Tällöin aika, kustannukset ja laatu ovat kiinteitä muuttujia, ja ainut asia mikä muuttuu on ominaisuudet ja niiden laajuus. Tässäkin mallissa laatu voi joustaa, mutta se tapahtuu aina tietoisella päätöksellä siitä, että laadun muuttaminen aiheuttaa teknistä velkaa. Ketterässä projektinhallinnassa on olemassa eri aikalaatikoita (engl. *timeboxing*), kuten projektin aikalaatikko, julkaisun aikalaatikko ja iteraation aikalaatikko. Näistä aikalaatikoista iteraation aikalaatikko on kiinteä, eli tiimi sitoutuu tuottamaan kyseisessä aikalaatikossa arvoa asiakkaalle. Joillan sprinteillä asiakasarvoa pystytään tuottamaan enemmän, joillakin vähemmän. Se, mikä ero tällä lähestymistavalla on perinteiseen projektinhallintaan on se, että ketterässä projektinhallinnassa tavoitetta ei ole tarkkaan määritelty etukäteen, vaan se voi muuttua projektin edetessä esimerkiksi asiakkaan tarpeiden muuttuessa. (Measey, 2015)

2.2.2 Ketterät menetelmät ja organisaatiokulttuuri

Tolfo, Wazlawick, Ferreira ja Forcellini (2009) ovat tutkineet erilaisia organisaatiokulttuureita teoreettisen viitekehyksen avulla tunnistuen, mitkä tekijät organisaatiokulttuurissa voivat vaarantaa ketterien menetelmien omaksumisen. Tolfo ym. (2009) käyttivät oman teoreettisen viitekehyksen luomisessa Scheinin (1999) organisaatiokulttuurin eri tasoja. Kuviossa (KUVIO 3) on kuvattuna ketterän kulttuurin eri tasot.



KUVIO 3 Ketterän kulttuurin tasot (Tolfo ym. 2009, s. 430)

Jäävuoren huippu kuvastaa ketterään kulttuuriin kuuluvia näkyviä artefakteja, kuten ketterän kehityksen käytänteitä ja teknologioita. Jos taustalla olevat perusolettamukset eivät tue näitä ketterän kehityksen arvoja, ketteristä käytänteistä ei tule pysyviä. Jäävuoren toinen taso kuvastaa niitä ketterään kulttuuriin kuuluvia asioita, jotka eivät ole näkyvissä. Kolmas taso kuvastaa organisaatiokulttuurin syvintä olemusta: taustalla vallitsevat perusoletukset, jotka muovaavat ja ohjaavat organisaatiokulttuuria. Esimerkiksi demokraattista johtamistyyliä ja valmentavaa johtamista voidaan pitää ketterän kulttuurin ilmentyminä. Sen lisäksi, että organisaatiokulttuuri koostuu kolmesta eri syvyystasosta, organisaatiokulttuuri jakaantuu myös strategiseen, taktiseen että operatiiviseen tasoon. (Tolfo ym. 2009)

Ketteryyden omaksumisen esteenä toimiikin usein organisaatiokulttuurissa piilossa olevat asiat. Ketteriä menetelmiä omaksuessa olisikin tärkeä, että organisaatio tunnistaisi ja ymmärtäisi organisaation kulttuuria ja siihen liittyvät näkökulmat. Oleellista on tunnistaa, onko organisaation nykyisen kulttuurin ja ketterän kulttuurin välillä ristiriitoja, mitkä näkyvät arvoissa, uskomuksissa ja käyttäytymisessä. (Tolfo ym. 2009)

2.3 Laajamittaiset ketterät menetelmät

Ketteryyttä on tutkittu myös isommassa mittakaavassa, sillä ketteryyden määritelmä tiimitasolla voi tarkoittaa eri asiaa kuin laajemmasta näkökulmasta

katsottuna. Conboy ja Carroll (2019) nostavat esiin sen, että laajamittaisille ketterille menetelmille ei välttämättä löydy yhtä määritelmää ja käsitteen määrittely koetaan hankalaksi. Toisaalta jotkut ammatinharjoittajat ovat nostaneet esille sen, että käsitteen määrittely ei pitäisi olla pääosassa: laajamittaiset ketterät menetelmät eivät ole itseisarvo, vaan tarkoituksena on keskittyä siihen, että menetelmä tuo organisaatiolle arvoa ja lyhentää toimitusaikoja. (Conboy & Carroll, 2019).

Alqudahin ja Razalin (2016) mukaan organisaatioissa laajamittaisilla ketterillä menetelmillä pyritään siihen, että alkuperäiset ketterät menetelmät laajennettaisiin koskemaan myös isompia tiimejä, jolloin menetelmää pystyisi hyödyntämään koordinoinnissa, tiimien välisessä sekä tiimien ja asiakkaiden välisessä kommunikoinnissa. Lisäksi menetelmä auttaisi valvonnassa (Alqudah & Razal, 2016).

Laajamittaisten ketterien menetelmien ero pelkkiin ketteriin menetelmiin on tiimien ja ihmisten määrässä. Dikert, Paasivaara ja Lassenius (2016) mukaan kyseessä on laajan mittakaavan ketterä ohjelmistokehitys, kun ohjelmistokehitysorganisaatiossa on vähintään 50 henkilöä tai vähintään kuusi eri tiimiä. Määritelmässä on huomioitu myös se, että kaikkien henkilöiden ei tarvitse olla ohjelmistokehittäjiä, mutta yhteisen tuotteen tai projektin kehittämisen pohjalta heidän tulee kuulua samaan ohjelmistokehitysorganisaatioon ja tehdä yhteistyötä muiden kanssa.

2.3.1 Laajamittaisen ketteryyden omaksumisen haasteet ja menestystekijät

Conboy ja Carroll (2019) ovat tutkineet laajamittaisten ketterien viitekehysten omaksumista 15 vuoden ajan ja he ovat löytäneet näiden pohjalta erilaisia omaksumisen haasteita ja menestystekijöitä. Seuraavaksi esitellään haasteet, jotka voivat vaikuttaa laajamittaisen ketteryyden omaksumiseen.

Ensimmäisenä on tärkeää varmistaa, että skaalautuvan ketterän viitekehysten konsepti ja termit on määritelty. Kun termejä ja konseptia ei ole määritelty riittävän tarkasti tai terminologia on hyvin abstraktia, ihmisten on helpompi jatkaa työskentelyä samalla vanhalla tyyllillä kuin omaksua uusi malli toimia. Esimerkiksi SAFea on kritisoitu siitä, että viitekehyksessä kuvataan peruseriaatteet selkeästi, mutta kun sitä yritetään soveltaa tarkoitettun kontekstin ulkopuolella, viitekehysten käyttämiseen ei ole saatavilla opastusta. Epätarkka terminologia voi aiheuttaa epä johdonmukaisia merkityksiä ja tulkintoja. (Conboy & Carroll, 2019)

Seuraava haaste on oikean viitekehysten valinta. Laajamittaisia ketteryyden viitekehymiä on olemassa useita, kuten SAFe, LeSS ja Scrum at Scale, mutta oikean viitekehysten valintaan ei ole tarjolla kattavaa työkalua. Ongelmaksi nousee se, että viitekehysten valinnassa ei kiinnitetä tarpeeksi huomiota viitekehysten omaksumisen kannalta kriittisiin päätöksiin, vaan viitekehysten valinta tehdään melko kevyesti ja päätöksentekoa ei pysty edes jäljittämään, että millä valinnalla viitekehys valittiin. Viitekehysten valinnassa joku haastateltavista nosti esille, että hänelle jäi sellainen mielikuva, että johtoa ei edes kiinnostanut, mikä viitekehys valitaan, kunhan valinnassa on arvioitu

vaihtoehtoja ja päätös on perusteltu ennen kuin muutokseen on saanut rahoitusta. (Conboy & Carroll, 2019)

Kolmantena haasteena on organisaation valmiuden puute ottaa käyttöön laajamittainen ketterän menetelmän viitekehys. Muutos ketterään vaatii organisaatiolta paljon valmiutta niin henkilöstön kuin rakenteiden puolesta. Varsinkin useiden viitekehysten käyttäminen lyhyellä aikavälillä voi hankaloittaa viitekehysten omaksumista. Joissakin organisaatioissa hypitään viitekehuksesta toiseen niin usein, että kehitystiimit turhautuvat muutoksiin, kun edelliseen viitekehysten käyttöön ei ehdi edes tottua, kun hypätään jo seuraavan viitekehysten käyttämiseen. Tällaisissa tapauksissa kehitystiimit saattava vastustaa viitekehysten omaksumista ja yleistä on passiivinen lähestymistapa eli asioita ei vaan tehdä eri tavalla. Laajamittaisen ketterien menetelmien viitekehysten on kritisoitu siitä, että viitekehysten tarjoavat vain hyvin vähän tukea siihen, miten organisaatiot pystyisivät arvioimaan heidän valmiuttaan käynnistää ketteryyden muutosprosessi. Johtajat saattavat vaistota kehittäjien ja ryhmien välillä valmiuden puutetta, mutta heillä ei ole mekanismeja tunnistaa näitä ongelmia selkeästi. (Conboy & Carroll, 2019)

Neljäntenä haasteena on tasapainottelu organisaatorakenteen ja laajamittaisen viitekehysten välillä. Laajamittaisissa ketteryyden viitekehysten on ennalta määritetyt rakenteet, rutiinit ja työkalut, joita voi olla haastava yhdistää jo olemassa olevan organisaatorakenteen kanssa. Olemassa olevat organisaatorakenteet ovat jatkuvassa muutoksessa ulkoisen kilpailun tai voimakkaasti säännellyn ympäristön takia, minkä vuoksi viitekehysten käyttöönoton ja sen ylläpitämisen yhdistäminen muuttuvaan organisaatorakenteeseen voi olla vaikeaa. Lisäksi laajamittaisen viitekehysten omaksuminen vaatii usein organisaatorakenteiden muuttamista, mikä voi olla erittäin haastavaa. (Conboy & Carroll, 2019)

Viidentenä haasteena on laajamittaisen viitekehysten käyttöönoton yllyttäminen vastaan alhaalta-ylöspäin. Alhaalta-ylöspäin etenevä muutos on tehokkain pieniin Scrum-käyttöönottoihin, mutta laajan mittakaavan viitekehysten käyttöönottoissa asia ei ole niin yksiselitteinen: isoissa muutoksissa ylimmän johdon tuki ja osallistuminen ovat avaintekijöitä onnistumiseen. Alhaalta-ylöspäin etenevissä muutoksissa johdon tuki ja osallistuminen usein puuttuu, mikä johtaa siihen, että tiimit ottavat omia versioita viitekehystä käyttöön. Ylhäältä-alaspäin etenevässä muutoksessa on myös omat ongelmansa. Esimerkiksi se, että samat ihmiset, jotka määräävät, että uusi viitekehys otetaan käyttöön, vaativat edelleen vanhoja raportteja ja viisivuotissuunnitelmia. (Conboy & Carroll, 2019) Esimerkki kuvastaa hyvin sitä ongelmaa, että johtotasolla ei ole tarvittavaa osaamista ja ymmärrystä ketteristä menetelmistä. Usein Scrum-valmentajia on kaikkialla, mutta johto- ja portfoliotasojen SAFE valmentajia on vaikea löytää (Conboy & Carroll, 2019).

Kuudentena haasteena on viitekehysten noudattamisen korostaminen arvontuoton sijaan. Ketteryyden muutosprosessia mitataan usein virheellisesti sen kautta, kuinka hyvin viitekehystä noudatetaan eikä mitä arvoa viitekehysten käyttöönotto on tarjonnut. Ketteryyden muutosprosessia mitataan virheellisesti

kahdesta syystä: ensinnäkin on paljon helpompi mitata, kuinka moni tiimi on ottanut Scrumin käyttöön kuin se, että mitattaisiin sitä, kuinka monta prosenttia ketteryys on kasvanut ja toiseksi mitattaessa voi olla vaikea määrittellä, johtuiko muutos viitekehyksen käyttöönotosta vai muista tekijöistä. Ongelmana viitekehyksen täydellisenä noudattamisessa on myös se, että viitekehyksen täydellinen noudattaminen ei ole välttämättä kannattavaa ja järkevää: usein 80 % stressistä ja ponnistelusta menee viimeisen 5%:n saavuttamiseen. Johtajilla on usein vain hyvin vähän opastusta siihen, mikä on se optimaalinen taso, mihin muutoksessa pyritään. (Conboy & Carroll, 2019)

Seitsemäntenä haasteena nähdään laajamittaisten ketterien viitekehysten näyttöön perustuvan käytön puute. SAFe, Scruma, LeSS ja Scrum at Scale -viitekehysistä puuttuu empiiriset tapaustutkimukset, joiden avulla pystyttäisiin näkemään, miten viitekehukset toimivat käytännössä. Viitekehukset eivät pysty tarjoamaan tukea tai opastusta organisaatioille, joihin liittyy vahvoja kontekstuaalisia ongelmia. Esimerkiksi organisaatiot, joissa on vahvasti säädelty toimia-ala tai tuote ovat lähtökohtaisesti aivan eri asemassa kuin organisaatiot, joissa tuotetaan uusia sovelluksia suoraan loppuasiakkaalle. (Conboy & Carroll, 2019)

Kahdeksas haaste on kehittäjien autonomian säilyttäminen laajamittaisissa ketterissä viitekehysissä. Ohjelmistokehittäjät odottavat ja vaativat autonomiaa työltä: etätömahdollisuus, joustavat työajat, omien työkalujen ja ohjelmistokehitysvälineiden valinta ja jne. Ketterien arvojen vastaisesti laajamittaiset ketterät menetelmät asettavat ja luovat rajoitteita ja jäykkyyttä. Laajamittaisissa ketterissä viitekehysissä autonomian säilyttäminen on haastavaa, sillä jos kaikki työntekijöiden toiveet toteutetaan, viitekehuksesta jää jäljelle vain jäännökset, joita ei pysty tunnistamaan enää edes viitekehukseksi.

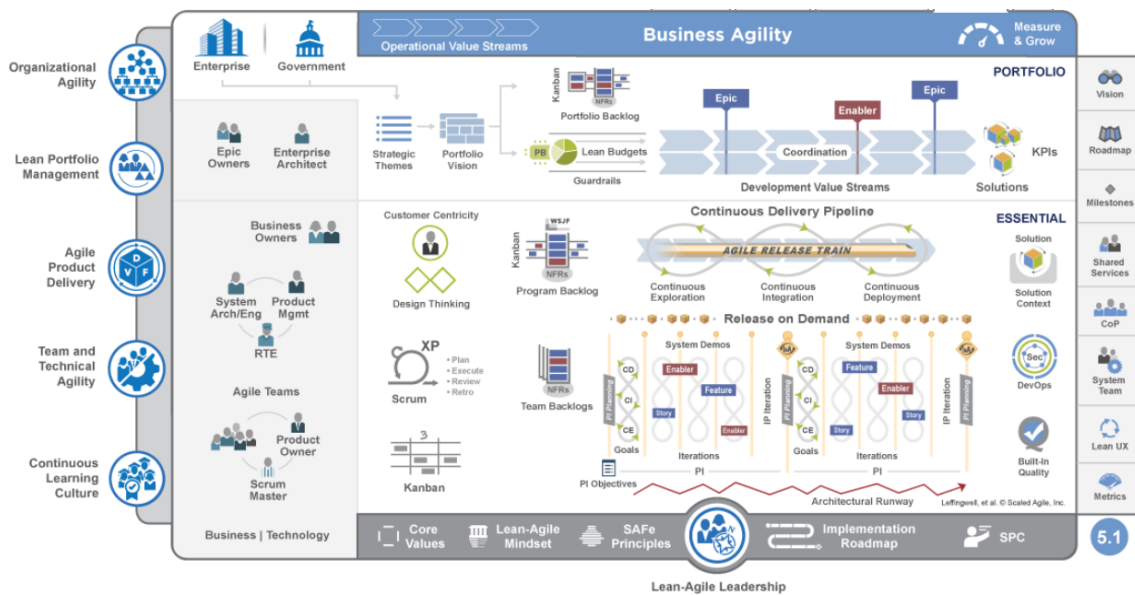
Viimeinen eli yhdeksäs haaste on asiakasprosessien ja laajamittaisten ketterien viitekehysten epäyhteensopivuus. Organisaatioiden odotetaan ottavan asiakkaita enemmän mukaan suunnitteluprosesseihinsa. Laajamittaisissa ketterissä viitekehysissä on paljon käytäntöjä, joita on yhteensovitettava asiakasprosessien kanssa, ja tämä ei ole mutkatonta, jos yhdellä organisaatiolla on satoja asiakkaita pienistä aina isoihin yrityksiin. Joissain tapauksissa organisaatioiden on tarvinnut kehittää erillisiä sopimuksia, joilla pystytään kiertämään viitekehyksen kohtia, jotta yhteistyö ja ohjelmistojen toimittaminen asiakkaille olisi sujuvaa. (Conboy & Carroll, 2019)

Conboyn ja Carrollin (2019) kuvaamista haasteista pystyy näkemään sen, että laajamittaisessa ketterissä viitekehysissä viitekehyksen käyttäminen kääntyy välillä itseään vastaan eli ketteryyden saavuttaminen laajemmassa mittakaavassa ei täyty vaan itseasiassa viitekehys luo välillä niitä esteitä, joita viitekehys pyrkii poistamaan.

2.3.2 SAFe

Scaled Agile Framework (SAFe) on Dean Leffingwellin suunnittelema skaalautuvien ketterien menetelmien viitekehys ja se on yksi yleisimmin käytössä olevista skaalautuvista viitekehysistä (Laanti & Kettunen, 2019).

SAFe koostuu neljästä eri tasosta tai kerroksesta, jotka ovat tiimi-, julkaisu-, arvovirta- sekä portfoliotaso. Tasot ovat linkittyneenä toisiinsa ja jokaisella tasolla on omat tehtävänsä. Tiimitaso sisältää ketterien menetelmien käytänteitä muun muassa Scrumista, Kanbanista, Scrumbanista, DevOpista sekä XP:stä. (Alqudah, Razali, 2016) Julkaisutasolla varmistetaan se, että useat eri ketterät tiimit tekevät yhteistyötä yhteisen päämäärän saavuttamiseksi. Portfoliotasolla hallitaan sitä, että kehittämisen strategia on yhdessä linjassa toteutuksen kanssa. Tämä tapahtuu arvovirtojen avulla. (Leffingwell, 2011b) Alla olevassa kuviossa (KUVIO 4) on kuvattuna SAFe viitekehys portfoliotasolle saakka. Kuvioista on kuvattuna SAFEn eri tasot ja niiden linkittyminen toisiinsa.



KUVIO 4 Kuvaus SAFe viitekehuksesta (*Portfolio SAFe*, Scaled Agile Inc., 2022)

2.3.3 Essence

Essence on Object Management Groupin eli OMG:n luoma standardi ohjelmistokehityksen käytäntöjen ja metodien luomiseen, käyttöön ja parantamiseen. Essencen tarkoituksena on keskittyä olennaiseen informaatioon ja optimoida tekniikan alan sekä tekninen että ihmisläheinen näkökulman tarjoamalla kevyitä käytänteitä. Tarkoituksena essensessä on keskittyä tuloksiin ja samalla minimoida dokumentaation tuottaminen. (Pocket Guide, 2019)

Essencen avulla tiimit voivat itse valita olemassa olevista metodeista ne käytänteet, jotka sopivat juuri heidän kontekstiinsa, ja luoda niiden perusteella oma metodi. Essence toimii siten, että menetelmät tai käytänteet kuvataan käyttämällä Essencen kieltä, jolla pyritään kuvaamaan se, mikä menetelmässä tai käytännössä on oleellista. (Jacobson, Lawson, Ng, McMahon & Goedicke, 2017)

Essencen perustana ovat seuraavat oivallukset:

1. Menetelmät ovat käytäntöjen yhdistelmiä

- a. Menetelmien määrä maailmassa arvioidaan olevan valtava (>100 000), osa näistä menetelmistä on tunnustettu ja niillä on laaja käyttäjäkunta
 - b. Vain muutama sata käytänteistä on uudelleenkäytettäviä. Teoreettisesti mahdollisten yhdistelmien määrä voi kasvaa hyvin suureksi.
2. Kaikkien näiden menetelmien ja käytänteiden välillä on yhteinen ydin tai perusta.
 3. Antaessa ohjeita menetelmän tai käytännön käyttöön, keskity olennaiseen.
 4. Tarjoa sitouttava käyttäjäkokemus, kun opetat ja opit metodeja ja käytänteitä. (Jacobson ym., 2017, 38)

Yhdistellessä eri käytänteitä yhdeksi metodiksi menetelmäasiantuntija tarkistaa, miten nämä käytänteet sopivat yhteen: onko näissä käytänteissä olemassa olevia päällekkäisiä osia tai ristiriitoja keskenään. Menetelmäasiantuntijan tehtävänä on ratkaista nämä mahdolliset ongelmat. Usein yhdistellessä useampia käytänteitä yhdeksi menetelmäksi päällekkäisyyksiä tai ristiriitoja ei yleensä kuitenkaan ole. Päällekkäisyyksien ja ristiriitojen ratkaiseminen ei ole kuitenkaan yksinkertaista, sillä käytännöissä voi olla osia, jotka linkittyvät toisiin komponentteihin. Tällöin menetelmäasiantuntijan tulee ratkaista se, miten kaksi eri käytäntöä pystytään yhdistämään mielekkäällä ja rakentavalla tavalla. (Jacobson ym., 2017)

Käytäntöä voidaan ajatella minimimetodina, joka ohjaa tiimiä suorittamaan yksi tietty asia heidän työssään. Esimerkiksi kehittäessä ohjelmistoa testaaminen on vain yksi osa-alue, työhön kuuluu paljon muutakin: vaatimusmäärittely, suunnittelu, toteutus, projektin hallinta, jne. Käytäntöjä on mahdollista luoda yksin kokemuksen perusteella, mutta menetelmän luominen vaatii vähintään sitä, että muista menetelmistä lainataan käyttöön hyväksi todettuja osia. Menetelmä on yhdistelmä käytäntöjä, joiden tarkoituksena on ohjata tiimi tekemään kaikki ne asiat, joita tarvitaan ohjelmistokehityksessä. Menetelmä voidaan ajatella olevan täydellinen opastus, joka tarvitaan, jotta kaikki osat työstä tulee tehtyä. Menetelmä koostuu useasta käytännöstä, mutta käytäntökin voi koostua useasta käytännöstä: esimerkiksi Scrum koostuu dailysta, tuotteen kehitysjonosta, retrospektiivistä ja sprintin suunnittelusta. (Jacobson ym., 2017)

Käytännön tehtävänä on ehdottaa, mitä tiimin tulee tehdä. Se, kuinka yksityiskohtaisesti käytäntö kuvataan, riippuu tiimin kyvykkyyksistä ja taustasta. Kyvykkyydellä tarkoitetaan ihmisten kykyä selvittää jokin asia olemassa olevan tiedon perusteella. Taustalla viitataan tiimin yhteiseen taustaan aikaisemmista olemassa olevista käytännöistä. Mitä enemmän tiimi on työskennellyt keskenään, sitä todennäköisempää on, että tiimin jäsenillä on yhteinen ymmärrys niistä käytänteistä, mitä on kokeiltu. Jos tiimillä ei ole yhteistä historiaa, sitä tarkemmalla tasolla käytänteet tulisi kuvata, jotta vältetään väärintulkintoja. (Jacobson ym., 2017) Seuraavassa kuviossa on kuvattuna käytännön tarkkuustasot kyvykkyyden ja taustan perusteella (KUVIO 5).

Kyvykkyys	Korkea	Hiljainen käytäntö riittää	Tarvitaan selkeitä käytänteitä
	Matala	Hiljaiset käytänteet valmennuksen kanssa	Selkeät käytänteet valmennuksen kanssa
		Yleinen	Erilainen
		Tausta	

KUVIO 5 Käytännön tarkkuustasot (Jacobson ym., 2017)

2.4 Julkishallinnon erityispiirteet ja haasteet ketterissä menetelmissä

Nuottila, Aaltonen ja Kujala ovat tutkineet, mitä haasteita julkishallinnollisilla organisaatioilla on, kun ketteriä menetelmiä yritetään omaksua. Nuottila ym. (2016) tutkivat yhden viraston kohdalla näitä haasteita, mutta haasteiden voidaan nähdä koskettava yleisesti koko julkishallintoa. Haasteita on tunnistettu yhteensä seitsemän, jotka ovat jaoteltuna tarkemmin alakohtiin. Tunnistettuja haasteita ovat dokumentointi, koulutus, osaaminen ja sitoutuminen, sidosryhmäkommunikointi ja osallistuminen, roolit ketterässä kokoonpanossa, lainsäädäntö, ohjelmistoarkkitehtuurin ja järjestelmänintegraation kompleksisuus. Alla olevassa taulukossa (TAULUKKO 3) on esiteltynä, mitä kaikkia ketterien menetelmien omaksumisen haasteita julkishallinnollisessa organisaatiossa on sekä tarkemmin haasteen kuvaus.

TAULUKKO 3 Ketterien menetelmien omaksumisen haasteen julkishallinnollisessa organisaatiossa (Nuottila, Aaltonen & Kujala, 2016)

Haaste	Haasteen kuvaus
Dokumentointi	Isossa julkishallinnollisessa organisaatiossa tarve informaation jakamiselle on paljon suurempi kuin pienessä julkisessa organisaatiossa. Dokumentointia on tehtävä, jotta järjestelmää on mahdollista ylläpitää, esimerkiksi se takia, kun ihmiset jossain vaiheessa poistuvat organisaatiosta. Toisaalta julkishallinnollisessa organisaatiossa on useita eri ulkoisia käyttäjäryhmiä, jotka tarvitsevat dokumentaatiota.
Koulutus, osaaminen, sitoutuminen -	Ketterien menetelmien käyttöönotto tapahtuu usein ICT osaston kautta, jolloin muut tiimit osallistuvat ketterien menetelmien omaksumiseen myöhemmin. Usein se, että

a. Organisaation valmius ja sitoutuminen	ketterät menetelmät tuodaan ICT osaston tai johdon kautta, koetaan organisaatiossa ongelmaiseksi. Monesti käyttöönoton suunnittelu ja organisaation valmius ottaa malli käyttöön on puutteellista.
Koulutus, osaaminen, sitoutuminen - b. Henkilöstön koulutus ja sitoutuminen	Ketterien menetelmien käyttöönotto tapahtuu kouluttamalla eri rooleissa olevia henkilöitä. Alussa vain muutama osallistuu pilotointiin, jonka jälkeen ketterät menetelmät otetaan laajemmin käyttöön hyvin nopeasti. Alussa koulutus on riittämätöntä. Toisaalta yhä useampi sellainen henkilö, jolle ohjelmistokehitys ja ketterät menetelmät eivät ole tuttuja, altistuvat uusille menetelmille, monesti nämä henkilöt ovat liiketoiminta-aloilta.
Koulutus, osaaminen, sitoutuminen - c. Ketteryyden tuntemus, tietoisuus menetelmästä ja kokemus	Vesiputousmalli perustuu tarkkaan määriteltyyn prosessiin, jossa painotetaan määrittelyä ja dokumentointia. Ketterät menetelmät perustuvat ajatukseen taitavista, innovatiivisista yksilöistä ja tiimeistä, jotka ratkaisevat ongelmia yhdessä hyödyntäen tiimin jäsenten vahvuuksia ja kommunikointia. Ajatuksen muutos vesiputousmallista ketteriin menetelmiin on iso, mikä vaatii organisaatiokulttuurinmuutosta.
Sidosryhmäkommunikointi ja osallistuminen - a. Ketterä suunnittelu vs. sidoryhmäkommunikointi	Ohjelmistojärjestelmien välillä on keskinäinen riippuvuus, jolloin muutokset on ennakoitava riittävän ajoissa, jotta kaikki siihen liittyvät organisaatiot pystyvät tekemään tarvittavat muutokset. Rajapintoja ei kuitenkaan määritellä välttämättä riittävän ajoissa, jotta asiasta voisi kommunikoida tarpeeksi ajoissa sidosryhmille.
Sidosryhmäkommunikointi ja osallistuminen - b. Ketterät menetelmät mahdollistavat joustavat muutokset vs. sidoryhmäkommunikointi	Koska ketterä suunnittelu tapahtuu iteratiivisesti sprintiltä toiseen, on mahdollista, että joitakin alustavasti sovittuja suunnitteluvaihtoehtoja täytyy muuttaa ja toteutuksen järjestystä on muutettava tai ominaisuuksia jää tekemättä. Jos näihin muutoksiin tarvitaan sidosryhmää, haasteena on, että sidosryhmä saattaa saada tiedon liian myöhään.
Sidosryhmäkommunikointi ja osallistuminen - c. Ketterät prosessit vs. sidosryhmän sitoutuminen	Sidosryhmien saaminen mukaan suunnittelutyöhön on haastavaa. Koska ominaisuuksia tuotetaan julkaisusykleissä, asiakkaiden ja loppukäyttäjien ottaminen mukaan pilottitestaukseen on monimutkainen tehtävä.
Roolit ketterässä kokoonpanossa - a. Tuoteomistajan rooli	Onnistumisen kannalta on oleellista, että tuoteomistaja on aina tiimin saatavilla ja hän pystyy tarjoamaan tiimille selkeästi määritellyt käyttäjäkokemukset tai vaatimukset hyvissä ajoissa. Lisäksi tuotteen kehitysjonon priorisointi ja tiimin tukeminen sprintin sisällön valitsemisessa on keskeisiä tuoteomistajan tehtäviä. Tuoteomistajan rooli voi olla haastava, jos rooli on henkilölle uusi ja tiheä yhteistyö ihmisten ja tiimin kanssa ei ole tuttua.
Roolit ketterässä kokoonpanossa - b. Tuoteomistajien useat liittymäkohdat	Tuoteomistaja tekee yhteistyötä scrummasterin ja tiimin kanssa, mutta tietoa hän hankkii liiketoiminta-alueiden omistajilta, teknisiltä asiantuntijoilta, oikeudellisilta neuvonantajilta ja jne. Tuoteomistajan panos tiimin, sisäisten sidosryhmien saatavuus ja heidän toimittamiensa tiedot ovat riippuvaisia toisistaan.
Roolit ketterässä kokoonpanossa -	Liiketoiminnan tuoteomistaja vastaa liiketoiminnan vaatimuksista ja ICT tuoteomistaja vastaa

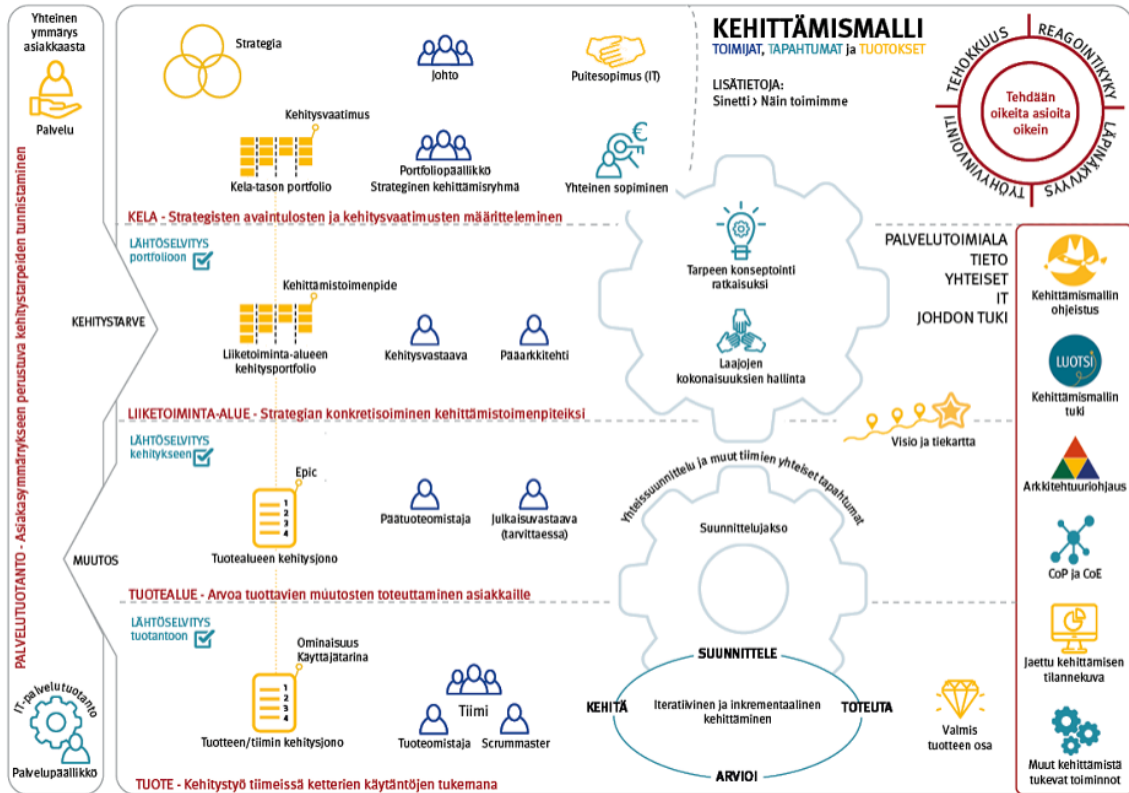
c. Liiketoiminta tuoteomistaja vs. ICT tuoteomistaja	järjestelmävaatimuksista ja teknisistä kysymyksistä. Kahden tuoteomistajan käyttäminen voi aiheuttaa sekaannusta tuotteen omistajuudesta ja vastuusta.
Roolit ketterässä kokoonpanossa - d. Projektijohtaja vs. liiketoiminta tuoteomistaja	Projektipäällikkö omistaa projektin budjetin, mutta ketterässä toimintamallissa tuoteomistaja hallinnoi budjettia ja käyttää sitä ominaisuuksien luomiseen. Riskinä on se, että tuoteomistaja jatkaa toimimista projektipäällikön roolissa, joka luo jännitteitä ja ristiriitoja muiden roolien välille.
Roolit ketterässä kokoonpanossa - e. Scrummaster vs. liiketoiminta tuoteomistaja	Scrummaster ja tuoteomistaja tekevät avainrooleja Scrumin näkökulmasta. Tuoteomistaja on vastuussa siitä, mitä tehdään ja scrummaster puolestaan siitä, miten se toteutetaan. Roolien välisessä yhteistyössä ja rooliassa voi olla epäselvyyksiä.
Ketterien tiimien sijainti	Etätiimeissä yhteistyö ei välttämättä ole samalla tasolla kommunikoinnin suhteen, kun muut paikalliset tiimit.
Lainsäädäntö - a. Julkisia hankintoja koskeva laki	Suomen julkisia hankintoja koskevassa laissa säädetään, minkälaista ja miten tietoa yrityksistä voidaan käyttää yritysten arviointiin. Lain tavoitteena on taata oikeudenmukainen ja neutraali asema kaikille tarjouksiin osallistuville yrityksille. Ketterien menetelmien onnistunut omaksuminen vaatii kuitenkin luottamuksen rakentamista ja hyvää dialogia organisaatioiden välillä, joten lainsäädäntö voi olla este tai hidaste optimaalisen toimittajan valinnalle.
Lainsäädäntö - b. Tiedon jakaminen	Julkisia hankintoja koskeva laki saattaa pahimmassa tapauksessa estää tietojen jakamisen kaikille sidosryhmille tarpeen mukaan. Ketteryyden kannalta tämä on ongelmallista, sillä ketterät menetelmät perustuvat avoimeen viestintään ja informaation jakamiseen.
Lainsäädäntö - c. Uuden lainsäädännön ajoitus	Lait tulevat voimaan usein ennalta määrättyinä päivinä, joka tarkoittaa sitä, että laki asettaa ohjelmistoprojektille tavoitepäivämäärän, kuten vesiputousmallissa asetetaan. Ketteriin menetelmiin kuuluu jatkuva integroiminen, joten tällainen lähestymistapa luo eroa ketterään projektidynamiikkaan.
Ohjelmistoarkkitehtuurin ja järjestelmäintegraation kompleksisuus - a. Ohjelmistonarkkitehtuurin kompleksisuus	Palveluiden arkkitehtuuri on usein monimutkainen ja siinä on monia rajapintoja ja integraatioita. Ketterät menetelmät on alun perin tarkoitettu pienille ja eristetyille järjestelmille, joten järjestelmän monimutkaisuus aiheuttaa omat haasteensa ketteryyden omaksumiselle.
Ohjelmistoarkkitehtuurin ja järjestelmäintegraation kompleksisuus - b. Järjestelmäintegraation kompleksisuus	Järjestelmien kompleksisuus tarkoittaa sitä, että järjestelmien kanssa voi olla useita eri yrityksiä, jotka osallistuvat kahdenvälisiin sopimuksiin ja kahdenvälisiin palvelutasosopimuksiin. Rahapintojen muutosten ja päivitysten tekeminen vaikuttaa siis useampaan tahoon, mikä on haastavaa ketterille menetelmille.

2.5 Julkishallinnollisen organisaation kehittämismalli

Kohdeorganisaation kehittämismallissa on yhdisteltynä useiden skaalautuvien ketterien menetelmien viitekehyksiä, minkä vuoksi organisaation kehittämismallia ei ole mahdollista arvioida sellaisenaan (Kehittämismalliryhmä, 2022). Kehittämismallin taustalla vaikuttaa kuitenkin ketterän kehittämisen arvot ja periaatteet, jotka on esitelty kappaleessa 2.1 Ketteryys ohjelmistokehityksessä.

Kuten SAFessa, myös kehittämismallissa on neljä eri tasoa. Tiimitasolla hyödynnetään pääasiassa Scrum-viitekehystä, kuten myös SAFessa. Organisaatiossa useat tiimit suunnittelevat ja synkronoivat tekemisiään muiden tiimien kanssa suunnittelujaksojen aikana. Suunnittelujakso koostuu neljästä peräkkäisestä sprintistä, eli yhteensä työtä suunnitellaan 12 viikkoa kestäväälle ajalle. Organisaation kehittämismallilla alueella ei ole erillistä nimeä, mutta käytännössä taso vastaa SAFen julkaisutasoa. Liiketoiminta-alueella kehittämistä tarkastellaan sen mukaan, mikä kehittämistoimenpide tuottaa arvoa ja yhdessä kehitysvastaava ja päätuoteomistaja sopivat siitä, minkä kokonaisuuden kehittämiseen keskitytään ja mitkä osa-alueet jätetään priorisoinnissa pois. Ylimmällä mallin tasolla tarkastellaan sitä, että toimitaan Kelan strategisten kehittämisvaatimusten mukaisesti. Käytännössä tämä vastaa SAFen portfoliotasoa. (Kehittämismalliryhmä, 2022)

Eroavaisuuksia Portfolio SAFe 5.1 -malliin on monia. Kelan kehittämismallista puuttuu esimerkiksi julkaisujuna ja julkaisujunaan yhdistetty jatkuva integraatio ja jatkuva käyttöönotto, DevOps, sisäänrakennettu laatu, portfolion visio ja Lean portfolion hallinta. Lisäksi Portfolio SAFe 5.1 -malliin verrattuna Kelan kehittämismallissa on yhden portfolion sijasta viisi eri portfoliota: Palvelutoimiala, Tieto, Yhteiset, IT ja johdon tuki -kehitysportfoliot. Alla olevassa kuvossa (KUVIO 6) on esiteltynä Kelan oma kehittämismalli, jossa on paljon yhtäläisyyksiä SAFe viitekehukseen.



KUVIO 6 Kelan kehittämismalli (Kehittämismalliryhmä, 2022)

2.6 SAFe MM

Scaled Agile Framework Maturity Model (SAFe MM) eli skaalautuvan ketterän viitekehysten kypsyysmalli perustuu Sidkyn ja Arthurin (2007) kypsyystasomallin tulkintaan. Kypsyysmallissa on viisi eri kypsyystasoa, joilla jokaisella tasolla on omat ketteryyden periaatteet. SAFe MM tasot, periaatteet ja käytännöt esitellään tarkemmin seuraavassa alaluvussa 2.6.1 SAFe MM tasot, periaatteet ja käytännöt.

2.6.1 Ketteryyden kypsyystasomalli

Ketteryyden kypsyystasomalli on työkalu, jonka avulla organisaatiot voivat arvioida joko ketterien menetelmien ja ketteryyden periaatteiden omaksumista (Leppänen, 2013). Useat ketteryyden kypsyystasomallit on perustettu Capability Maturity Model (CMM) tai Capability Maturity Model Integration) -kypsyysmallien pohjalta, mutta mallien taustalla näkyy osassa myös ketterän ohjelmistokehityksen ketterät arvot (Leppänen, 2013; Nurdiani, Börstler, Fricken, Petersen & Chatzipetrou, 2019). Näin on esimerkiksi Sidkyn ja Arthurin (2007) sekä Turetken ym. (2016) mallissa.

Ketteryyden kypsyystasomallien vertailussa Nurdiani ym. (2019) keskittyvät tutkimaan, missä järjestyksessä organisaation kannattaisi ottaa ketteryyden periaatteet käyttöönsä, jotta ketteryyden omaksuminen olisi mahdollisimman tehokasta. Tutkimuksessa vertailtiin yhteensä 12 eri kypsyystasomallia. Ketteryyden kypsyystasomallien vertailusta käy ilmi, että eri mallien välillä ei ole yhtenäistä ymmärrystä siitä, millä kypsyystasolla mikäkin ketteryyden periaate tulisi olla (Nurdianin ym., 2019). Lisäksi kaikkia ketteryyden periaatteita ei mainita myöskään kaikissa kypsyystasomalleissa. Tulosten perusteella Nurdiani ym. (2019) esittävät, että ketterien periaatteiden omaksumiseen ei ole yhtä oikeaa strategiaa tai mallia, joka sopisi paremmin kuin muut mallit.

Leppänen (2013) esittää, että kypsyystasomallin valinnassa olisi tärkeää painottaa mallin sopivuutta organisaatiokontekstiin. Körnerin ja Plöschin (2020) mukaa kypsyysmallin valinnassa tulisi kiinnittää huomiota seuraaviin ominaisuuksiin, jotta malli tuottaisi organisaatiolle käytännön arvoa. Ketteryyden kypsyysmallin tulisi:

- huomioida liiketoiminnan tarpeet,
- rakentua ketterien arvojen ja periaatteiden ympärille,
- huomioida pehmeät tekijät,
- olla ei viitekehyskohtainen,
- tarjota eri tasoille määritelmät,
- tarjota tukea työkaluun sekä
- olla joustava palvellakseen eri organisaatiokonteksteja (Körner & Plösch, 2020)

Kypsyysmallien vertailussa käy ilmi, että monet mallit jättävä huomiotta pehmeät arvot, joita ovat muun muassa yhteistyö, kommunikointi, oman toiminnan reflektointi ja jatkuva oppiminen. Ketterää kypsyysmallia valitessa pehmeiden tekijöiden huomioimen välttämätöntä, sillä pehmeät tekijät määrittelevät usein sen, että muutos onnistuu (Iveroth, 2010). Kohdeorganisaation kohdalla on siis tärkeää, että kypsyysmalli huomioi organisaation pehmeät arvot, ketterät arvot ja periaatteet sekä sopivuuden organisaatiokontekstiin. Koska kohdeorganisaation kehittämismalli perustuu pääosin ketterien arvojen ja periaatteiden sekä SAFen pohjalle, kypsyysmalleista Turetkenin ym. (2016) malli on sopiva.

2.6.2 SAFe MM tasot, periaatteet ja käytännöt

Scaled Agile Framework Maturity Model eli SAFe MM on Turetkenin, Stojanovin ja Trienekensin tekemä kypsyystasomalli. Kypsyystasomalli on kehitetty organisaatioiden työkaluksi tukemaan SAFen käyttöönottoa ja omaksumista. SAFe MM perustuu olemassa olevaan Sidkyn ja Arthurin (2007) ketteryyden kypsyysmalliin, jota on laajennettu SAFen periaatteilla (Turetken ym., 2016).

Kuviossa (KUVIO 7) on kuvattuna SAFe MM:n tasot, periaatteet ja käytännöt. Malli koostuu viidestä eri tasosta ja viidestä eri periaatteesta. Kypsyysmallissa ensimmäinen taso on matalin ja viides taso ylin taso. Mitä ylempää kypsyystasoa mennään, sitä vaikeampaa periaatteiden omaksuminen on. Kypsyysmallia luodessa periaatteet on sijoitettu eri tasoille sillä tavalla, että alemman tason periaatteiden omaksuminen tukee ylempään tason periaatteiden omaksumista, ja joissain tapauksissa ylempään tason periaatteita on mahdotonta omaksua, ellei alempaa tasoa ole omaksuttu riittävällä tasolla. Periaatteet rakentuvat siis alempien tasojen päälle. Ensimmäisellä kypsyysmallin tasolla korostetaan yhteistyötä, toisella evolutiivisuutta, kolmannella vaikuttavuutta, neljännellä adaptiivisuutta ja viidennellä eli ylimmällä tasolla kattavuutta. SAFe MM periaatteet ovat muutoksen omaksuminen asiakasarvon tuottamiseksi, ohjelmistojen suunnittelu ja toimittaminen usein, ihmiskeskeisyys, tekninen huippuosaaminen sekä asiakasyhteistyö. Nämä periaatteet on muodostettu 12 ketterän periaatteen pohjalta, jotka on kategorisoitu viiteen eri ryhmään. (Turetken ym., 2016). Tarkempi yleiskuva SAFe MM -mallista löytyy liitteistä (LIITE 2).

	Principles				
	Embrace change to deliver customer value	Plan and deliver software frequently	Human centricty	Technical excellence	Customer collaboration
Level 5 Encompassing	<ul style="list-style-type: none"> • L5P1: Low process ceremony • L5P2: Continuous SAFe Capability Improvement 	<ul style="list-style-type: none"> • L5P3: Agile project estimation 	<ul style="list-style-type: none"> • L5P4: Ideal Agile physical setup • L5P5: Changing organization 	<ul style="list-style-type: none"> • L5P6: Test driven development • L5P7: No/minimal number of level -1 or 1b people on team • L5P8: Concurrent testing 	<ul style="list-style-type: none"> • L5P9: Frequent face-to-face interaction between developers and users (collocated)
Level 4 Adaptive	<ul style="list-style-type: none"> • L4P1: Client driven iterations • L4P2: Continuous customer satisfaction • L4P3: Lean requirements at scale 	<ul style="list-style-type: none"> • L4P4: Smaller and more frequent releases • L4P5: Adaptive planning • L4P6: Measuring business performance 	<ul style="list-style-type: none"> • L4P7: Managing highly distributed teams 	<ul style="list-style-type: none"> • L4P8: Intentional architecture • L4P9: Daily progress tracking meetings 	<ul style="list-style-type: none"> • L4P10: CRACK customer immediately accessible • L4P11: Customer contract revolves around commitment of collaboration
Level 3 Effective	<ul style="list-style-type: none"> • L3P1: Regular reflection and adaptation 	<ul style="list-style-type: none"> • L3P2: Risk driven iterations • L3P3: Plan features not tasks • L2P4: Roadmap • L3P5: Mastering the iteration • L3P6: Software Kanban Systems • L3P7: PSI/Release • L3P8: Agile Release Train 	<ul style="list-style-type: none"> • L3P9: Self-organizing teams • L3P10: Frequent face to face communication • L3P11: Scrum of Scrum 	<ul style="list-style-type: none"> • L3P12: Continuous integration • L3P13: Continuous improvement (refactoring) • L3P14: Unit tests • L3L15: 30% of level 2 and level 3 people 	<ul style="list-style-type: none"> • L3P16: DevOps (Integrated Development and Operations) • L3P17: Vision, features • L3P18: Impact on customers and operations
Level 2 Evolutionary	<ul style="list-style-type: none"> • L2P1: Evolutionary requirements • L2P2: Smaller, more frequent releases • L2P3: Requirements discovery 	<ul style="list-style-type: none"> • L2P4: Continuous delivery • L2P5: Two-level planning and tracking • L2P6: Agile Estimating and Velocity • L2P7: Release planning 	<ul style="list-style-type: none"> • L2P8: Define/Build /Test team 	<ul style="list-style-type: none"> • L2P9: Software configuration management • L2P10: Automated testing • L2P11: Tracking iteration progress • L2P12: No big design up front (BDUF) • L2P13: Product Backlog • L1P5: Coding standards • L1P6: Knowledge sharing • L1P7: Task volunteering • L1P8: Acceptance testing 	<ul style="list-style-type: none"> • L2P14: Customer contract reflective of evolutionary development
Level 1 Collaborative	<ul style="list-style-type: none"> • L1P1: Reflect and tune process 	<ul style="list-style-type: none"> • L1P2: Collaborative planning 	<ul style="list-style-type: none"> • L1P3: Empowered and motivated teams • L1P4: Collaborative teams 		<ul style="list-style-type: none"> • L1P9: User stories • L1P10: Customer commitment to work with development team

KUVIO 7 SAFe Maturity Model: tasot, periaatteet ja käytännöt (Turetken ym., 2016)

Taulukossa (TAULUKKO 4) on kuvattuna eri kypsyystasojen käytännöt ja käytännön kuvaukset.

TAULUKKO 4 Kypsyystason käytänteiden kuvaukset (Turetken ym., 2016)

Kypsyystaso	Käytäntö	Käytännön kuvaus
Taso 1 Yhteistyössä tehty	Prosessin reflektointi ja parantaminen	Retrospektiivien pitäminen säännöllisesti. Käytännön tavoitteena on parantaa prosessia.

	Yhteissuunnittelu	Yhteissuunnitteluun osallistuvat kaikki tarvittavat sidosryhmät. Yhteissuunnittelu lisää näkyvyyttä, luotettavuutta ja hyväksyntää.
	Voimaantuneet ja motivoituneet tiimit	Esihenkilöt mahdollistavat sen, että tiimeillä on tarvittavat valtuudet ja edellytykset tehdä päätöksiä itse. Auktoriteettiasemassa olevat motivoivat tiimin jäseniä ratkaisemaan ongelmat ja tehtävät itse.
	Yhteistyötä tekevät tiimit	Tiimin jäsenet kommunikoivat ja tekevät yhteistyötä keskenään ja muiden tiimien kanssa.
	Koodausstandardit	Kehittäjillä on oma yhteinen kieli ja käytännöt koodin kirjoittamisen suhteen. Koodausstandardi helpottaa koodin jakamista tiimin jäsenten kesken.
	Tiedon jakaminen	Käytössä on tiedonjakamiseen liittyviä työkaluja, jotka auttavat dokumentoimaan ja ylläpitämään informaatiota ja jakamaan tietämystä tiimin jäsenten välillä.
	Vapaaehtoisuus tehtävissä	Tiimin jäsenet osallistuvat tehtävien tekemiseen vapaaehtoisesti, eivätkä esihenkilöt päättää, kuka tekee minkäkin tehtävän. Jos vapaaehtoisia tehtävän suorittamiseen ei ole, tiimi vastaa kollektiivisesti tehtävän suorittamisesta.
	Käyttäjätarinat	Ketterä käytäntö, jossa ohjelmistovaatimukset muodostetaan parilla lauseella käyttäjän kielellä.
	Hyväksymistestaus	Hyväksymistestauksessa varmistetaan, että järjestelmä toimii käyttäjätarinan mukaisesti. Testit ovat automatisoidut aina kuin mahdollista.
	Asiakas sitoutuu työskentelemään kehitystiimin kanssa	Asiakkaan on sitouduttava toimimaan kehitystiimin kanssa, jotta yhteistyö kehitystiimin kanssa olisi sujuvaa.
Taso 2	Evolutiivinen	
	Vaatimusten evolutiivisuus	Vaatimuksia päivitetään iteratiivisesti, eikä määrittelyä tehdä kerralla hyvin tarkkaan loppuun saakka. Määrittely kehittyy ja muuttuu asiakaspalautteen myötä.
	Pienemmät, useammat julkaisut	Tiimin tulisi tuottaa pieniä ja useita julkaisuja asiakkaille ja loppukäyttäjille. Tämä mahdollistaa paremman reagoitakyvyn ja vähentää organisaatioon kohdistuneita riskejä. Tarkempaa tietoa siitä, kuinka usein organisaation tulisi julkaista, ei ole määritelty.
	Vaatimusten löytäminen	Ketterä harjoitus, jonka tarkoituksena on auttaa ymmärtää paremmin, mitä pitää rakentaa ja miksi. Erilaisia ohjelmistovaatimustekniikoita esim. Spike, käyttötapaus, käyttäjäkokemuksen mallit jne. tiimien kannatta hyödyntää.
	Jatkuva tuottaminen	Ohjelmiston tuottaminen pienissä iteraatioissa säännöllisellä rytmillä. Tämä käytäntö varmistaa sen, että tiimin on jaettava tuotteen kehitys pienempiin iteraatioihin.

Kahden tason ja suunnittelu ja seuraaminen	Työtä tulisi suunnitella lyhyemmällä sekä pidemmällä aikavälillä ja eri tarkkuustasoilla. Lyhyen aikavälin suunnittelulla tarkoitetaan iteraatioiden suunnittelua ja pidemmällä välin suunnittelulla tarkoitetaan julkaisusuunnittelua, joka pysyttelee karkeammalla ja vähemmän tarkalla tasolla.
Ketteryyden arviointi ja kiertonopeus	Työmääriä arvioidaan tarinapisteiden (engl. <i>story points</i>) avulla. Tiimi tuottaa tietyn määrän tarinapisteitä iteraatioissa, jotka määrittävät tiimin nopeuden. Nopeusarvioita hyödynnetään aikataulun ja kustannusten arvioimisessa.
Julkaisun suunnittelu	Useat tiimit synkronoivat tekemisensä julkaisun suunnittelussa. Tilaisuus kestää kerrallaan kaksi päivää.
Määrittely/ohjelmointi/testaus-tiimi	Timissä on kaikki tarvittava osaaminen määrittelystä, ohjelmointiin ja testaamiseen. Moniosaava tiimi pystyy tuottamaan toimivan osan lyhyessä rajatussa aikavälissä.
Ohjelmistokokoonpanon hallinta	SCM eli software configuration management -työkalujen käyttäminen kehitettävän ohjelmiston eri versioiden hallinnassa.
Automaatiotestaus	
Iteraation edistymisen seuranta	Tiimillä on mahdollisuus mitata kehitystyön edistymistä jokaisen iteraation aikana.
Ei isoa suunnittelua etukäteen	Tuotetta kehitetään evolutiivisella tavalla eli kehitystä tapahtuu koko kehitysprosessin ajan. Vesiputousmallisessa kehitysprosessissa ensin suunnitellaan iso määrä tekemistä, jonka jälkeen vasta ohjelmoidaan ja testataan. Tätä kuvataan usein termillä Big Design Up Front eli BDUF. Ketterässä Ketterässä suunnittelussa suunnittelu, ohjelmointi ja testaus kulkee koko kehitysprosessin ajan.
Tuotteen kehitysjono	Tuotteen kehitysjonon on tulevien töiden varasto. Kehitysjonoa voidaan hyödyntää eri tarkkuustasoilla, mm. tiimi, ohjelma ja portfoliotasoilla.
Asiakassopimus heijastaa evolutiivista kehitystä	Asiakas ymmärtää evolutiivisen ohjelmistokehityksen luonteen, jonka seurauksena asiakassopimuksessa vältetään määrittelemästä päivämäärät, jolloin virstanylväät on saavutettava.
Taso 3 Tehokas	Säännöllinen reflektointi ja sopeutuminen Tiimit refleктоivat omaa toimintaa mittareiden avulla sekä sopeutuvat muutoksiin kullakin iteraatio- ja julkaisutasolla. Reflektointiin liittyy iteraatioiden ja julkaisun arviointi kvantitatiivisia arviointimenetelmiä käyttäen.
Riskilähtöiset iteraatiot	Riskien vähentäminen varhaisessa vaiheessa varmistaa sen, että tiimi ei käytä liikaa aikaa sellaisen järjestelmän rakentamiseen, jonka

	suorittamista he eivät voi suorittaa loppuun saakka.
Ominaisuuksien, eikä tehtävien suunnittelu	Asiakkaan tarpeet ilmaistaan ominaisuuksien ei tehtävien muodossa. Jos asiakas haluaa tehdä muutoksia, kehitystiimin on helpompi ymmärtää, mitä asiakas haluaa ja siten muutoksen vaikutus tehtäviin minimoituu.
Tiekartta (engl. <i>roadmap</i>)	Käytäntö, jonka avulla julkaisujunan tiimien välille pysytään sovittamaan yhteen. Tiekartta luo myös ennustettavuutta tuotoksille, millä aikavälillä tuotokset saadaan tuotettua.
Iteraation hallitseminen	Iteraatiossa tarkoituksena on pystyä luomaan toimivia, testattuja ohjelmistoja lyhyessä aikataulussa.
Kanbanin hyödyntäminen	SAFe ehdottaa, että Kanbanin hyödynnettäisiin myös liiketoiminta ja arkkitehtuuriportfolioiden Epiceissä. Kanbania käytetään työnkulun visualisointiin, käynnissä olevan työn rajoittamiseen, virtauksen mittaamiseen ja hallintaan.
PSI (engl. <i>potentially shippable increment</i>)/julkaisu	PSI:n tai julkaisun tavoitteena on jatkuva integrointi sekä viivästyneestä integraatiosta johtuvan riskin vähentäminen.
Julkaisujuna (engl. <i>Agile Release Train</i>)	Julkaisujuna koostuu useasta eri ketterästä tiimistä, jotka organisoituvat yhden arvovirran ympärille. Julkaisujunalla varmistetaan se, että tiimit tekevät töitä saman mission, aikataulun ja tahdin mukaan.
Itseorganisoituneet tiimit	Tiimit ovat valtuutettuja tekemään päätöksiä itsenäisesti odottamatta johdon hyväksyntää. Tiimit ovat poikkitoiminnallisia, roolit ja vastuut tiimin jäsenten kesken ei ole ollenkaan tai on löyhästi määriteltyjä. Koko tiimi on yhdessä vastuussa toimivan ohjelmiston tuottamisesta.
Säännöllinen henkilökohtainen vuorovaikutus	Tehokkaan ja toimivan kehitysprosessin luomiseksi tiimin jäsenten välinen säännöllinen kommunikointi on välttämätöntä.
Scrumien Scrum	Scrumien scrum eli scrumin sisällä pidettävä scrum.
Jatkuva integraatio	Tiimin jäsenten työt integroidaan mahdollisimman usein. Jokainen integraatio olisi hyvä varmentaa automatisoidulla rakennustyökalulla, jotta mahdolliset integroimisvirheet huomataan mahdollisimman nopeasti.
Jatkuva parantaminen (refaktorointi)	Refaktoroinnilla tarkoitetaan koodin uudelleenkirjoittamista siten, että sisällöllisesti koodi pysyy samana, mutta rakenteellisesti koodin rakennetta parannetaan ja on sitä kautta jatkossa ylläpidettävämpi. Yleensä refaktorointi keskittyy koodin päällekkäisyyden poistamiseen.
Yksikkötestit	Yksikkötesteillä varmistetaan, että lähdekoodin yksittäiset osat toimivat oikein. Suositeltavaa on, että yksikkötestit on automatisoitu.

	30% tason 2 ja tason 3 ihmisistä	Cockburnin mukaan henkilön kokemuksen määrä alalla liittyy henkilön ymmärtämistasoon. Ymmärtämistasoja on Cocburnin mukaan kolme. Tasolla kolme tarvitaan tason 2 ja tason 3 osajia.
	DevOps	Ketterä käytäntö, jolla varmistetaan nopeampi arvon kulku käyttäjälle integroimalla tiiviimmäksi ohjelmistokehitys sekä ylläpitoon liittyvät toiminnot.
	Visio, ominaisuudet	Visio on sidosryhmien näkemys kehitettävästä ratkaisusta sidosryhmien tarpeiden ja ehdotettujen ominaisuuksien perusteella. Siinä huomioidaan suunnitellun ratkaisun korkean tason ominaisuuksien, ei-toiminnallisten vaatimusten ja suunnittelusta johtuvien rajoitteiden muodossa ja annetaan yleiskatsaus kehitettävästä järjestelmästä.
	Vaikutus asiakkaisiin ja toimintaan	Kehitysmallin muutosten vaikutuksia myyntiin, toimintaan ja asiakkaaseen tulisi mitata.
Taso 4 Adaptiivinen	Asiakaslähtöiset iteraatiot, L4P1	Asiakas voi pyytää ja priorisoida ominaisuuksia eri iteraatioissa.
	Jatkuva asiakastyytyväisyys palaute, L4P2	Jatkuvalla asiakaspalautteella varmistetaan asiakastyytyväisyys.
	Leanin vaatimukset skaalautuvassa mittakaavassa, L4P3	Isoissa organisaatioissa on vaikeaa saada koko tiimi työskentelemään kohti yhteistä tavoitetta. Tätä varten on luotava skaalautuva vaatimusmalli, joka koostuu visiosta, tiekartasta ja oikea-aikaisesta työstämisestä, joka varmistaa ketteryyden hyödyt laajemmille skaalautuville tiimeille.
	Pienemmät, useammat julkaisut, L4P4	Pienemmät, mutta useammat julkaisut auttavat tekemään nopeammin muutoksia. Julkaisu tulisi tapahtua noin 8 viikon välein.
	Adaptiivinen suunnittelu, L4P5	Iteraation yksityiskohdat tarkennetaan vasta seuraavaa iteraatiota edeltävänä ajanjaksona, jotta kaikki edellisen iteraation aikana opitut asiat ja palaute saadaan sisällytettyä iteraatioon.
	Liiketoiminnan tehokkuuden arvioiminen, L4P6	Käytössä tulisi olla joustava, automatisoitu ja merkityksellinen BSC eli Balanced Scorecard, joka mittaa suorituskykyä tehokkuuden, arvon toimittamisen, laadun ja ketteryyden perusteella.
	Hajautettujen tiimien hallinta, L4P7	Hajautettujen tiimien hallinta tulisi hoitaa asianmukaisella viestinnällä sekä tarvittavalla verkko- ja työkalurakenteilla.
	Tarkoituksenmukainen arkkitehtuuri, L4P8	Arkkitehtuuri on joukko tarkoituksenmukaisia, suunniteltuja arkkitehtonisia aloitteita ratkaisujen suunnittelun, suorituskyvyn ja käytettävyyden parantamiseksi, ja se tarjoaa ohjeita ryhmien väliseen suunnitteluun ja toteutuksen synkronointiin.
	Päivittäiset edistymisen	Tiimissä iteraation edistymistä seurataan päivittäin.

	seurantakokoukset, L4P9	
	Asiakas on heti käytettävissä, L4P10	Asiakas on yhteistyöhön osallistuva, edustaja, valtuutettu, sitoutunut ja asiantunteva.
	Asiakasyhteys pyörii yhteistyön sitoutumisen ympärillä, L4P11	Asiakas suostuu sopimaan yhteistyön määrästä eikä vaatimusten ja ominaisuuksien määrästä.
Taso 5 Kattava	Matala prosessiseremonia, L5P1	Prosesseissa usean eri johtotason ei tarvitse hyväksyä muutosta.
	Jatkuva SAFe kyvykkyyksien parantaminen	Tätä käytäntöä ei ole määritelty Turetkenin ym. lisämateriaalissa.
	Ketterien projektien arviointi, L5P3	Projektien ketterä arviointia tulisi tehdä, sillä onnistunut lopputulos on kiinni siitä, kuinka hyvä arvio suunnitelmasta on. SAFe käyttää useita arviointitekniikoita, kuten viivästyskustannusta (engl. CoD), WSJF ja story pointteja arvioinnissa.
	Ihanteellinen fyysinen ympäristö ketterälle kehittämiselle, L5P4	Ihanteellinen ketterä fyysinen kokoonpano auttaa luomaan oikean ympäristön ketterän ohjelmistokehityksen menestymiseen. Tämän järjestelyn tarkoituksena on se, että tiimi on yhdessä paikkaa ja tietoa voidaan jakaa tiimin jäsenten kesken heti.
	Organisaation muuttaminen, L5P5	Organisaatioiden on tehtävä olennaisia muutoksia saavuttaakseen ketterän hyödyn yrityksen tasolla. Muutokset voivat tarkoittaa muutoksia suunnittelukäytäntöihin, operaatioihin kohdistuvien vaikutusten hallintaan, uudelleenorganisointumista jne. Organisaatiomuutoksissa johdon tuki ja sitoutuminen prosessiin sekä esimerkiksi johtaminen ovat avaintekijöinä hyötyjen saavuttamisessa.
	Testilähtöinen kehittäminen, L5P6	Ohjelmistokehitystekniikka, jossa testit kirjoitetaan ensin, jonka jälkeen vasta tarvittava koodi kirjoitetaan testien läpäisemiseksi.
	Tason -1 tai 1b ihmisiä vähän tai ei ollenkaan tiimissä, L5P7	Cockburnin mukaan henkilön kokemuksen määrä alalla liittyy henkilön ymmärtämistasoon. Kehittäjiä, jotka ovat tasoilla 1B ja -1, tulisi olla hyvin vähän.
	Rinnakkainen testaaminen, L5P8	Käytäntö sisällyttää kehitystestaus-, testausautomaatio- ja testauksen suorituskäytännöt iteraation aikana. Samanaikainen testaus sisältää useita ketterän testauksen strategioita, kuten: yksikkötestaus, hyväksyntätestaus, komponenttitestaus, järjestelmä-, suorituskyky- ja luotettavuustesti.
	Kehittäjien ja käyttäjien välinen vuorovaikutus, L5P9	Ihanteellista olisi, että ei vaan kehittäjät, vaan myös asiakkaat ja käyttäjät tekisivät yhteistyötä fyysisesti samoissa tiloissa.

2.6.3 Cockburnin henkilöstötasot

Cockburnin mukaan henkilön kokemuksen määrä alalla liittyy henkilön ymmärtämisen tasoon. Ymmärtämistasoja on Cockburnin mukaan kolme: seuraaminen, irrottautuminen ja sujuva. Uuden taidon hallitsemisen vaatii sen, että henkilö käy nämä kolme käyttäytymisvaihetta läpi. Seuraamisvaiheessa eli tasolla 1 henkilö etsii yhtä menettelytapaa, joka toimii joka tilanteessa. Irrottautumisvaiheessa eli tasolla 2 henkilö tunnistaa yhden menettelytavan rajoitukset ja oppii mukauttamaan menettelytavan vaihtelevissa olosuhteisissa. Suvuussa vaiheessa eli tasolla 3 henkilö on oppinut, miten halutun lopputuloksen saa aikaan, jolloin menettelytapojen noudattaminen ei ole relevanttia. (Cockburn, 2002)

Boehm ja Turner kehittivät Cockburnin ymmärtämistason pohjalta mittarin, jolla voi arvioida ohjelmoijan ymmärrystä ohjelmoinnista ja sitä kautta mitata ohjelmoijan pätevyyttä sekä sopivuutta eri tilanteisiin. Jotta nämä kaikki asiat pystyttäisiin arvioimaan mittarin avulla, Boehm ja Turner jakoivat Cockburnin tason 1 kolmeen alaosiioon. Tason -1 henkilöt kannattaisi siirtää työskentelemään muuhun kuin ketteriä menetelmiä tai kurinalaisia menetelmiä noudattavaan tiimiin. Tason 1B henkilöt toimivat hyvin vakaassa, ennalta-arvattavassa ympäristössä, mutta ketterissä menetelmissä kokemattomuus toimii hidasteena, varsinkin jos suurin osa tiimistä kuuluu 1B tason henkilöihin. Tason 1A henkilöt pystyvät toimimaan hyvin sekä vakaissa että nopeasti muuttuvissa tilanteissa, jos tiimissä on tason 2 henkilöitä opastamassa heitä. Tason 2 ja 3 henkilöt toimivat hyvin vakaissa ja nopeasti muuttuvissa ympäristöissä sekä heillä on kyky muokata metodeja sopimaan uusiin tilanteisiin (Boehm & Turner, 2003). Tason 3 henkilöt eroavat tason 2 henkilöistä siinä, että he ovat aidosti oppineet hallitsemaan ohjelmoinnintaidon, jonka vuoksi he osaavat uudistaa menetelmiä täysin uusiin tilanteisiin. (Boehm & Turner, 2003; Cockburn, 2002). Taulukossa (TAULUKKO 5) on koottuna yhteen ohjelmoijan osaamistasot.

TAULUKKO 5 Ohjelmoijan osaamistasot (Cockburn, 2002; Boehm & Turner, 2003)

Taso	Kriteeri
Taso -1	Kykenemätön tai haluton tekemään yhteistyötä tai noudattaa jaettuja menetelmiä
Taso 1B	Keskinkertainen, vähemmän kokenut, mutta ahkera ohjelmoija. Harjoittelun avulla pystyy suorittamaan vakiintuneiden menetelmien vaiheita. (Esim. yksinkertaisen metodin ohjelmointi, yksinkertainen refaktorointi, koodausstandardien ja konfiguraatioiden hallintamenettelyjen noudattaminen, testien suorittaminen). Kokemuksen kautta pystyy hallitsemaan joitain 1A tason taitoja.
Taso 1A	Harjoittelun avulla pystyy suorittamaan harkinnanvaraisesti menetelmien vaiheita (esim. käyttäjätarinoiden sovittaminen inkrementteihin, refaktorointimenetelmien yhdisteleminen, kompleksiset kuluttajatuoteintegraatiot (engl. COTS). Kokemuksen kautta pystyy siirtymään tasolle 2.

Taso 2	Kykenee räätälöimään menetelmät sopimaan ennakkotapauksen perusteella uuteen tilanteeseen. Onnistuu hyvin pienen tiimin johtamisessa, mutta tarvitsee tason 3 ohjausta isoissa ja ennen kokemattomissa projekteissa.
Taso 3	Kykenee muokkaamaan menetelmän (sääntöjen rikkominen) sopimaan ennen kokemattomaan uuteen tilanteeseen. Onnistuu hyvin pienten ja isojen tiimien johtamisessa.

2.6.4 SAFe MM mallin käyttäminen

Kypsyystason arviointi tapahtuu haastattelemalla organisaation RTE:tä (engl. *Release Train Engineer*) ja scrummastereita. Haastattelemalla eri rooleja saadaan kattavampi kuva, millä tasolla organisaatio on SAFe mallin alemmilla ja toisaalta ylemmillä tasoilla. Haastattelussa tarkastellaan, mitkä kypsyyden periaatteet täyttyvät ja minkä asteisesti (TAULUKKO 4). ISO / IEC 15504 -arviointistandardi 52 mukaisesti periaatteiden täyttymistä arvioidaan seuraavanlaisesti:

- N tarkoittaa "Not achieved", eli tavoitetta ei ole saavutettu, näyttöä käytännön saavuttamisesta ei ole ollenkaan tai hyvin vähän.
- P tarkoittaa "Partially achieved", eli tavoite on osittain saavutettu. Todisteita käytännön saavutuksista on jonkin verran.
- L tarkoittaa "Largely achieved", eli laajasti saavutettu. Näyttöä systemaattisesta lähestymistavasta ja käytännön merkittävästä saavutuksesta on näyttöä, joitakin heikkouksia voi vielä kuitenkin olla.
- F tarkoittaa "Fully achieved", eli täysin saavutettu. tarkoittaa vahvaa näyttöä täydellisestä ja järjestelmällisestä lähestymistavasta ja käytännön täydellisestä saavuttamisesta ilman merkittäviä heikkouksia. (Turetken ym., 2016)

Haastattelun tulosten perusteella arvioidaan, millä ketteryiden kypsyystasolla organisaatio on. Analysoimalla tarkemmin tuloksia selviää se, kuinka laajasti tutkittava organisaation on saanut saavutettua esimerkiksi 1 tason periaatteet. Jos organisaatio on onnistunut omaksumaan laajasti tai täysin kaikki tason 1 periaatteet, organisaation kypsyys on vähintään tasolla 1. Arviointia suoritetaan eteenpäin niin kauan, kunnes jokainen ketteryiden käytäntö ja kypsyystaso on käyty läpi. Kypsyystason arvioinnissa tulee huomioida se, että vaikka organisaatio olisi saavuttanut joitakin periaatteita tasolta 3, se ei tarkoita sitä, että organisaatio olisi välttämättä ketterä. Taulukon periaatteet tukevat organisaation ketteröitymistä, mutta periaatteiden omaksuminen ei kuitenkaan tarkoita vielä automaattisesti sitä, että organisaatio olisi ketterä. (Turetken ym., 2016) Organisaation ketteryiden kypsyystason arvioinnissa on tarkoitus hyödyntää haastattelun tuloksia ja niiden analysointia, mutta selvää vastausta siitä, millä tasolla organisaatio on ketteryiden suhteen kypsyystasomalli ei anna. SAFe MM kypsyystasomallin käyttäjän vastuulle jää se, miten tuloksia analysoi ja mille tasolle tutkittava organisaatio sijoitetaan.

2.6.5 Millä tapaa SAFe MM mallia on sovellettu aikaisemmassa kirjallisuudessa?

Google Scholar haun perusteella viittauksia Turetkenin ym. (2016) SAFen kypsyysmalliin löytyy yhteensä 53 kappaletta. Viittaukset ovat löytyneet 1.6.2021 ja tämän jälkeen viittauksia on tullut enemmän, mutta niitä ei ole huomioitu tämän tutkielman puitteissa. Näistä viittauksista kahteen artikkeliin oli virheellinen viittaus Google Scholarin perusteella: viittauksia ei löytynyt tekstistä tai lähdeluettelosta. Yhdessä tutkimuksessa viittaus löytyy lähdeluettelosta, muttei tekstistä. Tietokantahausta tuli huomioida myös se, että viiden eri aineiston tietoja ei pystytty tarkistamaan, sillä pääsyä aineistoon ei ollut. Tieteellisiksi lähteiksi luetaan dokumenttityypit, jotka ovat konferenssijulkaisuja, tieteellisiä julkaisuja tai kirjoja (Garousi, Felderer, Mäntylä, 2019). Opinnäytetöitä, kuten pro gradu -tutkielmia tai väitöskirjoja ei ole hyväksytty tieteelliseksi julkaisuksi.

Artikkelit, joissa SAFen kypsyysmalliin on viitattu, löytyvät liitteestä (LIITE 1). Muulla kuin englannin tai suomen kielellä kirjoitetut aineistot jätettiin arvioimatta ja lukematta tutkiessa tapaa, miten SAFe MM mallia on sovellettu aiemman kirjallisuuden perusteella. Löydetyissä artikkeleissa Turetkenin ym. (2016) malliin on eniten viitattu julkaisujen toisessa kappaleessa, jossa luodaan pohjaa tutkimuksille aiemman kirjallisuuden avulla. Viidessä eri viittauksessa Turetkenin ym. (2016) mallia on arvioitu ketterien kypsyysmallien vertailussa ja yhdessä tutkimuksessa eri olemassa olevista kypsyysmalleista on tehty yhteenveto, mutta malleja ei ole vertailtu keskenään.

Löydösten perusteella SAFe MM -mallia ei ole aiempien viittausten perusteella testattu empiirisesti. Körnerin ja Plöschin (2020) arvion mukaan nykyiset ketteryuden kypsyysmallit eivät palvele toimialan oikeita tarpeita, sillä kypsyysmallit eivät ole vielä riittävän kypsiä tuottaakseen oikeaa hyötyä toimialoille. Myös Leppäsen (2013) ketteryuden kypsyysmallien vertailussa nousee esille se, että kypsyysmalleja on vielä kehitettävä sekä teoreettisen että empiirisen tutkimuksen pohjalta.

3 TUTKIMUSMENETELMÄ

Tämän luvun tavoitteena on luoda käsitys tutkielman empiirisen osuuden prosessista, tarkoituksesta sekä valitusta tutkimusmenetelmästä. Luvussa esitellään valittu tiedonkeruumenetelmä, kohdeorganisaatio sekä haastateltavien valintaan vaikuttavat tekijät ja kuinka haastattelut toteutettiin. Viimeisenä luvussa kerrotaan aineiston analysointitavasta.

3.1 Tutkimuksen tavoite ja tutkimusmenetelmä

Tässä tutkielmassa on käytetty laadullista tutkimusmenetelmää. Laadullinen tutkimusmenetelmä valittiin menetelmäksi, sillä se soveltuu hyvin sosiaalisten ja kulttuuristen ilmiöiden tutkimiseen ja tietyn toiminnan ymmärtämiseen (Myers, 1997; Tuomi & Sarajärvi, 2009). Tämä tutkielma pyrkii arvioimaan Kelan ketteryuden omaksumista ketterien arvoihin ja periaatteisiin peilaten sekä vastaamaan tutkimuskysymykseen:

- Kuinka hyvin kohdeorganisaatio on onnistunut omaksumaan ketterän menetelmien arvot ja periaatteet Turetkenin ym. (2016) kypsyystasomallin perusteella?

Lähtökohta tutkielman aiheelle oli tunnistaa nimenomaan julkishallinnollisen organisaation kypsyystasoa ketteryuden suhteen. Tutkimuksen laajempi tavoite oli kasvattaa ymmärrystä ensinnäkin siitä, miten ketterät menetelmät soveltuvat isoille julkishallinnollisille organisaatioille ja toisaalta kuinka hyvin nykyiset ketteryuden kypsyystasoa mittaavat viitekehukset soveltuvat julkishallinnollisen organisaation kypsyystason mittaamiseen.

Tutkimusstrategian valintaa ohjasi kaksi kysymystä: soveltuuko tutkimusstrategia organisaation ja ketteryys -ilmiön syvälliseen tutkimiseen ja tukeeko tutkimusstrategia nimenomaan sosiaalisten ja kulttuuristen ilmiöiden tutkimista? Lisäksi olennaista oli huomioida se, että tutkielmassa tutkitaan vain

yhtä organisaatiota. Näiden kysymysten perusteella tutkimusstrategiaksi valittiin tapaustutkimus.

Tapaustutkimus tutkii tässä hetkessä tapahtuvaa ilmiötä tosielämän kontekstissa ja se soveltuu erittäin hyvin tilanteisiin, missä raja tutkittavan ilmiön ja kontekstin välillä ei ole selkeä (Yin, 2003). Tutkittava asia, ketteryuden omaksuminen julkishallinnollisessa organisaatiossa, vaatii erityisesti sosiaalisten ja kulttuuristen ilmiöiden ymmärtämistä nykyhetkessä, joten tapaustutkimuksen todettiin sopivan hyvin tutkimusstrategiaksi.

Hirsjärvi ja Hurme (2015) toteavat, että tutkija on aina osa sosiaalista todellisuutta ja siten osa tutkimusprosessia, joten tutkijan, eli tässä kohtaa tutkielman kirjoittajan, tulisi tunnistaa ja tunnustaa, että hänen läsnäolonsa on vaikuttanut esimerkiksi haastateltavan vastauksiin ja tutkielman tuloksiin.

3.2 Kohdeorganisaation esittely

Tutkittava kohdeorganisaatio on Kansaneläkelaitos eli Kela. Kela vastaa Suomessa asuvien ja ulkomailla asuvien suomalaisten sosiaaliturvasta eri elämäntilanteissa (Kela, 2022a). Organisaatio ei siis keskity pelkästään ohjelmistokehitykseen, mutta toimivat tietojärjestelmät ovat keskeinen osa liiketoimintaa. Kohdeorganisaatioksi valikoitu Kela, sillä tutkielman kirjoittaja on työsuhteessa kyseissä organisaatiossa ja omasta tahdostaan hän toivoi ymmärtävänsä paremmin organisaation ketteryuden omaksumisen tasoa toimiessaan Agile Coachina ja Scrum Masterina.

Kela koostuu viidestä eri tulosityksiköstä: Asiakkaan lähipalvelujen, Valtakunnallisten asiakkuuspalvelujen, IT-palveluiden, Tietopalveluiden sekä Yhteisten palveluiden tulosityksiköstä (Kela, 2022b). Kelassa vallitsee sisäinen tilaaja-toimittajamalli, jossa Kelan liiketoiminnot toimivat IT-ratkaisujen tilaajana ja IT-palveluiden tulosityksikkö vastaa IT-ratkaisujen toimittamisesta (Kehittämismalliryhmä, 2022). IT-palveluiden tulosityksikössä vastataan IT:n kehittämisestä, ylläpidosta sekä tuotantopalveluiden tuottamisesta ja tulosityksikkö jakaantuu tarkemmin kymmeneen eri yksikköön (Kela, 2022b; Kela, 2022c). Yhteensä työntekijöitä Kelan IT-palveluissa työskentelee yli 800 henkilöä (Kela, 2022d).

3.3 Tiedonkeruumenetelmä

Empiirinen aineisto kerättiin yksilöhaastattelujen avulla. Haastatteluissa käytettiin puolistrukturoituja haastatteluja, tarkemmin sanottuna teemahaastatteluja. Teemahaastattelun teemoina hyödynnettiin Turetkenin ym. (2016) kypsyystasomallista löytyviä viittä eri periaatetta: asiakasarvo, suunnittelu ja jatkuva tuottaminen, ihmiskeskeisyys, tekninen huippuosaaminen sekä asiakasyhteistyö.

Tiedonkeruumenetelmäksi teemahaastattelu valittiin tässä tutkielmassa kolmen asian perusteella. Ensinnäkin Hirsjärven ja Hurmeen mukaan (2015) teemahaastattelu sopii hyvin sellaisiin tutkittaviin aiheisiin, joista on vielä vähän tietoa. Toiseksi teemahaastattelu nähtiin hyvänä menetelmänä kerätä informaatiota laajemmin organisaation ketteryuden tilasta, sillä teemahaastattelussa vastausten taustalla vaikuttavat motiivit voivat tulla paremmin esille ja vastauksia on mahdollista selventää erilaisin lisäkysymyksin. (Hirsjärvi & Hurme, 2015) Kolmanneksi tutkimuksen onnistumisen kannalta nähtiin erittäin kriittisenä se, että haastattelija ymmärtäisi ja tulkitseisi haastateltavien sanoman sisällön mahdollisimman samalla tapaa kuin haastateltava oli sen tarkoittanut. Teemahaastattelussa myös haastateltava voi toimia haastattelijan lisäksi tarkentajana, joten tämä nähtiin tiedonkeruumenetelmän valinnassa oleellisena asiana (Hirsjärvi & Hurme, 2015).

Puolistrukturoimattoman haastattelun sijaan tutkielmassa käytettiin nimenomaan teemahaastattelua sen vuoksi, että teemahaastattelussa kysymysten tai kysymysten muodon ei tarvitse olla kaikille haastateltaville sama ja haastattelutyypin voi vaihdella lähes avoimesta aina lähes strukturoituun haastattelutyypin (Hirsjärvi & Hurme, 2015; Myers & Newman, 2006). Olennaisempaa onkin se, että haastattelussa käydään eri aihepiirit, eli teemat läpi eri haastateltavien kanssa. (Hirsjärvi & Hurme, 2015) Tässä tutkielmassa haastattelut vaihtelivat melko avoimesta haastattelutyypistä aina enemmän puolistrukturoituun haastattelutyypin. Haastattelutyypin käyttäminen riippui siitä, kuinka avoimesti haastattelija kykeni kertomaan havainnoistaan eri teemoihin liittyen: osa haastateltavista halusi tuoda omia havaintojaan itsenäisesti esille, toiset tarvitsivat teemoihin liittyviä tarkempia apukysymyksiä. Myers ja Newman (2006) ehdotuksen mukaisesti haastattelijan tavoitteena oli antaa haastateltavien puhua vapaasti, kiinnittäen huomiota kuitenkin siihen, että haastattelu pysyy aiheessa.

Teemahaastatteluissa, joissa haastateltava kertoi omista näkemyksistään ja kokemuksistaan monisanaisesti, pyrittiin hyödyntämään "itseäänkorjaavaa" haastattelua. Itseäänkorjaavassa haastattelussa haastattelija ensin kuuntelee tarkkaan haastateltavan sanoman, jonka jälkeen haastattelija tiivistää ja tulkitsee sanoman jälleen haastateltavalle, jolloin haastateltava voi vahvistaa tai hylätä esitetyn tulkinnan tai tiivistyksen (Hirsjärvi & Hurme, 2015).

Tiedonkeruumenetelmänä haastattelusta tulisi huomioida se, että käsitteet, merkitys ja kieli ovat haastattelun perusta: tässä tutkielmassa haastatteluissa on pyritty siihen, että haastattelija ja haastateltava jakaisi mahdollisimman pitkälle saman kielen ja käsitteistön (Hirsjärvi & Hurme, 2015). Tämä on pyritty huomiomaan tässä tutkielmassa siten, että haastattelija on työskennellyt organisaatiossa puolentoista vuoden aikana eri organisaation tasoilla eri roolien sekä eri yksiköiden parissa luoden itselleen ymmärrystä organisaatiossa käytetyistä termeistä ja niiden merkityksistä.

Haastatteluun tiedonkeruumenetelmänä liittyy myös monia haasteita: ensinnäkin haastattelujen pitäminen vaatii haastattelijalta osaamista, jotta

tiedonkeruu olisi optimaalista. Toiseksi haastattelut sisältävät monia virheitä: virheitä tulee niin haastattelijan kuin haastateltavan toimisesta. Kolmanneksi aineiston analysointi on vaikeaa, sillä aineiston analysointiin, tulkintaan ja raportointiin ei ole olemassa valmista mallia. Viimeisenä haasteena on se, että aineiston keruussa saadaan valtavasti sellaista materiaalia, joka ei kosketa tutkimusaihetta tai tutkimuskysymystä. (Hirsjärvi & Hurme, 2015) Myös tämän tutkielman analysointivaiheessa oli haasteita sen suhteen, että kiinnostavaa materiaalia oli paljon ylimääräistä, mikä aiheutti haasteita sille, että kaiken materiaalin seasta tutkielman tekijä tunnistaisi olennaisen aineiston.

3.3.1 Haastateltavien valinta ja haastateltavien taustatiedot

Tavoitteena oli saada haastattelu 6-9 henkilöltä aineiston keräämiseksi, lopullinen haastateltavien määrä oli kahdeksan henkilöä. Haastateltavien valinnassa hyödynnettiin lumipallo-otantaa, jossa aluksi tunnistettiin ketteryuden avaintekijöitä, eli henkilöitä, jotka edistävät Kelaa ketterien menetelmien ja ketteryuden omaksumisessa (Tuomi & Sarajärvi, 2009). Ensimmäisten haastateltavat valittiin sen mukaan, joiden katsottiin ymmärtävän, minkälaisia onnistumisia ja haasteita Kelalla on ketteryuden suhteen. Tämän jälkeen nämä avaintekijät nimesivät ja suosittelivat haastatteluun sopivia ihmisiä, jos useampi haastateltava mainitsi saman henkilön nimeltä, hänet valittiin potentiaalisesti haastateltavaksi. Yksikään potentiaalisista haastattelijoista ei kieltäytynyt haastattelusta, joten tätä tutkimusta varten on saatu ensimmäisten haastateltavien suosittamat avaintekijät mukaan.

Tutkimuksen tavoitteena oli tutkia ja arvioida ketteryuden periaatteiden ja arvojen omaksumista koko organisaatiossa IT:n osalta, joten rajaaminen yhteen yksikköön olisi antanut tietoa vain yhden yksikön tilasta. Haastateltavien valinnassa huomioitiin erityisesti kaksi asiaa: ensinnäkin se, että haastateltavat eivät olisi kaikki samasta organisaatioyksiköstä ja toiseksi se, että haastateltavat olisivat roolien puolesta avainasemassa ketteryuden suhteen. Nämä kaksi asiaa toimivat haastateltavien pääasiallisina valintakriteereinä. Myös haastateltavien rooli vaikutti olennaisesti haastateltavien valintaan, sillä Turetkenin ym. (2016) kypsyystasomallissa haastateltavina tulisi käyttää nimenomaan Scrum Mastereita ja RTE:n roolissa toimivia henkilöitä. Tätä tutkielmaa varten valittiin myös näiden roolien ulkopuolelta olevia henkilöitä kahdesta syystä. Ensinnäkin Scrum Masterilla ei ole välttämättä laajempaa ymmärrystä eri tiimien ja eri organisaatioyksiköiden toiminnan tilasta, kun taas esimerkiksi agile coachilla, RTE:llä tai yksikön päälliköllä on. Ja toiseksi Scrum Masterin ja RTE:n roolin ulkopuolella olevat kyseiset henkilöt toimivat sellaisessa roolissa, jossa heillä on mahdollisuus edistää ketterien menetelmien ja ketteryuden omaksumista organisaatiossa ja/tai henkilöllä on laaja-alaista ymmärrystä ketteryuden omaksumisesta organisaatiossa.

Haastattelujen alussa haastateltavilta kysyttiin taustatietoja heidän työroolistaan, työkokemuksesta Kelassa, kokemusta ketteristä menetelmistä sekä formaaleista ketterien menetelmien koulutuksista ja kokemusta

perinteisistä menetelmistä. Haastateltavat työskentelevät Kelassa Scrum Masterina, Agile Coachina, Release Train Engineerinä (RTE), asiantuntijana sekä yksikön päällikkönä. Haastateltavien työkokemus Kelassa vaihteli alle 2 vuodesta yli 20 vuoteen. Kelassa Scrum Master, Agile Coach, RTE sekä asiantuntija osana ketterää tiimiä kuuluvat Kelan kehittämismallin rooleihin. Ainoastaan yksikön päällikön rooli kuuluu linjaorganisaation rooliin. Yksi haastateltavista toimii kahden tiimin Scrum Masterina ja vastauksissaan hän peilaa molempien tiimien tilannetta. Muilla kuin Scrum Mastereilla ja asiantuntijoilla on käsitys vähintään 15 tiimin toiminnasta.

Kokemusta ketteristä menetelmistä haastatteliijoilla oli keskimäärin noin 7 vuotta. Kelassa ensimmäisiä ketterien menetelmien projekteja alettiin ottamaan käyttöön noin kuusi vuotta sitten. Yksi haastateltavista oli kerryttänyt teoriapuolen kokemusta ketteristä menetelmistä noin kymmenen vuoden ajan. Lisäksi haastateltavien kokemus ketterien menetelmien koulutuksista vaihteli: haastateltavista seitsemän oli käynyt esimerkiksi Scrum Master sertifiointikoulutuksen tai tuoteomistajakoulutuksen. Suurimmalla osalla sertifiointi ei ollut enää voimassa tai haastateltava oli käynyt pelkän koulutuksen, muttei läpäissyt tai tehnyt sertifiointia. Vain yksi haastateltavista oli käynyt pelkästään Kelan sisäisesti järjestettävä tuoteomistajakoulutuksen.

Haastateltavien kokemus perinteisistä projektimenetelmistä vaihteli 0 vuodesta aina yli 20 vuoteen ja henkilöt, joilla oli perinteisistä projektimenetelmistä kokemusta, olivat jossain vaiheessa työuraansa toimineet projektipäällikkönä Kelalla tai muissa organisaatioissa. Kolmella haastateltavalla kokemus perinteisistä projektimenetelmistä oli 0 vuotta.

3.3.2 Haastattelujen suunnittelu ja toteutus

Myers ja Newman (2006) on tunnistaneeet kirjallisuuden kautta laadullisiin haastatteluihin liittyviä haasteita, ongelmia ja sudenkuoppia, jotka olisi hyvä ottaa huomioon haastattelua suunnitellessa. Haastattelutilanne on lähtökohtaisesti keinotekoinen tilanne, jossa tuntematon haastattelija kuulustelee haastateltavaa rajatussa ajassa. Haastattelija on yleensä täysin tuntematon henkilö, joten haastatteludataan voi vaikuttaa sen, uskaltaako haastateltava jakaa hänen mielestään sensitiivistä informaatiota. Tämä voi johtaa siihen, että kerätty data ei ole kattavaa. (Myers & Newman, 2006). Tässä tutkielmassa nämä kaksi sudenkuoppaa on pyritty huomioimaan sillä, että ensinäkin haastattelija on tutkittavassa organisaatiossa töissä ja haastatteluun kutsuttaessa haastateltava kertoi itsestään ja omasta taustastaan tuoden itseään hieman tutumaksi haastateltaville. Osa haastateltavista henkilöistä oli ollut tekemisissä haastattelijan kanssa työtehtävien merkeissä jo ennen haastatteluja. Eli kaikille haastateltaville haastattelija ei ollut täysin tuntematon henkilö. Toiseksi haastateltaville pyrittiin luomaan turvallinen ympäristö jakaa myös sensitiivisempää informaatiota siten, että haastattelun vastauksissa käytetään nimen tai roolin sijaan tunnistetta esimerkiksi "H1".

Ennen oikeita haastatteluja tutkielman kirjoittaja piti testihaastattelun organisaation työntekijälle. Tällä tutkielman kirjoittaja pyrki siihen, että hän saisi paremman käsityksen siitä, miten toimia haastattelijana ja toisaalta tämä oli mahdollisuus tutkia teemahaastattelun toimivuutta tutkittavaan ilmiöön.

Haastattelut pidettiin vuosien 2021–2022 aikana ja keskimäärin haastattelut kestivät noin 75 minuuttia. Haastatteluiden alussa haastatteli kävi haastateltavan kanssa läpi tutkielman aihepiirin ja empiirisen materiaalin käsittelyyn ja tallentamiseen liittyvät periaatteet. Tämän jälkeen haastattelun alussa käytiin haastateltavien taustatiedot läpi, jonka jälkeen aloitettiin varsinainen teemahaastattelu. Haastattelut etenivät teema kerrallaan. Haastateltavilla oli mahdollisuus kysyä tarkentavia kysymyksiä haastateltavalta ja toisaalta haastateltava pystyi itse tarkentamaan omaa vastaustaan, vaikka teemassa olisikin jo menty eteenpäin. Kaikki haastattelut nauhoitettiin ja litteroitiin datan analysointia varten.

3.4 Aineiston analysointi

Teemahaastatteluiden kautta saatu litteroitu aineisto analysoitiin sisällönanalyysin avulla. Tuomen ja Sarajärven mukaan (2008) sisällönanalyysi on tekstianalyysi, jonka tarkoitus on luoda aineistoon selkeyttä ja sitä kautta mahdollistaa selkeiden ja luotettavien johtopäätösten tekeminen tutkittavasta ilmiöstä. Hsieh ja Shannon (2005) näkevät, että yhden menetelmän sijaan sisällönanalyysi pitää sisällään kolme eri lähestymistapaa: tavanomainen sisällönanalyysi, suunnattu sisällönanalyysi sekä summaava sisällönanalyysi.

Sisällönanalyysi soveltuu hyvin strukturoimattomankin aineiston analysointiin, mikä oli aineiston analysointimenetelmää valitessa olennainen kriteeri, sillä osa teemahaastatteluista oli luonteeltaan melko avoimia haastattelutyypiltään. Sisällönanalyysia on kritisoitu siitä, että johtopäätösten tekeminen on jäänyt tekemättä, kun järjestetty aineisto on nähty johtopäätöksinä. (Tuomi & Sarajärvi, 2008)

Tässä tutkielmassa käytetty sisällönanalyysi noudattaa tarkemmin Hsiehin ja Shannonin tunnistamaa suunnattua sisällönanalyysin lähestymistapaa. Suunnatussa sisällönanalyysissa lähtökohtana toimii teoreettinen viitekehys tai teoria, joka ohjaa koodien muodostamista ja lähestymistavan tavoitteena on validoida tai laajentaa teoreettista viitekehystä tai teoriaa (Hsieh & Shannon, 2005). Tämän tutkielman yhtenä tavoitteena oli tutkia sitä, kuinka hyvin Turetkenin ym. (2016) SAFe MM kypsyyntasomalli soveltuu julkishallinnollisen organisaation kypsyyntason mittaamiseen eli kasvattaa ymmärrystä teoreettisen viitekehysten sopivuudesta empiriaan. Lisäksi suunnatun sisällönanalyysin käytössä tulisi huomioida se, että haastatteluissa avoimet kysymykset, joita seuraa tarkempia kysymyksiä ennalta määrättyjen kategorioiden mukaisesti, sopii parhaiten menetelmän käyttöön. (Hsieh & Shannon, 2005) Tässä tutkielmassa teemahaastattelujen käyttäminen tiedonkeruumenetelmänä tukee hyvin aineiston analysoinnissa käytettävää menetelmää.

Suunnatussa sisällönanalyysissa sisällönanalyysia ohjaa teoreettinen malli, tässä tutkielmassa teoreettisena viitekehyksenä toimi Turetkenin ym. (2016) SAFe MM -kypsyystasomalli. Olemassa olevaa teoriaa käyttäen avainasiat tunnistetaan ja ne toimivat ensimmäisinä koodauskategorioina, tässä koodauskategorioina toimivat SAFe MM -mallin 63 käytäntöä. Tämän jälkeen eri koodauskategorioille annetaan selitys teorian pohjalta, tässä tutkielmassa selitykset löytyvät alaluvun 2.6.2 SAFe MM tasot, periaatteet ja käytännöt taulukosta (TAULUKKO 4). Teksti, jota ei pysytä kategorisoimaan ensimmäisten koodauskategorioiden mukaisesti luokitellaan uuden koodin mukaisesti. (Hsieh & Shannon, 2005) Aineiston tarkemmassa jaottelussa ja tunnistamisessa ne kohdat, jotka eivät osuneet ensimmäisiin koodauskategorioihin, jaoteltiin SAFe MM -mallin periaatteiden koodauskategorioihin: asiakasarvo, suunnittelu ja jatkuva tuottaminen, ihmiskeskeisyys, tekninen huippuosaaminen sekä asiakasyhteistyön. Ne aineiston kohdat, jotka eivät sopineet teoriamallin kategorioihin, mutta olivat kuitenkin tutkimuskysymyksen kannalta relevantteja, tunnistettiin ja nostettiin heti empiirisen tutkimuksen tulokset osan alkuun.

Suunnatun sisällönanalyysin hyvänä puolena on se, että olemassa olevaa teoriaa voidaan tukea ja laajentaa, sillä empiiriset löydökset ovat joko teoriaa tukevia tai ei-tukevia löydöksiä. Toisaalta, juuri tämä on suunnatun sisällönanalyysin huono puoli: tutkijat saattavat löytää enemmän teoriaa tukevia kuin ei-tukevia todisteita ja teoriapohja saattaa ohjata tutkijaa korostamaan teoriaa, samalla unohtaen ilmiön kontekstuaaliset kohdat. (Hsieh & Shannon, 2005)

4 EMPIIRISEN TUTKIMUKSEN TULOKSET

Tässä luvussa esitellään empiirisen tutkimuksen tulokset. Tulokset on teemoitettu aihealueittain. Teemat ovat syntyneet Turetkenin ym. (2016) kypsyystasomallin periaatteiden sekä aineiston kautta esiin nousseiden aihealueiden kautta. Ensimmäisenä esitellään aineiston kautta esiin nousseet teemat ja aihealueet ja sen jälkeen teoriapohjan mukaiset koodauskategoroihin kuuluvat empiiriset havainnot.

4.1 Organisaatiokulttuuri ja toimintatavat ketteryiden omaksumisen esteenä

Tässä alaluvussa on koottuna yhteen aineiston kautta nousseita teemoja ja havaintoja siitä, mitkä toimivat Kelan ketteryteen ja ketterien arvojen ja periaatteiden omaksumisen esteenä. Tähän on koottua sellaiset teemat, jotka eivät sovi tarkemmin Turetkenin ym. (2016) SAFe MM -mallin periaatteisiin.

Ketterien menetelmät perustuvat ihmisenäkökulmaan, joten ketterien menetelmien onnistunut käyttö riippuu nykyisen organisaatiokulttuurin ja menetelmien yhteensopivuudesta (Tolfo ym., 2009). Ihmisenäkökulma näkyy vahvasti ketterän ohjelmistokehityksen julistuksessa. Julistuksen ensimmäinen arvo on "Yksilöitä ja kanssakäymistä enemmän kuin menetelmiä ja työkaluja ja kahdessatoista periaatteessa ihmiskeskeisyys näkyy muun muassa sanoissa "yhteistyö", "motivoituneet yksilöt", "luottamus", "kommunikointi" ja "ihmiset" (Beck ym., 2001). Ketteryyden voidaankin sanoa perustuvan avoimuuteen ja luottamukseen, jossa virheitä ja kokeiluja uskalletaan tehdä.

Aineistosta nousi ihmiskeskeisyyteen liittyviä havaintoja, jotka osoittavat, että organisaatiossa ei vallitse optimaalinen kulttuuri ketteryydelle. Se, miksi organisaatiokulttuuri on muovautunut sellaiseksi kuin se on, johtuu monista tekijöistä, mutta yksi asia on ihmisten johtaminen eli johtajuus. Hyvä johtajuus edistää tiimin ja ryhmän tehokkuutta, mikä puolestaan luo hyvinvointia (Hogan

& Kaiser, 2005). Datan perusteella työntekijöihin saatetaan suhtautua resurssimaisesti unohtaen, että työntekijät ovat yksilöllisiä.

”Et tavallaan välillä tuntuu varsinkin linjaorganisaation suunnalta, että ei ajatella ihmisinä -- Jotenkin meillä on ihmisiä, ne on resursseja, ne on päänuppia, jotka on kaikki samanlaisia. -- Tavallaan sitten kun lähdetään siitä [RYHMÄPÄÄLLIKÖISTÄ] ylöspäin, niin unohtuu se yksilöllisyys. - Haastateltava H6

”Alihankkijat nähdään semmoisena vaihdettavissa olevina resursseina, että siihen ei lasketa mitään hintaa sille yliheittovaiheelle ja palvelun tason laskulle mikä siinä tulee. Mä kyllä näen, että sitä resurssiajattelua tapahtuu, mutta ei meidän tiimin kohdalla” - Haastateltava H7

Nämä havainnot saattavat olla ääriesimerkkejä siitä, minkälaista suhtautumista työntekijöihin on organisaatiossa olemassa ja havainnot eivät välttämättä edusta kuin pientä osaa organisaatiossa. Osalla haastateltavista oli kokemuksia myös siitä, että johtotasolla ei ymmärretä yksilöllisyyttä ja sitä, mistä työhyvinvointi koostuu.

”Piti ratakiskosta vääntää välillä tuossa, että ei, mä en halua et tätä tiimiä räjäytetään, koska silloin te räjäytätte koko noiden ihmisten työhyvinvoinnin.” - Haastateltava H6

Nykyisen organisaationkulttuurin tarkempi tutkiminen vaatisi omaa tutkimusta, mutta huomioitavaa on se, että organisaation kulttuuri ei ainakaan kaikissa tiimeissä luo mahdollisuuksia toimia ketterien arvojen ja periaatteiden mukaisesti.

”Psykologisesta turvallisuudesta, niin jollain tavalla sellaista syyllisen etsimistä tai pelkoa tai jotain sellaista on, joka estää ihmisiä osallistumasta. Semmoista kulttuuritaustaa minusta on kyllä tunnistettavissa tässä. -- Täällä myös ilmenee hyvin suoranaista käytöstä välillä, joka ei kuuluisi työpaikalle. Jos alkaa hirveästi tuomaan uusia ideoita ja kommentoimaan nykytoimintoja, niin voi saada aikamoiset lataukset niskaansa.” - Haastateltava H2

”Välillä tuntuu siltä, et ihan niin kuin ihmisiltä ei saisi vaatia tarpeeksi. On jotain semmoista itsetyytyväisyyttä, joka on saatu kulttuurillisesti luotua, että minä nyt osaan tämän jutun ja mun ei tarvitsekaan opetella mitään muuta.” - Haastateltava H2

Organisaatiokulttuurissa saattaa olla piilossa erilaisia esteitä ketterien menetelmien omaksumiselle. Kelan organisaatiokulttuurin alle piiloutuneet esteet saattaisivat olla johtamistyyli, asenteet toisia työntekijöitä kohtaan, sekä työympäristössä vallitsevat arvot, jotka eivät välttämättä edusta johtajien julistamia organisaation arvoja (Tolfo ym., 2009). Tämä muodostaa empiirisen johtopäätöksen EC1.

EC1: Kelan organisaatiokulttuuriin kuuluu ketteryuden omaksumisen esteitä. Nämä esteet ovat henkilöstöjohtaminen sekä kulttuurissa piilossa olevat työympäristön arvot.

Ketteryyden omaksumisen esteenä voi toimia myös vanhojen rakenteiden ylläpitäminen ketterän kehittämisen rakenteiden rinnalla.

”Samaan aikaan jäi olemaan esimerkiksi meidän ohjausryhmä. Eli jäi olemaan vanhat foorumit sen lisäksi.” - Haastateltava H4

”Toimintamallista poiketen meillä on ohryn kokoukset. - Haastateltava H6

”Scrumissa esimerkiksi halutaan toimivan mutta samaan aikaan meillä on osa organisaatioista edelleen ylläpitää projektimaisia tapoja ja rakenteita -- tiimejä synnytetään tietyn asian tekemiseksi ja sitten niitä miksatetaan yhteen tai erotellaan. Eli ei toimita sillä idealla, että meillä on Scrum-tiimi, joka saa tehtäviä vaan meillä on tehtäviä, joihin etsitään ihmisiä tekemään niitä tehtäviä. -- Ylläpidetään ihmeellistä rakennetta menneisyydestä.” - Haastateltava H6

Ketterän kehittämisen rakenteiden rinnalla ylläpidetään vanhoja rakenteita, kuten ohjausryhmiä. Tämä muodostaa empiirisen johtopäätöksen EC2.

EC2: Ketterän kehittämisen rakenteiden rinnalla ylläpidetään vanhoja rakenteita, kuten ohjausryhmiä.

Vanhojen rakenteiden ylläpitäminen, kuten ohjausryhmät eivät välttämättä palvele ketteriä tiimejä oikealla tapaa, mutta ketteriltä tiimeiltä ja ketteryyttä tukevilta tahoilta saattaa puuttua sellainen foorumi tai taho, joka tukisi ketteryyteen kuuluvien toimintatapojen edistämistä. Tätä tulkintaa tukevat haastateltavan vastaukset:

”Me ollaan annettu paljon Kelalle asioita ja tapoja hoitaa asioita --ja kokeiltu kaiken maailman härpäkkeitä. On semmoinen olo, että ei me kuitenkaan voida vaikuttaa mihinkään. Se voi olla, että kohta tulee jostain ylempää, että voitte muuten deletoida noi. Kun ei ole sitä suhdetta meille, niin en koe olevani lähellä ja sen takia mulla on semmoinen olo, että mä en pysty vaikuttamaan.” - Haastateltava H4

”Musta se oli aivan loistavaa ajatuksena, että mulla on se kummiluotsi, jonka kanssa teen yhteistyötä, joka näkee mitä ongelmia meillä täällä tasolla on. Ja, jonka kautta pystyn viemään asioita eteenpäin.” - Haastateltava H4

Haastateltavan vastauksesta voisi tulkita, että organisaatiolta puuttuu sellainen foorumi tai taho, missä Kela-tason kehittämisorganisaatioon liittyviä asioita käsitellään, oli kyse asioiden päättämisestä tai haasteiden ratkomisesta. Tämä muodostaa empiirisen johtopäätöksen EC3.

EC3: Organisaatiolta puuttuu sellainen foorumi, jossa Kela-tason kehittämisorganisaatioon liittyviä haasteita ratkotaan.

Organisaationkulttuuri ja toimintatavat ketteryyden omaksumisen esteenä - teemaan löytyi yhteensä 3 empiiristä yhteenvetoa. Näiden yhteenvetojen pohjalta muodostettiin ensisijainen empiirinen johtopäätös PEC1.

PEC1: Kelan organisaatiokulttuurin ja -toimintaan kuuluu erilaisia asioita, jotka toimivat ketteryyden omaksumisen esteenä.

Organisaatiokulttuuriin liittyvät empiiriset johtopäätökset ovat esiteltyinä seuraavassa taulukossa (TAULUKKO 6).

TAULUKKO 6 Aineiston kautta nousseet empiirisen johtopäätökset

Tunniste	Empiirinen johtopäätös
EC1	Kelan organisaatiokulttuuriin kuuluu ketteryyden omaksumisen esteitä. Nämä esteet ovat henkilöstöjohtaminen sekä kulttuurissa piilossa olevat työympäristön arvot.
EC2	Ketterän kehittämisen rakenteiden rinnalla ylläpidetään vanhoja rakenteita, kuten ohjausryhmiä.
EC3	Organisaatiolta puuttuu sellainen foorumi, jossa Kela-tason kehittämisorganisaatioon liittyviä haasteita ratkotaan.
PEC1	Kelan organisaatiokulttuurin ja -toimintaan kuuluu erilaisia asioita, jotka toimivat ketteryyden omaksumisen esteenä.

4.2 Asiakasarvon tuottaminen

Asiakasarvon tuottaminen on kypsyytasomallin ensimmäinen periaate. Tähän alalukuun on koottuna asiakasarvon tuottamisen periaatteeseen liittyviä havaintoja, ja havainnot ovat jaoteltuna tarkempiin arvioitaviin käytänteisiin. Ensimmäisenä kuitenkin esitellään asiakasarvon tuottamisen periaatteeseen liittyviä havaintoja, jotka eivät suoraan liity arvioitavien käytänteiden kategorioihin. Asiakastyytyväisyyden ja asiakasarvon tuottaminen on ketteryyden tärkein periaate: tarkoituksena on tuottaa asiakkaalle arvoa mahdollisimman nopeasti ja aikaisessa vaiheessa (Beck ym., 2001). Yksi haastateltava toteaa, että asiakasarvon tuottaminen ei välttämättä ole aina organisaation omissa käsissä sen toimeenpanevan luonteen takia.

”Tulee pakollisia lakimuutoksia, olen monesti joutunut sanomaan, että ei tällä muutoksella ole mitään tekemistä asiakasarvon kanssa. Mutta se on pakko tehdä, koska se lukee laissa. Joudutaan tekemään oikeasti vähemmän arvoa tuottavia asioita vain sen takia, että se lukee laissa, ja siellä on joku määräaika.” - Haastateltava H4

Lakimuutokset saattavat toimia järjestelmien kehittämisen ajureina, jolloin järjestelmän kehittäjätiimit eivät pysty vaikuttamaan siihen, tuottaako tarvittava muutos asiakasarvoa. Tämä muodostaa empiirisen johtopäätöksen EC4.

EC4: Lakimuutokset saattavat toimia järjestelmien kehittämisen ajureina, jolloin järjestelmän kehittäjätiimit eivät pysty vaikuttamaan siihen, tuottaako tarvittava muutos asiakasarvoa.

Asiakasarvon tuottaminen niissä tilanteissa, joissa se on mahdollista, ei ole myöskään nopeaa.

”Jos [ASIAKASARVON] pitäisi olla fokuksessa, niin hirveän hitaasti edelleen me tuotamme sitä.” – Haastateltava H4

Asiakasarvon tuottamisen hitaus voi liittyä organisaatiokontekstin lisäksi myös tavoiteasetantaan. Joko asetetut tavoitteet ovat liian kunnianhimoisia, liian kevyitä tai tavoitteita ei aseteta, mikä johtaa siihen, että kehitystiimit eivät keskity oikeiden asioiden tuottamiseen ja asiakasarvon tuottaminen on hidasta. Tätä tulkintaa tukee haastateltavan vastaukset:

”Suoraan sanottuna tavoitteen asetanta on saattanut olla hyvin kevyt.” – Haastateltava H2

”Nyt me tavoitellaan jotain tuotantoon julkaistavaa uutta versiota, joka sisältää valtavan palon sisältä. Näistä voisi tehdä pienempiä julkaisuja. – Haastateltava H6

Liian kevyt tavoiteasetanta voi liittyä organisaation kulttuuriin. Tavoitteita ei uskalleta asettaa korkealle, sillä pelätään, että työtehtäviä ei saadakaan suoritettua. Tätä tulkintaa tukee yhden haastateltavan vastaukset:

”Me seurataan velositeettiä ja halutaan, että suunniteltu työ vastaisi sitä toteutunutta työtä ja korostetaan, että tiimillä on oikeus sanoa, että ei käy, nämä me ehditään tekemään. Vai haluatko vaihtaa jotain? Niin ei sitä kuitenkaan tapahdu.” – Haastateltava H4

”Kelassa on tosi sitoutunutta väkeä ja väitän, että meillä on hyvin paljon ylisitoutuneita ihmisiä.” – Haastateltava H4

Liian kunnianhimoiset tavoitteet saattaisivat johtua siitä, että organisaation kulttuurissa ei ole totuttu sanomaan ”ei”. Toisaalta työn ajatellaan koostuvan itse tehtävän suorittamisesta, mutta esimerkiksi osaamisen laajentaminen ja työhön liittyvä metatyö, kuten tehtävän suunnittelu, ja oman sekä tiimin kapasiteetin määrittäminen nähdään toissijaisena asiana.

”Mä en opettelekaan tota asiaa tai tarkoittaisi tämä sitä, että me ei kehitetäkään tätä meidän toimintaa tai mikä vaan, että jätetään tavallaan ne ketteryyden edellyttämätkin asiat pois ymmärtämättä, että se tuhoaa sen koko ketteryyden.” – Haastateltava H4

Organisaatiolla on puutteita asettaa oikeantasoiset ja -suuriset tavoitteet, mikä saattaa liittyä organisaatiokulttuuriin. Tämä muodostaa empirisen johtopäätöksen EC5.

EC5: Organisaatiolla on puutteita oikeantasoisien ja -suuruisten tavoitteiden asettamisessa, mikä saattaa liittyä organisaatiokulttuuriin.

Organisaatiolla on omia tapoja varmistaa, että kehitystiimeissä tehdään asiakaslähtöistä kehittämistä. Yhtenä erityispiirteenä organisaatiossa on se, että

järjestelmiä käyttäviä sisäisiä asiakkaita hyödynnetään palkkaamalla heidät osaksi kehitystiimiä esimerkiksi tehtävankiertojen kautta.

”Meillä on testaaja, tuli mukaan pari vuotta sitten, ja hän on tullut suoraan kentältä siitä käyttäjäryhmästä, joka sitä tulee käyttämään eli sen varaan on laskettu tosi paljon, että meillä on suora yhteys sinne kentälle.” – Haastateltava H6

”Meillä on jonkun verran porukkaa tullut kentältä, että kyllähän heillä on se käsitys, että miltä se siellä näyttää. – Haastateltava H3

Se, että asiakas otetaan osaksi kehitystiimiä kehittämään tuotetta käyttäjien tarvitsemaan suuntaan, osoittaa hyvin asiakaslähtöistä ja siten myös ketterien menetelmien mukaista ajattelutapaa. Kehitystiimi voi saada hyvinkin arvokasta näkemystä siihen, miten tuotetta kannattaisi kehittää, jotta se tuottaisi asiakkaalle mahdollisimman paljon arvoa. Tämä muodostaa empiirisen johtopäätöksen EC6.

EC6: Käyttäjän ottaminen osaksi kehitystiimiä mahdollistaa asiakaslähtöisen kehittämisen.

4.2.1 Prosessin reflektointi ja parantaminen

Prosessin reflektoinnilla ja parantamisella pyritään siihen, että tunnistetut haasteet eivät toistuisi seuraavilla kehityskerroilla. Tähän prosessin kehittämiseen vastaa retrospektiivien pitäminen säännöllisesti. (Turetken ym., 2016) Kuusi haastateltavaa totesi, että tiimeillä on retrospektiivit käytössä säännöllisesti ja toiminnan reflektointia tehdään.

”Meillä on joka sprintissä sprintin lopussa retrot.” – Haastateltava H1

”Meillä on ihan Scrum-mallin mukaisesti reviewt ja retrot” – Haastateltava H5

”Sprintti retro on.” – Haastateltava H6

”Retrot -- pidetään säännöllisesti.” – Haastateltava H8

”Tiimeillä on retroja, mutta toisaalta kyllä tehdään erinäisiä asiakastyytyväisyyskyselyitä.” – Haastateltava H2

”Kyllä, sanotaan, teknisesti tätä toteutetaan.” – Haastateltava H5

Ilmaisun ”teknisesti tätä toteutetaan” voisi viitata siihen, että prosessin reflektoinnissa ja parantamisessa jompikumpi jää toteutumatta. Kahden haastateltavan vastaukset tukevat sitä tulkintaa, että reflektointia tapahtuu, mutta prosessi ei parane, sillä kehittämiskohteita ei oteta aina käyttöön:

”Siinä olisi parantamisen varaa et ne [KEHITTÄMISKOHTEET] tulisi oikeasti tehtyä.” – Haastateltava H1

”Testausautomaation kannalta on kirjattu ylös [KEHITTÄMISKOHTEITA], joissa olisi parantamisen varaa -- ne tuppaa olemaan pitkässä lieassa ihan sen takia, että on vähän resursseja, ja se kaikki [AIKA] menee melkein itse käytännön sovelluskehittämiseen.” - Haastateltava H5

”Keskustelin tiimiläisen kanssa, että hän on todella turhautunut. Hän oli katsonut, että hän on vuosi sitten retrosta tuonut yhden asian esille ja toistanut sitä ja aina sanotaan, kun ei meillä ole aikaa tähän. Tai sitten siihen ei ole palattu.” - Haastateltava H4

Yhden haastateltavan vastaus tukee sitä, että tiimeissä aidosti reflektoidaan ja kehitetään omaa toimintaa.

”Tehdään käytännön parannusehdotus heti ja pyritään ottamaan se heti käyttöön.” - Haastateltava H6

Prosessin reflektointia ja parantamista tapahtuu, mutta ei kaikilla osa-alueilla systemaattisesti. Tämä muodostaa empiirisen johtopäätöksen EC7.

EC7: Prosessin reflektointia ja parantamista tapahtuu, mutta ei kaikilla organisaation osa-alueilla eikä täysin systemaattisesti. (L)

4.2.2 Vaatimusten evolutiivisuus

Vaatimusten evolutiivisudella tarkoitetaan sitä, että vaatimuksia kehitetään koko ajan asiakaspalautteen myötä sen sijaan, että vaatimukset kirjattaisiin hyvin tarkalla tasolla, yhdellä kertaa. (Turetken ym., 2016) Vaatimusten evolutiivisuutta korostettiin haastatteludatassa kahdessa tapauksessa.

”Meillä on aika hyvin ketterää määrittelyä. -- Ei ole missään nimessä mitään vesiputousmallia käytössä.” - Haastateltava H1

”Ei jäädä odottelemaan, että joku muu määrittelisi, vaan me tehdään itse paras mahdollinen määrittely- Haastateltava H8

”[Vaatimusmäärittelyä] tehdään hirveän tarkalla tasolla, ennen kuin on annettu toteuttajille kommentoitavaksi ja yleisfiilis on, että se on vielä aika tiukassa täällä tämä vanha ajattelu.” - Haastateltava H2

Vastauksista voisi tulkita sen, että aiemmin vaatimusten tarkka kirjaaminen ennen ohjelmoinnin aloittamista on ollut tapana, mutta toimintatavat ovat muuttuneet entistä evolutiivisempaan suuntaan. Vahva asiakaspalautteen kautta vaatimusten kehittäminen ei näkynyt datassa, mikä voi johtua organisaation kompleksisesta toimintakentästä, jossa palveluita kehitetään usealle asiakkaalle ja asiakas on vaikea tunnistaa. Tätä tulkintaa vahvistaa haastateltavan vastaus:

”En mä ainakaan osais kysyä tai sanoa, että keneltä lähdetään kysyä, kokeeko asiakas meidän toiminnan hyväksi tai ollaanko me onnistuttu asiakkaan näkökannasta.” - Haastateltava H1

Vaatimusmäärittely on kehittynyt koko ajan evolutiivisempaa suuntaan, mutta vaatimusten kehittäminen asiakaspalautteen myötä on paikoittain haasteellista toimintaympäristön vuoksi. Tämä muodostaa empiirisen johtopäätöksen EC8.

EC8: Vaatimusmäärittely on aiempaan verrattuna evolutiivisempaa, mutta vaatimusten kehittäminen asiakaspalautteen pohjalta on haasteellista toimintaympäristön vuoksi. (P)

4.2.3 Pienemmät, useammat julkaisut

Pienemmällä, useammalla julkaisulla mahdollistetaan parempi reagointikyky ja organisaatioon kohdistuvien riskien pieneneminen. (Turetken ym., 2016) Pienemmät, useammat julkaisut ovat ainakin osalla tiimeistä käytössä, ja haastateltavien vastauksissa tämä nousi esille kolmessa vastauksessa:

”Meillä on käyttöönottopäiviä kolme vuodessa, missä isot muutokset yleensä viedään. Tässä verkkoasioinnissa ne tekee pienempiä julkaisuja siellä.” –Haastateltava H2

”Kyllä, joka viikoittainhan on tällaiset tomut eli torstaisin viedään muutoksia tuotantoon. Tässä sprintin aikana muutokset etenee, niillä tähdätään johonkin tomuun, että sen sprintin aikana saataisiin asiat vietyä tuotantoon.” – Haastateltava H5

”Oliko se jo yli 20 julkaisua vai melkein keskimäärin joka toinen viikko jotain vietiin tuotantoon” – Haastateltava H8

Kaikki tiimit eivät tee kuitenkaan pienempiä, useampia julkaisuja.

”Nyt ne [JULKAISUT] tulee hirveän isoissa erissä.” – Haastateltava H6

Vastausten perusteella voisi tulkita, että pienempien, useampien julkaisujen teko on melko kattavasti tiimeillä käytössä. Joissain organisaation yksiköissä tai tiimeissä asioita lähestytään yhä perinteisemmällä projektimaisella ajatuksella, jossa projektin lopuksi pyritään saamaan lopullinen tuote valmiiksi. Tätä tulkintaa vahvistaa yksi haastateltavan vastaus:

” -- julkaisujen pilkkominen pienemmäksi, niin se on ehkä suurin opittava asia mikä musta tuntuu et on vaikein tiimillekin mieltää, että me voitais tehdä julkaisuja pienemmissä, nopeissa sykleissä ja niin, että ei ne tarvii olla semmoisia, mitkä viedään tuotantoon asti.” – Haastateltava H6

Pienempien, useampien julkaisujen tekeminen on tiimeillä jo melko laajasti käytössä, kuitenkin kaikki tiimit eivät toimi tällä mallilla. Tämä muodostaa empiirisen johtopäätöksen EC9.

EC9: Pienemmät, useammat julkaisut ovat melko laajasti tiimeillä käytössä, mutta osittain julkaisut ovat vielä isoja, mitä viedään tuotantoon harvoin. (L)

4.2.4 Vaatimusten löytäminen

Tarkoituksena on ymmärtää paremmin, mitä pitää rakentaa ja miksi. Esimerkiksi Spikes, käyttötapausmallinnus ja käyttäjäkokemusmallit ovat olemassa ohjelmistovaatimustekniikoita, joita suositellaan käytettävän asiakkaan toivomusten ymmärtämisessä. (Turetken ym., 2016) Erilaisten ohjelmistovaatimustekniikoiden käyttäminen on kolmen haastateltavan vastauksen perusteella käytössä.

”Käyttötapaukset ovat olleet se edellinen tapa määritellä, siksi niitä varmaan on enemmän, mutta kyllä käyttäjätarinoita aika yllättävästi jo näkee.” – Haastateltava H2

”Meidän yksi käyttäjätapauskuvauksia saattaa koostua useammasta Epicistä” – Haastateltava H7

”Story on hyvä määrittely, et tällainen pitäisi saada syntymään” – Haastateltava H8

Tiimit käyttävät erilaisia ohjelmistovaatimustekniikoita vaatimusten löytämiseksi. Osalla tiimeistä on käytössä jo käyttäjätarinat, joissa ohjelmistovaatimukset muodostetaan parilla lauseella käyttäjän kielellä. (Turetken ym., 2016). Tämä muodostaa empiirisen johtopäätöksen EC10.

EC10: Erilaisten ohjelmistovaatimustekniikoiden käyttö tiimeillä on yleistä. (L)

4.2.5 Säännöllinen reflektointi ja sopeutuminen

Tiimit refleктоivat ja mukauttavat omaa toimintaansa säännöllisesti kullakin iteraatio- ja julkaisutasolla. Reflektoinnissa hyödynnetään kvantitatiivisia arviointimenetelmiä. (Turetken ym., 2016) Oman toiminnan reflektointia sekä sprintti- että julkaisukohtaisesti tapahtuu kolmen haastateltajan perusteella.

”Meillä on joka sprintissä, siinä sprintin lopussa on retrot ja on ollut nykyisessä tiimissä alusta asti.” – Haastateltava H1

”Tiimeillä on retroja, mutta toisaalta tehdään erinäisiä asiakastytyväisyyskyselyitä.” – Haastateltava H2

”Me ollaan kehitetty semmoinen oma asiakaspalautekysely missä me saadaan oikeaa palautetta ja sillä perusteella me aina tehdään parantavia toimenpiteitä.” – Haastateltava H7

Tiimeissä toiminnan reflektointia tapahtuu, mutta arvioinnissa kvantitatiivisten menetelmien käyttö on vaihtelevaa.

”Ei ihan kauhean laajasti meidän tiimissä ole noi Jira-metriikat käytössä.” – Haastateltava H1

”Kun ollaan suunnittelujakson suunnittelut tehty, ne on kaikki sprinteillä meidän tekemiset niin mä laitan niille versioksi sen suunnittelujakson. Sit pystyn siellä dashboardilla näyttämään, että milloin me ollaan ehkä valmiita” – Haastateltava H7

”Tehdään erillinen pöytäkirja, että tilanne oli näin kuin sprintti päättyi. Kuvakaappausraportti siihen, että mitkä valmistui, mitkä jäi kesken ja mitä keskusteltiin siinä katselmoinnissa ja piirakkakuvioita, kuinka paljon tuli uusia tehtäviä sprintin aikana” – Haastateltava H8

Tiimit refleктоivat omaa toimintaansa retrospektiiveissä, mutta kvantitatiivisten menetelmien käyttäminen esimerkiksi sprinttiraporttien tai julkaisuraporttien muodossa on vaihtelevasti käytössä. Tämä muodostaa empiirisen johtopäätöksen EC11.

EC11: Tiimit refleктоivat ja mukauttavat omaa toimintaansa säännöllisesti, mutta reflektionissa ei käytetä kvantitatiivisia arviointimenetelmiä kovinkaan laajasti. (L)

4.2.6 Asiakaslähtöiset iteraatiot

Asiakaslähtöisillä iteraatioilla tarkoitetaan sitä, että asiakas hallitsee ominaisuuksien kehittämistä iteraatiokohtaisesti. Asiakas ohjaa projektin kehittymistä pyytämällä sellaisten ominaisuuksien valmistamista, jotka tuottavat eniten arvoa liiketoiminta näkökulmasta katsottuna. (Turetken ym., 2016) Tämän käytänteen arvioinnissa tulisi huomata, että organisaatiossa yhdellä tiimillä voi olla useita asiakkaita, joita heidän tulisi palvella. Yksi haastateltavista nostaa esille, että hänelle ei ole selkeää, kuka asiakas edes on.

”Nimenomaan, en mä ainakaan osais kysyä tai sanoa, että keneltä lähdetään kysyä, kokeeko asiakas meidän toiminnan hyväksi tai ollaanko me onnistuttu asiakkaan näkökannasta.” – Haastateltava H1

Yksi tiimi on kehittänyt kuitenkin toimintamallin, jolla varmistetaan se, että vaikka tiimin täytyy palvella useita eri asiakkaita, kyseinen tiimi keskittyy eniten liiketoiminta-arvoa tuottaviin asioihin. Tässä tiimissä tiimin tuoteomistaja priorisoi ikään kuin asiakkaiden eli tuotteiden tuoteomistajien tekemiset eniten asiakasarvoa tuottavaan järjestykseen.

”Meillä on suunnittelujaksojen jokaiselle suunnittelujaksolle tehty taulukko ja siinä on se tavoite, mitä se tuoteomistaja sille tuotteillaan tämän suunnittelujakson aikana haluaa. Siellä on perustelut esimerkiksi, että kuinka monta käyttäjää tästä hyötyy ja mitä riskejä tämän tekemisessä ja tekemättä jättämisessä on ja niiden kohtien perusteella meidän tiimin tuoteomistaja tutkii näiden kaikkien tuotteiden tuoteomistajan tekemät jutut ja arvottaa ne prioriteettijärjestykseen.” – Haastateltava H7

Riippuen ajattelumallista, ketä asiakkaalla tässä kohtaa tarkoitetaan, asiakas hallitsee joko täysin tai ei ollenkaan ominaisuuksien kehittämistä. Jos asiakkaalla tarkoitetaan tilaavaa tahoja, eli tuotteen tuoteomistajaa, hän ei saa välttämättä

palvelua ollenkaan, saati ohjaa kehittämistä sprinteittäin. Toisaalta jos asiakkaalla tarkoitetaan tiimin tuoteomistajaa, joka priorisoi tilaavien tahojen toiveet keskenään, hän nimenomaan hallitsee ominaisuuksien kehittämistä sprinttikohtaisesti. Tässä kohtaa haastateltava on tiiminsä kanssa määritellyt, että nimenomaan tuotteiden tuoteomistajat ovat heidän asiakkaansa, joten sen perusteella asiakas ei ohjaa ominaisuuksien kehittämistä iteraatiokohtaisesti.

Havaintoja siitä, että asiakas ohjaisi ominaisuuksien priorisointia iteraatiokohtaisesti, ei ole. Tämä muodostaa johtopäätöksen EC12.

EC12: Asiakaslähtöisiä iteraatioita, joissa asiakas ohjaa ominaisuuksien priorisointia liiketoiminta-arvon maksimoimiseksi, ei ole käytössä. (N)

4.2.7 Jatkuva asiakastytyväisyyspalaute

Jatkuvalla asiakaspalautteella varmistetaan asiakastytyväisyys. Jatkuva asiakastytyväisyyspalaute mahdollistaa sen, että projektia kehitetään kohti haluttua lopputulosta sen sijaan, että projektin lopussa huomataan, että itseasiassa se mitä kehitettiin ei ollutkaan sitä mitä asiakas tarvitsi. (Turetken ym., 2016) Usean haastateltavan vastauksessa nostettiin esille se, että asiakastytyväisyyspalautetta hyödynnetään tuotteita kehittäessä.

”Ja esimerkiksi nythän siinä [KEHITYSTIIMIN NIMI], niin siitähän tuotettiin pilotti eli se pääsi itse asiassa oikeat käyttäjät pilotoimaan sitä ratkaisua. Ja sitä kautta kerättiin palautetta” – Haastateltava H3

”Kyllähän asiakastytyväisyyttä mitataan ja raportoidaan ja siitä ollaan kiinnostuneita Kela-tasolla – – Varmaan jotain suuntausta tähän on olemassa, et kysytään palautetta saman tien.” – Haastateltava H2

”Kun meillä rupesi olemaan tämmöinen ihan ensimmäinen miniversio valmiina se oli meillä kolmisen viikkoa tuotantokäytössä, jolloin me kerättiin palautetta käyttäjiltä.” – Haastateltava H6

Kaksi haastateltavista nosti erikseen esille sen, että asiakastytyväisyyspalautteen kysyminen on jatkuvaa tai ainakin jatkossa tulee olemaan.

”Me ollaan kehitetty semmoinen oma asiakaspalautekysely missä me saadaan sitä oikeaa palautetta ja sillä perusteella me aina tehdään parantavia toimenpiteitä.” – Haastateltava H7

”Et sieltä tullaan sit jatkossa keräämään varmastikin huomioimaan sitä käyttäjäpalautetta.” – Haastateltava H6

Haastateltavien vastausten perusteella jatkuvan asiakaspalautteen kysyminen on tällä hetkellä yksittäistä eikä jatkuvaa. Jatkuvaa asiakastytyväisyyspalautteen kysymistä tehdään harvakseltaan. Tämä muodostaa empiirisen johtopäätöksen EC13.

EC13: Jatkuva asiakastyytyväisyyspalautteen kysyminen on harvinaista (P)

4.2.8 Leanin vaatimukset skaalautuvassa mittakaavassa

Isoissa organisaatioissa on vaikeaa saada koko tiimi työskentelemään kohti yhteistä tavoitetta. Tätä varten on luotava skaalautuva vaatimusmalli, joka koostuu visiosta, tiekartasta ja oikea-aikaisesta työstämisestä, joka varmistaa ketteryyden hyödyt laajemmille skaalautuville tiimeille. (Turetken ym., 2016) Kysyttäessä, onko organisaatiossa skaalautuvassa mittakaavassa ketteryyttä, haastateltava vastasi:

”No ei ei olla päästy siihen ja sehän vaatisi ihan organisaatiomuutoksia.” – Haastateltava H7

Organisaatiossa on olemassa ketteriä tiimejä, mutta julkaisujunien käyttö on vielä harvinaista. Olemassa olevassa julkaisujunassa on käytössä julkaisun suunnittelut, tavoitteet asetetaan seuraavalle julkaisulle, tarkemmin tavoitteet ja yhteinen suunta käydään Portfolio Epicien kautta läpi:

”Me kerrotaan joka julkaisun aluksi meidän tavoitteet kolmelle kuukaudelle. Ja meillä on demotilaisuus junassa jaettu niin, että -- päätuoteomistajat kertoo, mitkä meidän tavoitteista täyttyi, mitkä ei, missä tilanteissa me ollaan minkäkin Portfolio Epicin kanssa.” – Haastateltava H6

Skaalautuvaa ketteryyttä ei ole vielä laajasti tiimeillä käytössä, sillä suurin osa tiimeistä ei ole organisoitunut arvovirran ympärille. Olemassa olevassa julkaisujunassa täyttää Leanin vaatimukset skaalautuvassa mittakaavassa, eli tiimit on saatu työskentelemään kohti yhteistä tavoitetta. Tämä muodostaa empiirisen johtopäätöksen EC14.

EC14: Yksi organisaation julkaisujuna on saavuttanut skaalautuvan ketterän mallin, jossa tiimit työskentelevät kohti yhteistä tavoitetta. Organisaatiossa skaalautuva ketteruus ei ole muuten käytössä. (P)

4.2.9 Matala prosessiseremonia

Matalampi prosessiseremonia mahdollistaa reagoinnin muutoksiin. Matalalla prosessiseremonialla tarkoitetaan sitä, että muutoksia ei tarvitse hyväksyä vähintään kolmelta johtotasolta ja prosessiin liittyvä paperityön taso ei ole kohtuuton (Turetken ym., 2016). Kysyttäessä, kuinka hyvin tiimi pystyy itse päättämään asioistaan, haastateltavat vastasivat:

”Aiheesta riippuen, joissakin aiheissa tiimi pystyy hyvinkin itsenäisesti, ja mä olen oikeastaan vähän hämmästynyt siitä, että miten itsenäisesti saadaan nykyään tehdä asioita ja päättää.” – Haastateltava H6

”Meidän yksikössä menee tosi hienosti, aika paljon ollaan kumpaakin tiimiä muutettu, niin on täysi vapaus kehittää toimintatapoja” – Haastateltava H8

Yksi haastateltavista nosti esille, että tiimillä olisi mahdollisuus päättää itse muutoksista, mutta kaikki tiimit eivät sitä haluaisi.

”Varmaan jossain tiimit haluaisivat päättää paljon enemmänkin. Ja toisaalta jossain taas haluttaisiin nimenomaan, että tulisi asiat keskitetympään, riippuen asiasta ja riippuen tiimistä.” – Haastateltava H3

Toisen haastateltavan vastaus kuvastaa puolestaan sitä, että tiimillä tai tuoteomistajalla ei ole itsellä aina mahdollisuutta päättää asioista.

”Tuoteomistajien pitäisi pystyä päättää asioita, ja sitten just toi ideologinen asia, et pitäisi päätöksentekoa viedä alemmaksi niin ei se kyllä ole kauhean pitkällä se asia.” – Haastateltava H2

Yhdenkään haastateltavan vastauksesta ei käy ilmi, kuinka monen portaan päässä päätökset lopulta tehdään, jos tiimi ei saa itsenäisesti hyväksyä muutoksia. Vastausten perusteella organisaatiossa on suhteellisen matala prosessiseremonia ja tämä on saavutettu ainakin osassa tiimejä. Tämä muodostaa empiirisen johtopäätöksen EC15.

EC15: Tiimit saavat itse päättää asioista melko vapaasti ja päätöksentekoon liittyvä prosessiseremonia on matala, mutta paikoittain päätöksiä on hyväksyttävä ylempänä organisaatiota. (P)

4.2.10 Jatkuva SAFe kyvykkyyksien parantaminen

Turetken ym. (2016) kypsyytastomallissa ei ole erikseen määritelty, mitä jatkuva SAFe kyvykkyyksien parantaminen periaatteena tarkoittaa. Määritelmän puuttuessa on vaikea varmistua siitä, mitä asiaa haastateltavilta olisi tarkoitus varmistaa. Eli sen takia tämä käytäntö jätetään kokonaan arvioimatta.

Yhteenvedona asiakasarvo periaatteeseen löytyi yhteensä 12 empiiristä yhteenvedoa. Näiden yhteenvedojen pohjalta muodostettiin ensisijainen empiirinen johtopäätös PEC1.

PEC2: Asiakaspalautteen kautta toiminnan kehittäminen ja asiakasarvon tuottaminen eivät ohjaa ohjelmistokehitystyötä.

Asiakasarvon tuottamisen empiiriset johtopäätökset ovat esiteltyinä seuraavassa taulukossa (TAULUKKO 7).

TAULUKKO 7 Asiakasarvon tuottamisen empiirisen johtopäätökset

Tunniste	Empiirinen johtopäätös
----------	------------------------

EC4	Lakimuutokset saattavat toimia järjestelmien kehittämisen ajureina, jolloin järjestelmän kehittäjätiimit eivät pysty vaikuttamaan siihen, tuottaako tarvittava muutos asiakasarvoa.
EC5	Organisaatiolla on puutteita oikeantasoisten ja -suuruisten tavoitteiden asettamisessa, mikä saattaa liittyä organisaatiokulttuuriin.
EC6	Käyttäjän ottaminen osaksi kehitystiimiä mahdollistaa asiakaslähtöisen kehittämisen.
EC7	Prosessin reflektointia ja parantamista tapahtuu, mutta ei kaikilla organisaation osa-alueilla eikä täysin systemaattisesti. (L)
EC8	Vaatimusmäärittely on aiempaan verrattuna evolutiivisempaa, mutta vaatimusten kehittäminen asiakaspalautteen pohjalta on haasteellista toimintaympäristön vuoksi. (P)
EC9	Pienemmät, useammat julkaisut ovat melko laajasti tiimeillä käytössä, mutta osittain julkaisut ovat vielä isoja, mitä viedään tuotantoon harvoin. (L)
EC10	Erilaisten ohjelmistovaatimustekniikoiden käyttö tiimeillä on yleistä. (L)
EC11	Tiimit refleктоivat ja mukauttavat omaa toimintaansa säännöllisesti, mutta reflektoinnissa ei käytetä kvantitatiivisia arviointimenetelmiä kovinkaan laajasti. (L)
EC12	Asiakaslähtöisiä iteraatioita, joissa asiakas ohjaa ominaisuuksien priorisointia liiketoiminta-arvon maksimoimiseksi, ei ole käytössä. (N)
EC13	Jatkuva asiakastyytyväisyyspalautteen kysyminen on harvinaista (P)
EC14	Yksi organisaation julkaisujuna on saavuttanut skaalautuvan ketterän mallin, jossa tiimit työskentelevät kohti yhteistä tavoitetta. Organisaatiossa skaalautuva ketteruus ei ole muuten käytössä. (P)
EC15	Tiimit saavat itse päättää asioista melko vapaasti ja päätöksentekoon liittyvä prosessiseremonia on matala, mutta paikoittain päätöksiä on hyväksyttävä ylempänä organisaatiota. (P)
PEC2	Asiakaspalautteen kautta toiminnan kehittäminen ja asiakasarvon tuottaminen eivät ohjaa ohjelmistokehitystyötä.

4.3 Suunnittelu ja jatkuva tuottaminen

Suunnittelu ja jatkuva tuottaminen on kypsyytastason toinen periaate. Tähän alalukuun on koottuna suunnittelun ja jatkuvan tuottamisen periaatteeseen liittyviä havaintoja, ja havainnot ovat jaoteltuna tarkempiin arvioitaviin käytänteisiin.

4.3.1 Yhteissuunnittelu

Yhteissuunnittelussa kaikki tarvittavat sidosryhmät kokoontuvat yhteen suunnitteluvaiheen aikana. Yhteissuunnittelu lisää näkyvyyttä sekä kasvattaa uskollisuutta ja hyväksymistä suunnitelmaa kohtaan. (Turetken ym., 2016) Kysyttäessä onko tiimeillä käytössä yhteissuunnittelu, haastateltava vastasivat:

”Siinä todetaan ylätason tavoitteet seuraavalle kolmelle kuukaudelle” – Haastateltava H8

”Meillähän oli just tällä viikolla puoltoista päivää yhteiset suunnittelupäivät.” – Haastateltava H1

”Kyllähän aika moni, aika monet periaatteessa on mukana yhteissuunnitteluissa” – Haastateltava H2

”Meillä osallistuu itse suunnitteluun -- tiimin tuoteomistaja -- koko tiimi -- kaikkien meidän tuotteiden tuoteomistajat -- taitaa linjaesihenkilöt olla.” – Haastateltava H7

”Tiimi siinä keskiössä on, mutta ulkopuolisia suunnittelijoita eli liiketoiminnan edustajia on siinä ja siellä on käynyt päätuoteomistaja, joka ei kuulu tiimiin” – Haastateltava H8

Eli yhteissuunnitteluun osallistuu tarvittavat sidosryhmät ja joissain tapauksissa vielä ylimääräisiä henkilöitä tai tiimejä. Haastateltavien vastausten perusteella yhteissuunnittelut ovat laajasti käytössä. Tämä muodostaa empiirisen johtopäätöksen EC16.

EC16: Yhteissuunnittelut ovat tiimeillä laajasti käytössä. (F)

4.3.2 Jatkuva tuottaminen

Jatkuvassa tuottamisessa ohjelmistoa tuotetaan pienissä iteraatioissa säännöllisellä rytmillä. Säännöllinen rytmi varmistaa sen, että tuotteen kehittäminen on jaettava jokaiselle iteraatiolle. (Turetken ym., 2016) Alaluvussa 4.3.3 Pienemmät, useammat julkaisut tehtiin johtopäätös, että pienemmät, useammat julkaisut ovat melko laajasti tiimeillä käytössä, mutta osittain julkaisut ovat vielä isoja, mitä viedään tuotantoon harvoin. Samaa johtopäätöstä ja siihen johtaneita vastauksia hyödynnetään myös tämän käytännön arvioinnissa. Erona aikaisempaan käytäntöön, tässä käytännössä tarkoituksena on arvioida nimenomaan sitä, tuotetaanko ohjelmistoa säännöllisellä rytmillä. Datan perusteella kaksi tiimiä tuottaa säännöllisellä rytmillä ohjelmistoa.

”Oliko se jo yli 20 julkaisua vai melkein keskimäärin joka toinen viikko jotain vietiin tuotantoon” – Haastateltava H8

”Kyllä, joka viikoittainhan on tämmöiset tomut eli torstaisin viedään muutoksia tuotantoon. Tässä sprintin aikana muutokset etenee, niillä tähdätään johonkin tomuun, että sen sprintin aikana saataisiin asiat vietyä tuotantoon.” – Haastateltava H5

Jatkovaa tuottamista tapahtuu joissain tiimeissä, mutta organisaatiossa säännöllisellä rytmillä tuottaminen ei ole laajasti käytössä. Tämä muodostaa empiirisen johtopäätöksen EC17.

EC17: Jatkuvaa tuottamista tapahtuu joissain tiimeissä, mutta organisaatiossa säännöllisellä rytmillä tuottaminen ei ole laajasti käytössä. (P)

4.3.3 Kahden tason suunnittelu ja seuraaminen

Työtä tulisi suunnitella lyhyemmällä sekä pidemmällä aikavälillä. Lyhyen aikavälin suunnittelulla tarkoitetaan iteraatioiden suunnittelua, jossa lyhyen tähtäimen painopisteet suunnitellaan tarkasti. Pidemmällä aikavälin suunnittelulla tarkoitetaan julkaisusuunnittelua, joka keskittyy karkeampaan ja vähemmän kattavaan suunnitteluun. (Turetken ym., 2016) Haastateltavien vastausten perusteella kahden taso suunnittelua ja seuraamista tehdään.

”Meillä on ihan Scrum-mallin mukaisesti reviewt, retrot ja sprint planning.” - Haastateltava H5

”Kehitysjaksotasolla suunnittelevat, mitä tällä kehitysjaksolla pitäisi saada tehtyä, joita viedään viikoittain tuotantoon.” - Haastateltava H5

”Meillähän oli just tällä viikolla puoltoista päivää yhteiset suunnittelupäivät.” - Haastateltava H1

”Suunnittelupäivänä me suunnitellaan ne sprintit alustavasti” - Haastateltava H7

Kysyttäessä tarkemmin, mikä on se aikaväli, kun tiimissä tehdään pidemmän aikavälin suunnittelua, haastateltava tarkensi:

”Ne on ollut kvartaaleittain. Eli keskimäärin neljä sprinttiä, eli kolme viikon sprinttiä ollut se scope.” - Haastateltava H1

Sekä lyhyen että pidemmän aikavälin suunnittelu on tiimeillä käytössä. Pidemmän aikavälin suunnittelussa työn voi olla osalla tiimeistä puutteellista.

”Oli suunniteltu ensimmäinen [sprintti], mutta sitten siellä oli ihan muutama hassu tehtävä vaan niissä seuraavissa sprinteissä.” - Haastateltava H2

Haastateltavien vastausten perusteella kahden tason suunnittelua ja seuraamista tehdään monissa tiimeissä, mutta osassa tiimeistä pidemmän aikavälin suunnittelut ei toimi. Tämä muodostaa empiirisen johtopäätöksen EC18.

EC18: Tiimeissä tehdään kahden tason suunnittelua ja seuraamista, mutta joillain tiimeillä pidemmän aikavälin suunnittelu voi olla puutteellista. (L)

4.3.4 Ketteryyden arviointi ja kiertonopeus

Työmääriä arvioidaan tarinapisteiden (engl. *story points*) avulla. Tiimi tuottaa tietyn määrän tarinapisteitä iteraatioissa, jotka määrittävät tiimin nopeuden. Tätä kiertonopeutta voidaan käyttää aikataulun arvioinnissa ja

kustannusten arvioinnissa. (Turetken ym., 2016) Kysyttäessä, onko tiimillä story pointit käytössä, kolme haastateltavaa vastasi:

”On story pointit käytössä, kyllä.” – Haastateltava H6

”Joo, meillä on story pointit käytössä.” – Haastateltava H7

”On joo” – Haastateltava H8

Tarkennettaessa mitä story pointeilla tehdään, haastateltavat vastasivat:

”Koitan trendejä haistella säännöllisesti, onko meidän vauhti hiipumassa ja mistä se voisi johtua.” – Haastateltava H1

”Joo, meillä on se arviointi sille, että me kysytään aina sprinttiin suunnittelua varten tiimiläisistä htp-arviot.” – Haastateltava H6

Tarinapisteiden ja henkilötyöpäiväarvioita käytetään tiimin kapasiteetin laskennassa ja toisaalta arvioitaessa tiimin nopeutta. Tiimit arvioivat työmääriään erilaisin menetelmin. Tämä muodostaa empiirisen johtopäätöksen EC19.

EC19: Tiimit arvioivat työmääriään erilaisin menetelmin. (F)

Osa tiimeistä ei hyödynnä työkaluja niin laajasti, kuin olisi mahdollista. Se miksi työkaluja ei käytetä niin laajasti voi selittyä osaamisen puutteella. Tätä tukee haastateltavan vastaus.

”Esimerkiksi siellä ilmestyy noita story pointseja rapsalle vähän kummallisesti. Olen yrittänyt katsoa, mistä se johtuu.” – Haastateltava H1

Ilmaisut ”kummallisesti” ja ”olen yrittänyt katsoa” voisivat viitata siihen, että työkalun ominaisuudet eivät ole täysin tuttuja ja on epävarmaa, aiheuttaako ongelman työkalun rajoitteet vai se, että työkalua ei osata käyttää oikein. Toisaalta kyse voi olla myös siitä, että työkalu ei tue tiimien tarpeita riittävästi:

”Meillä suhteutetaan sitä [TIIMIN KAPASITEETTIA], mulla on oikein Excel graafit, vaikka kyllähän se Jirakin antaa, mut Jira antaa vaan sen kolmen sprintin perusteella sitä ennustetta. Olen itse tehnyt graafin, joka laskee ennustetta vähän pitemmällä aikasyklillä.” – Haastateltava H6

Työvälineet eivät tue tarpeeksi tiimien tarpeita, ja osalla työvälineiden osaamisen käytössä on puutteita. Tämä muodostaa empiirisen johtopäätöksen EC20.

EC20: Työvälineet eivät tue tarpeeksi tiimien tarpeita, ja osalla työvälineiden osaamisen käytössä on puutteita.

4.3.5 Julkaisun suunnittelu

Useat tiimit synkronoivat tekemisensä julkaisun suunnittelussa. Tilaisuus kestää kerrallaan kaksi päivää. Julkaisun suunnittelu pidetään yleensä kasvokkain tapahtuvana tapahtumana, johon osallistuvat kaikki ohjelman tai julkaisujunan tiimit ja jäsenet. (Turetken ym., 2016) Tämän käytänteen arvioinnissa tulisi huomioida se, että organisaation tiimeistä moni tiimi tekee töitä oman tavoitteensa eteen ja selkeitä riippuvuuksia toisiin tiimeihin ei välttämättä ole.

”Aika monellakin tiimillä on ideaalitilanne, että meillä ei ole mainittavia riippuvuuksia, jolloin me voidaan itse järjestellä töitä, niin kuin parhaaksi nähdään” – Haastateltava H8

”Meillä on niitä [RIIPPUVUUKSIA] ollut itse asiassa viime vuosina tosi vähän enää. Niitä oli ehkä silloin alussa enemmän ja nyt vähemmän, että ei ole Jiraan viety missään vaiheessa.” – Haastateltava H6

Tiimeillä saattaa olla joitakin tilanteita, joissa eri tiimien on synkronoitavansa tekemisensä muiden tiimien kanssa. Tällöin yhteensovittamisesta sovitaan käyttöönottosuunnitelmien kautta.

”Isommassa tuotantoon viennissä missä on kysymys useista tiimeistä -- me kehitettiin julkaisu, käyttöönottosuunnitelma. Me käytetään isommissa julkaisussa semmoista.” – Haastateltava H7

Julkaisuja ei kuitenkaan suunnitella erillisissä yhteissuunnittelupäivissä eri tiimien kanssa. Organisaatiossa puhutaan suunnittelupäivistä, joita järjestetään yleensä neljä kertaa vuodessa. Useampi haastateltavaa nosti esille, että yhteissuunnitteluja järjestetään, mutta siellä ei joko käsitellä yhteisiä asioita tai tiimit, jotka ovat koottuna yhteen, eivät tee keskenään yhteistyötä.

”Se on nimellinen se suunnittelupäivä ja kokoonnutaan kyllä sinne, mutta just se, että siellä tehtäisiin toisten tiimien kanssa yhteensovittamista jostain hommista, niin sitä ei välttämättä tapahdukaan.” – Haastateltava H2

”-- onko niillä tiimeillä toistensa kanssa tekemistä, niin ei välttämättä. – Haastateltava H3

”Siinä [SUUNNITTELUPÄIVÄSSÄ] ei kylläkään käyty tavallaan tiimien kesken yhteistä dialogia juurikaan. – Haastateltava H1

Se, että tiimit kootaan yhteen, mutta he eivät yhdessä synkronoi tekemistään voi ainakin osittain selittyä se, että suurin osa tiimeistä ei kuulu julkaisujunaan ja tiimeillä ei ole välttämättä riippuvuuksia muihin tiimeihin. Haastateltavien vastauksista voisi päätellä sen, että yhteissuunnitteluun eivät osallistu oikeat sidosryhmät, jos näillä tiimeillä ei ole keskenään mitään keskusteltavaa. Tätä tulkintaa tukee haastateltavan vastaus siitä, yhteissuunnittelutilaisuuteen on aiemmin koottu yhteen tiimejä, jotka tekevät eri päämäärän eteen töitä.

” -- se asia oli ymmärretty väärin, että ei tavallaan ymmärretty, että meillä pitäisi olla jotain yhteyttä, releasoitavaa” - Haastateltava H7

Yksi haastateltavista ihmettelee, miten muualla huolehditaan riippuvuuksista, jos julkaisujunaa ei ole käytössä:

”Tavalla tai toisella niistä [riippuvuuksista] pitäisi huolehtia. En tiedä miten Kelassa tavallaan niillä osa-alueilla, joissa ei ole tämmöistä junaa, että on vaan tiimejä, niin mikä rooli siellä huolehtii riippuvuuksista ja niiden läpinäkyvyydestä.” - Haastateltava H4

Datan perusteella julkaisujen suunnittelut, joissa useat tiimit synkronoivat tekemisensä keskenään ei tapahdu muualla kuin julkaisujunassa. Yhteissuunnittelutilaisuuksiin osallistuu julkaisun suunnittelun kannalta väärät tiimit. Tämän vuoksi julkaisun suunnittelussa ei synkronoida tekemisiä muiden tiimien kanssa ja hallita riippuvuuksia. Epäselväksi jää, miksi tiimejä kootaan yhteissuunnitteluun yhteen, jos heillä ei ole toistensa kanssa tekemistä.

Yhteissuunnittelua toisten tiimien kanssa ei välttämättä tarvita, jos tiimillä ei ole tai on hyvin vähän riippuvuuksia toisiin tiimeihin. Tarve hallita riippuvuuksia etukäteen ei välttämättä ole, vaan riippuvuuksista aiheutuneita ongelmia ratkotaan sitä mukaan, kun niitä tulee vastaan. Toisaalta työvälineet mahdollistavat läpinäkyvyyden rakentamisen ja riippuvuuksien hallinnan. Joten julkaisun suunnittelua ei ole välttämätöntä pitää fyysisesti samassa paikassa ja toisten tiimien kanssa töiden synkronointi voi tapahtua työvälineessä. Tämä nousee esille yhden haastateltavan vastauksessa:

”Meillä on Jirassa sellanen Easy Agile -työkalu käytössä, missä pääsee näkemään niitä riippuvuuksia ja merkitsemään niitä riippuvuuksia toisten tiimien tehtäviin, Epiceihin jne. Sanoisin, että se on vielä vähän alkuvaiheessa sen hyödyntäminen.” - Haastateltava H1

Julkaisu suunnittelu on mahdollista tehdä etänä, mutta työkaluja ei hyödynnetä systemaattisesti. Julkaisun suunnitteluja pidetään, mutta yhteissuunnitteluun on koottuna väärä tiimejä ja tilaisuudessa ei hallita riippuvuuksia toisten tiimien kanssa. Tämä muodostaa empiirisen johtopäätöksen EC21.

EC21: Julkaisun suunnitteluja pidetään, mutta tiimit eivät julkaisun näkökulmasta koordinoi tekemisiään oikeiden tiimien kanssa ja tilaisuudessa ei hallita riippuvuuksia toisiin tiimeihin. (P)

4.3.6 Riskilähtöiset iteraatiot

Riskilähtöisillä iteraatioilla pyritään hallitsemaan riskitekijöitä mahdollisimman varhaisessa vaiheessa. Riskien vähentäminen varhaisessa vaiheessa varmistaa sen, että tiimi ei vietä liikaa aikaa sellaisen järjestelmän rakentamiseen, jonka suorittamista he eivät voi suorittaa loppuun saakka. (Turetken ym., 2016)

Kysyttäessä suunnitellaanko sprinttejä tai julkaisuja riskinäkökulmasta, haastateltavat vastasivat:

”En ole kuullut kyllä.” – Haastateltava H2

”En muista kuulleen kertaakaan sanaa riski.” – Haastateltava H3

”Kummallakin tiimillä me ylläpidetään ihan semmoista virallista riskimatriisia. Pääasiassa sitä tarkistellaan suunnittelujakson suunnittelupäivillä, mutta kyllä niitä vilkuillaan esimerkiksi ihan joka sprintin vaihteessa vähintään Scrum Masterin toimesta” – Haastateltava H8

”Me ruvettiin viime vuoden loppupuolella näitä riskejä ihan sillä Jiran riski työkalulla kirjaamaan. Sitä ei ole ääneen sanottu, mutta minä miellän, että ne ovat julkaisukohtaisia ne riskit. -- Tuoteomistajien ja Scrum Mastereiden kesken me käydään ne riskit tarkemmin aina lävitse. Ne tulee kolmen viikon välein tsekattua.” – Haastateltava H6

Tarkemmin kysyttäessä, kirjataanko riskit pelkästään muistiin vai suunnitellaanko siihen samalla keinoja, miten riskejä voisi poistaa tai lieventää sprinteittäin, haastateltava vastasi:

”Se olis se seuraava askel meillä varmaan.” – Haastateltava H6

Haastateltavien vastausten perusteella riskejä tunnistetaan joissain tiimeissä julkaisu- ja sprinttikohtaisesti. Riskilähtöiset iteraatiot, jossa sprintti suunnitellaan riskien minimoimiseksi ei tapahdu lähes ollenkaan. Tämä muodostaa empiirisen johtopäätöksen EC22.

EC22: Riskilähtöiset iteraatiot, jossa sprintti suunnitellaan riskien minimoimiseksi ei tapahdu lähes ollenkaan. (N)

4.3.7 Ominaisuuksien, eikä tehtävien suunnittelu

Asiakkaan tarpeet ilmaistaan ominaisuuksina. Jos ominaisuus muuttuu, vaikutus siihen liittyviin tehtäviin minimoidaan. Suunnittelu tulee tehdä ominaisuuksien suhteen, jotta voidaan valmistautua tasolla 4 olevaan asiakaslähtöiseen kehitysprosessiin. (Turetken ym., 2016) Ainakin kolme haastateltavaa tuo esille, että heillä suunnitellaan ominaisuuksia eikä tehtäviä. Yksi haastateltavista mainitsee, että heillä haasteena ei ole välttämättä ominaisuuksien muuttuminen vaan ennemminkin kehitettävien ominaisuuksien määrä.

”Meillä valmistuu tällä hetkellä 30% suunnitelluista ominaisuuksista” – Haastateltava H4

Kysyttäessä, onko tiimeillä ominaisuudet eli Jirassa Epicit käytössä, haastateltavat vastasivat:

"Epic on se semmoinen käyttötapaus -kuvaustyyppinen" - Haastateltava H7

"Kantassa [KANTA-PALVELUISSA] tavoiteasetanta tehdään Epicien kautta." - Haastateltava H2

"Joo, on." - Haastateltava H6

Kysyttäessä tarkennusta siihen, miten Epicejä käytetään, haastattelija tarkensi:

"Epic toimii meillä myös vähän hakemistoina, suodattimina siellä. -- ne on ollut ahkerassa käytössä, mutta ne ei ehkä ole semmoisia niin kuin niitten alun perin kuuluis tässä mallissa olla." - Haastateltava H6

Tästä voisi tulkita sen, että tässä tiimissä asiakkaan tarpeiden ilmaisut tehdään tehtävien ei ominaisuuksien muodossa ja Epicien käyttö Jirassa ei noudata ominaisuuksien suunnittelua. Myös toisen haastateltavan vastaus tukee tätä tulkintaa.

"Kun päättyy se suunnittelujakso, me ihan keinotekoisesti heivataan sieltä kaikki keskeneräiset seuraavaan Epiciin, suljetaan se ja todetaan, että suunnittelujakso kahdessa tämmöinen Epic valmistui sisältäen nämä itemit." - Haastateltava H8

Asiakkaan tarpeita suunnitellaan sekä ominaisuuksien että tehtävien muodossa. Se, kumpaa tapaa tiimissä käytetään voisi liittyä esimerkiksi siihen, kuinka paljon ylläpitoa ja kuinka paljon kehittämistä tiimissä tehdään, sillä haastateltava H8 mainitsee, että heillä on paljon jatkuvaa ylläpitotyötä, jota ei voi etukäteen suunnitella. Tämä muodostaa empiirisen johtopäätöksen EC23.

EC23: Asiakkaan tarpeita ilmaistaan sekä ominaisuuksien että tehtävien muodossa. (P)

4.3.8 Tiekartta

Tiekartan avulla kaikki julkaisujunan tiimit pystytään synkronoimaan keskenään. Tiekartta tuo ennustettavuutta tuotoksille. (Turetken ym., 2016) Kysyttäessä, onko tiimeillä tiekarttaa käytössä, haastateltavat vastasivat:

"On." - Haastateltava H6

"En ole kertaakaan nähnyt mitään roadmappia tästä." - Haastateltava H5

"Me saatiin myös ratkaisuarkkitehti, joka oli meidän teknisen arkkitehtuuriin keskittyvä ihminen ja teknisestä velasta huolissaan ja rakensi meille roadmapit siihen ja toi sen asian näkyväksi ja kävi vahvaa keskustelua päätuoteomistajien kanssa." - Haastateltava H4

"Yritetään luoda sitä isompaa kuvaa siitä, että mikä vaikka se tuotealueen roadmap on" - Haastateltava H2

”Periaatteessa kyllä. Kelassa on järjestelmiä, jotka on tehty 20 vuotta sitten, niillä on nyt näkyvissä sellaista roadmappia, että niitä pitää ruveta uudistamaan” - Haastateltava H8

Vastauksissa on huomioitava se, että suurin osa tiimeistä ei ole organisoitunut arvovirran ympärille, joten tiekartan tarkoitus on toimia kommunikointivälineenä esimerkiksi kehitystiimin ja tuoteomistajan välillä, eikä niinkään julkaisujunan tiimien koordinoitua varten. Tiimillä saattaa olla myös kaksi tiekarttaa, jolloin yhdessä tiekartassa on esitetty tuotteen kehitettävät asiat ja toisessa tiekartassa esimerkiksi versioiden nostot ja muut tekniset asiat, joilla pyritään pääsemään teknisestä velasta eroon:

”Kehitysvastaava esittelee näitä isoja roadmappeja ja sitten palvelupäällikkö esittää, että tällaisia teknisiä asioita pitäisi huomioida ja tehdä ja kun niitä ei ole priorisoitu, se aiheuttaa hämmingin sinne tiimitasolle, että mitäs me nyt näistä tehdään.” - Haastateltava H2

Tästä voisi tehdä sen tulkinnan, että kaikki tiimit eivät saa yhtä priorisoitua tiekarttaa, jolloin näkyvyys siihen, mitä on tarkoitus saada realistisesti aikaiseksi ei toteudu ja täten tiekartan hyöty eli tuotosten ennustettavuus jää saamatta. Yksi tiimi on ratkaissut asian nojaamalla vahvasti tilaaja-toimittaja -malliin:

”Me tehdään vuosisuunnitelmataulukkoon kolmen kuukauden suunnittelujaksoja ja sinne tuoteomistajat käy etukäteen laittamassa toiveita mitä ne haluaa, niin tavallaan me nähdään sieltä mitä on tulossa. Jos joku haluaa vaikka käyttöliittymä uudistuksen ensi joulukuussa, niin me voidaan jo nyt miettiä, että mitä esimerkiksi meidän täytyisi opiskella sitä varten, että me pystytään tekemään.” - Haastateltava H7

Organisaatiossa yhden tiimin vastuulla voi olla useita tuotteita ja sitä kautta useita tuoteomistajia, joita yhden kehitystiimin pitää palvella. Koska rajatuilla kehittäjäresursseilla ei ole mahdollista saada kaikkea valmiiksi, tiimi on ratkaissut ongelman siten, että tilaaja ei luo epärealistista visiota tuotteesta vaan tiekartta tehdään yhdessä toimittajan kanssa, jolloin molemmat osapuolet tietävät, mitä odottaa. Haastateltavien vastausten perusteella joillain tiimeillä tiekartat ovat hyvin käytössä, mutta kaikilla tiimeillä tiekartta ei tuo ennustettavuutta, koska tiekartta on luotu vain liiketoiminnallisesta näkökulmasta katsottuna. Tämä muodostaa empiirisen johtopäätöksen EC24.

EC24: Tiekarttoja käytetään organisaatiossa melko laajasti. Kaikilla tiekarttoissa ei ole yhteensovitettu liiketoiminnallisia ja teknisiä vaatimuksia. (P)

4.3.9 Iteraation hallitseminen

Ketteryyden ja iteratiivisen kehittämisen perustana on iteraatio, joka tuottaa toimivaa ohjelmistoa. Iteraation tarkoituksena on pystyä luomaan toimiva, testattu ohjelmisto lyhyessä aikataulussa. (Turetken ym., 2016) Yhdessä haastateltavan vastauksessa käy ilmi, että tiimin toimintamalliin kuuluu se, että tavoite on tuottaa toimiva ohjelmamuutos yhden sprintin aikana:

”Kyllä, joka viikoittainhan on tämmöiset tomut eli torstaisin viedään muutoksia tuotantoon. Tässä sprintin aikana muutokset etenee, niillä tähdätään johonkin tomuun, että sen sprintin aikana saataisiin asiat vietyä tuotantoon.” – Haastateltava H5

”Me pyritään aina, että meillä olisi jotain demottavaa. -- eihän se tietenkään aina onnistu” – Haastateltava H7

”Oliko se jo yli 20 julkaisua vai melkein keskimäärin joka toinen viikko jotain vietiin tuotantoon” – Haastateltava H8

Havaintoja iteraation hallitsemisesta ei löytynyt muilta haastateltavilta. Muiden haastateltavien vastaukset tukevat enemmänkin sitä tulkintaa, että iteraatioita ei vielä hallita. Kysyttäessä, saadaanko melkein joka sprintin lopuksi jokin toimiva ohjelmamuutos, joka olisi julkaistavissa oleva lopputuotos, haastateltava vastasi:

”Ei ole semmoista.” – Haastateltava H6

Iteraation hallitsemiseen liittyy työn suunnittelu ja Epicin pilkkomien tarpeeksi pieniksi storyiksi, jotta se mahtuisi tehtäväksi yhden iteraation aikana. Ongelmia vaikuttaisi olevan työn pilkkomisen kanssa.

”Storyt on liian isoja ja samoin tietysti Epicit voi olla. – Haastateltava H2

Iteraation hallitsemisessa saattaisi olla vielä puutteita. Jos sprintille pyritään lisäämään liian isoja kokonaisuuksia, kokonaisuus ei todennäköisesti mahdu yhdelle sprintille ja toisaalta mitään valmista ei myöskään saada. Myös toisen haastateltavan vastaus tukee sitä tulkintaa, että iteraation hallitsemisessa on vielä puutteita.

”Se oli teknistä velkaa, josta me jouduttiin karsimaan taas aikataulusyistä pois, että me saatiin yleensäkin järjestelmä käyttöön, mutta sieltä ei käytetty aikaa niiden yksikkötestien rakentamiseen.” – Haastateltava H6

Tästä vois tulkita, että toimivaa ohjelmistoa julkaistaessa laadunvarmistus saattaa jäädä vajavaiseksi. Vastauksesta ei pysty kuitenkaan päättelemään sitä, kuinka kattavaa testaamista ohjelmistolle tehdään, kun se viedään tuotantoon. Todisteita iteraation hallitsemisesta löytyy kolmen haastateltavan vastauksen perusteella, mutta muuten iteraation hallitsemista ei löydy näyttöä. Tämä muodostaa empiirisen johtopäätöksen EC25.

EC25: Osa tiimeistä pystyy tuottamaan testattua toimivaa ohjelmistoa yhden sprintin aikana. (P)

4.3.10 Kanbanin hyödyntäminen

Kanbanin hyödyntäminen tulisi tapahtua myös liiketoiminta ja arkkitehtuuri Portfolioiden Epiceissä. Kanbania käytetään työnkulun visualisointiin, käynnissä olevan työn rajoittamiseen, virtauksen mittaamiseen ja hallintaan. Kanban-järjestelmässä Epiceillä on neljä jonoa, jonka läpi tehtävä kulkee aloituksesta lopetukseen: ideasuppilo, työjono, analyysi ja toteutus. (Turetken ym., 2016) Teknisesti Kanban-taulut ovat liiketoimintatasolla käytössä, mutta siellä ei noudateta Kanban metodia, jossa samanaikaista työtä rajoitettaisiin WIP (Work in Progress) rajojen käytöllä.

”Kelan kehittämismalli on hiukan samantyyppinen kuin SAFe. Siellä pyörii eri Kanbaneja -- tulee mieleen, että mitä siellä ylemmässä organisaatiossa oikeasti tehdään asian hyväksi? – Haastateltava H8

”Nythän meillä on neljä portfolioa -- meidän organisaatio ei ole muotoutunut sillä tavalla kuin Scaled Agilella ohjeistaa.” – Haastateltava H7

Kahden haastateltavan vastauksessa nostettiin esille Portfolion hallinta ja siihen liittyvä yhtäaikaisen työn rajoittaminen:

”Yleensä kehittämistoimenpiteitä tai Portfolio Epicejä suunnitellaan, et näitä ruvetaan tekemään, niin siinä pitäisi olla ymmärrys siitä onko meillä kaistaa tehdä näitä asioita, ja kuinka paljon meille menee yhtä aikaa. -- se ymmärrys puuttuu, että tämä on liikaa, ja meidän pitäisi jotain karsia.” – Haastateltava H2

”Konseptitoimistojen pitäisi arvioida, mitä me nyt oikeasti tehdään, onko tämä oikeaa asiaa. Sen pitäis katsoa, että voiko niitä laittaa eteenpäin vaikka idea on hyvä, onko sitä nyt mahdollista toteuttaa. Se ei kyl toimi.” – Haastateltava H4

Portfolion hallinnassa samanaikaista työtä ei rajoiteta ainakaan tarpeeksi, jolloin kehittämiskohteiden valmistuminen kestää ja työssä ei ole virtausta. Haastateltavien vastausten perusteella voisi tulkita, että organisaatiolla ei ole kykyä arvioida kapasiteettiaan, eikä kykyä tehdä päätöksiä siitä, mitä jätetään tekemättä.

”Mä en esimerkiksi tiedä, mitä esimerkiksi Kelan portfolioon kuuluu ja miten meidän kuuluisi edes kytkeytyä sinne. -- Helpottaisi tätä omaakin työtä, kun ymmärtäisi sieltä ylhäältä alas saakka” – Haastateltava H4

”Sitä en tiedä miten nuo kehitysportfoliot niitä priorisoivat siellä keskenään” – Haastateltava H7

Läpinäkyvyys portfolioon ja portfolion hallintaan puuttuu. Koska näkyvyys kehitystiimeistä ylöspäin puuttuu, myös näkyvyys ylhäältä alaspäin todennäköisesti puuttuu. Läpinäkyvyyden puuttuminen saattaisi liittyä myös siihen, että priorisointiin liittyviä päätöksentekoa on vaikea tehdä. Haastateltavien vastausten perusteella Kanban-tauluja käytetään, mutta

Kanban-menetelmää ei, sillä portfolion hallinnassa yhtäaikaisen työn määrää ei rajoiteta. Tämä muodostaa empiirisen johtopäätöksen EC26.

EC26: Portfolion hallinnassa yhtäaikaisen työn määrää ei rajoiteta, mikä saattaa johtua läpinäkyvyyden puutteesta. (N)

4.3.11 PSI / julkaisu

PSI:n / julkaisun tavoite on jatkuva integraatio ja järjestelmän validointi ja viivästyneestä integraatiosta johtuvan riskin vähentäminen. (Turetken ym., 2016) Leffingwellin mukaan julkaisujuna tuottaa tyypillisesti 60-120 päivän sisällä julkaisun tai julkaistavissa olevan inkrementin. Oleellista on, että julkaisulla on kiinteä aikaraja ja laatuaso, mutta julkaisun sisältö saattaa vaihdella (Leffingwell, 2011a).

Jatkuvaa integraatiota järjestelmään tapahtuu ainakin yhden haastateltavan perusteella:

”Tässä sprintin aikana nämä muutokset etenee, niillä tähdätään johonkin tomuun [TORSTAIN VIETÄVÄT MUUTOKSET] esimerkiksi, että saadaan tuossa tomussa vietyä näitä asioita, ja että sen sprintin aikana saataisiin ne asiat vietyä tuotantoon.” - Haastateltava H5

Muiden haastateltavien vastauksissa nostetaan esille julkaisujen koko eli julkaisut ovat kooltaan isoja, jolloin jatkuvaa integrointia ja järjestelmän validointia ei tapahdu kovin usein, mikä nostaa riskejä.

”Siinä tuntuu olevan vähän joka puolella haastetta vielä, että ei osata pilkkoa tehtäviä tarpeeksi pieniksi.” - Haastateltava H2

”Meillä tätä julkaisua on nyt tehty tällä hetkellä noin vuosi” - Haastateltava H6

”Julkaisujen pilkkominen pienemmäksi, niin se on semmoinen ehkä suurin opittava asia mikä musta tuntuu et on vaikein tiimillekin mieltää. Me voisimme tehdä julkaisuja myös pienemmissä, nopeissa sykleissä ja niin, että ei ne tarvitse olla sellaisia, mitkä viedään tuotantoon asti.” - Haastateltava H6

Se, että julkaisut ovat kooltaan isoja näyttäisi liittyvän vahvasti siihen, millä ajatusmallilla kehitettävää kokonaisuutta lähestytään. Kehitettävää kokonaisuutta on lähdetty ratkomaan perinteisen projektimenetelmän keinoin sen sijaan, että ohjelmistokehityksessä olisi hyödynnetty ketterää projektinhallintaa ja ajattelutapaa.

”Sitä ei yksinkertaisesti voi sitä järjestelmää ottaa tuotantokäyttöön ennen kuin siinä on tietyt ominaisuudet. Mut okei, oltaisiin me voitu rakentaa julkaisuina niitä pienempiä ominaisuuksia. -- ja sitten meillä olisikin ollut joku julkaisu x mikä on se mikä viedään tuotantoon asti.” - Haastateltava H6

”Edelleen meillä puhutaan paljon MVP:stä, mutta se MVP-käsite on Kelassa ihan eri asia kuin mitä se on yleensä maailmassa. Meillä se tarkoittaa kokonaista tuotannossa käytettävää järjestelmää millä voidaan etuusratkaisutyötä ruveta tekemään suunnilleen.” – Haastateltava H6

Eli ennen kaikkea kyse on ajattelutavan muutoksesta, eli siitä miten asiaa lähestytään. Tässä kyseisessä tiimissä vallitsevana ajatusmallina on ollut se, että ohjelmistokehityksessä on pystyttävä tuottamaan iso kokonaisuus valmiiksi, ennen kuin sen voi julkaista. Ketterässä projektinhallinnassa ja ohjelmistokehityksessä tuotoksista voi esimerkiksi julkaista julkaistavissa olevan version, jolle tehdään myöhemmin käyttöönotto sen sijaan, että isoja kehitettäviä kokonaisuuksia pyritään saamaan valmiiksi. Haastateltavan vastaukset osoittavat, että julkaisuajattelussa ei olla saavutettu ketterää ajattelutapaa. Tämä muodostaa empiirisen johtopäätöksen EC27.

EC27: Ketterän ajattelutavan vastaisesti julkaisujen koot ovat pääosin liian isoja, mutta osa tiimeistä on omaksunut jatkuvan julkaisun käytännön. (P)

4.3.12 Julkaisujuna

Julkaisujuna koostuu useasta eri ketterästä tiimistä, jotka organisoituvat yhden arvovirran ympärille. Julkaisujunalla varmistetaan se, että tiimit tekevät töitä saman mission, aikataulun ja tahdin mukaan, mikä auttaa jatkuvan tuotekehitysvirran toteuttamisen (Turetken ym., 2016) Vain yksi haastateltavista kertoi olevansa osa julkaisujunaa. Tarkemmin kuvattaessa julkaisujunan toimintaa, haastateltava vastasi:

”Me kerrotaan joka julkaisun aluksi meidän tavoitteet kolmelle kuukaudelle. Meillä on demotilaisuus junassa jaettu niin, että -- päätuoteomistajat kertoo, mitkä meidän tavoitteista täytyi, mitkä ei, missä tilanteissa me ollaan minkäkin Portfolio Epicin kanssa.” – Haastateltava H4

Haastateltavan vastauksen perusteella tämän kyseinen julkaisujunan toiminta vastaa hyvin julkaisujunan tarkoitusta. Kolme muuta haastateltavaa mainitsi, että heidän alueellaan julkaisujunaa ei ole. Yksi haastateltavista kertoi, että heidän tiiminsä oli aiemmin osa julkaisujunaa, mutta vääristä syistä.

”Nyt se on yksittäinen tiimi -- silloin kun mä tulin, niin se oli ikään kuin leivottu osaksi [JUNAN NIMI] junaa, mutta se asia oli ymmärretty väärin, että ei tavallaan ymmärretty, että meillä pitäisi olla jotain yhteyttä, releasoitavaa, joten kielsin tiimiä osallistumasta siihen ja poistin meidät siitä.” – Haastateltava H7

Kaikkialla organisaatiossa julkaisujunan ideaa ei ole ymmärretty ja toisaalta toiset tiimit eivät näe tarvetta julkaisujunalle, sillä he eivät tee töitä yhteisen päämäärän eteen.

”Meillä ei oikeasti ole tunnistettavissa olevaa junaa, vaan me hoidetaan ne hommat, jotka siihen alueeseen kuuluu.” – Haastateltava H8

”Kaikkien tiimien yksikön sisällä ei tarvitse olla tekemisissä kaikkien toistensa kanssa, koska ei niillä ole mitään tekemistä [KESKENÄÄN], toisaalta satunnaisesti totta kai”
- Haastateltava H3

Olemassa oleva julkaisujuna toimii systemaattisella tavalla. Muualla organisaatiossa julkaisujunalle ei ole löytynyt tarvetta, sillä tiimit tekevät töitä eri tavoitteiden eteen ja riippuvuuksia muihin tiimeihin ei ole tunnistettu. Tämä muodostaa empiirisen johtopäätöksen EC28.

EC28: Ainakin yksi julkaisujuna toimii systemaattisella tavalla, muuten julkaisujunien käyttö ei ole yleistä organisaatiossa ja niitä saatetaan käyttää väärin perustein. (P)

4.3.13 Pienemmät ja useammin toistuvat julkaisut

Pienemmät, mutta useammat julkaisut auttavat tekemään nopeammin muutoksia. Julkaisu tulisi tapahtua noin 8 viikon välein. Yksi julkaisu koostuu useasta iteraatiosta. (Turetken ym., 2016) Datan perusteella pieniä ja useammin toistuvia julkaisuja tehdään.

”Oliko se jo yli 20 julkaisua vai melkein keskimäärin joka toinen viikko jotain vietiin tuotantoon” - Haastateltava H8

Kehitysjaksotasolla kyllä suunnittelevat niin kuin, että mitä tällä kehitysjaksolla pitäisi saada tehtyä, joka on pilkottu pienemmiksi asioiksi, joita sitten viedään viikoittain tuotantoon.” - Haastateltava H5

Tämän käytänteen arvioinnissa on hyödynnetty aiemmin alaluvussa 4.3.3. Pienemmät, useammat julkaisut tehtyä empiiristä johtopäätöstä: ”Pienemmät, useammat julkaisut ovat melko laajasti tiimeillä käytössä, mutta osittain julkaisut ovat vielä isoja, mitä viedään tuotantoon harvoin”. Eli nämä kaksi esimerkkiä edustavat sitä joukko, jotka julkaisevat useasti. Muita havaintoja pienemmistä ja useammin toistuvista julkaisuista ei ole datan perusteella ja kaikilla tiimeillä ei ole välttämättä tähän mahdollisuutta:

”Me ei voida julkaista suoraan tuotantoon joka sprintin jälkeen. Meillähän on tarkasti rajattu välillä se, että milloin voidaan julkaista.” - Haastateltava H7

Se, että kaikki tiimit eivät käytä pienempiä ja useammin toistuvia julkaisuja ei ole välttämättä kiinni tiimistä itsestään. Kehittyneimmistä ketterissä tiimeissä muutoksia julkaistaan pienissä erissä ja usein. Tämä muodostaa empiirisen johtopäätöksen EC29.

EC29: Kehittyneimmistä ketterissä tiimeissä muutoksia julkaistaan pienissä erissä ja usein. (P)

4.3.14 Adaptiivinen suunnittelu

Iteraation yksityiskohdat tarkennetaan vasta seuraavaa iteraatiota edeltävänä ajanjaksona, jotta kaikki edellisen iteraation aikana opitut asiat ja palaute saadaan sisällytettyä iteraatioon. (Turetken ym., 2016) Datan perusteella suunnittelu on hyvin adaptiivista ja tarkempia yksityiskohtia ei lyödä lukkoon liian aikaisin.

”Jos puhutaan seuraavan sprintin asioista, aika raa’alla ja vapaalla kädellä saa säveltää uusiksi, kyllä niihin pystytään mukautumaan. Se periaate on koitettu pitää et se kuluva sprintti olisi pyhää maata, että säilyy se tekemisen rauha tiimissä.” - Haastateltava H1

”Sprintin suunnittelussa me lyödään lukkoon se [SPRINTIN SISÄLTÖ] kun ollaan lähempänä sitä itse hetkeä” - Haastateltava H7

”-kuinka paljon tuli uusia tehtäviä sprintin aikana, vaikka sinne ei saisi tulla yhtään, mutta meillähän tulee aina noin puolet.” - Haastateltava H8

”Alkuun kipuiltiin tosi paljon tämän kanssa, että me saatiin noin puolet suunnitelluista tehtävistä tehtyä, puolet jäi tekemättä ja puolen verran ilmestyi lisää tavaraa sinne sprintille.” - Haastateltava H6

Vastauksissa nousee esille oikeastaan se, että koska ollaan niin mukautuvaisia uusiin tilanteisiin, suunnittelu saattaa jäädä joko kokonaan tekemättä tai hyvin kevyeksi. Suunnittelun puuttuminen voi ainakin osittain selittyä sillä, että jotkut tiimeistä käyttävät niin uutta teknologiaa, että suunnitelmia on vaikea lyödä lukkoon ja järkevämpää on keskittyä sen hetken kannalta oikeisiin asioihin ja toisaalta toisilla tiimeillä yllättävät ylläpitotyöt sotkevat suunnittelun.

”Tehdään töitä uusilla järjestelmillä, uusilla teknologioilla. Tiimi opettelee koko aika näitä asioita itse samalla. Meidän on ihan mahdotonta yrittää suunnitella tehtäviä, että me tiedettäisiin oikeasti kaikki mitä pitää tehdä jonkun asian valmistumiseksi.” - Haastateltava H6

”Ollaan ylläpitomoodissa 50 prosenttisesti.” - Haastateltava H8

Molemmissa tapauksissa sprinttiä ei pysty suunnittelemaan täysin etukäteen. Yksi käytänteen arvioitavista asioista on oppien sisällyttäminen seuraavaan iteraatioon. Kuten alaluvussa 4.4.1. Prosessin reflektointi ja parantaminen arvioitiin, prosessin reflektointia ja parantamista tapahtuu, mutta oppeja ei aina hyödynnetä seuraavissa iteraatioissa. Havaintoja siitä, että suunnittelu olisi adaptiivista, löytyi datasta useamman vastauksen perusteella. Kuitenkin, koska reagointi muuttuviin tilanteisiin on niin adaptiivista, jossain tilanteissa suunnittelu jää joko kokonaan tekemättä tai se tapahtuu hyvin kevyesti. Tämä taas johtaa helposti siihen, että suunnittelun hyödyt jäävät toteutumatta. Tämä muodostaa empiirisen johtopäätöksen EC30.

EC30: Adaptiivista suunnittelua on muutamilla tiimeillä käytössä, osalla tiimeistä muuttuviin tilanteisiin reagoiminen aiheuttaa sen, että suunnittelua ei pystytä tekemään ja suunnittelun höydyt jäävät toteutumatta. (P)

4.3.15 Liiketoiminnan tehokkuuden mittaaminen

Käytössä tulisi olla joustava, automatisoitu ja merkityksellinen BSC eli Balanced Scorecard, joka mittaa suorituskykyä tehokkuuden, arvon toimittamisen, laadun ja ketteryyden perusteella. Organisaatio mittaa myös tiimien ja julkaisunien suoriutumista. (Turetken ym., 2016) Datasta ei löytynyt käytänteitä liiketoiminnan tehokkuuden mittaamisesta automatisoidun Balanced Scorecardin avulla, mutta valmistuneiden tehtävien määrää mitataan.

”Tällä hetkellä mitataan esimerkiksi junatasolla sitä, että kuinka monta meidän kehityksen Epiceistä valmistui, niin se ei kerro mitään siitä arvosta. Jos kaikki valmistui niin hienoa, mutta niin ei vaan tapahdu. Meillä valmistuu tällä hetkellä olisiko 30% suunnitelluista ominaisuuksista”- Haastateltava H4

Datassa ei ollut muita havaintoja siitä, että organisaatio mittaisi tiimien ja julkaisujunien suoriutumista ja arvontuottoa. Tämä muodostaa empiirisen johtopäätöksen EC31.

EC31: Organisaatiossa ei tehdä liiketoiminnan arvontuoton ja laadun mittaamista. (N)

4.3.16 Ketterien projektien arviointi

Projektien ketterä arviointia tulisi tehdä, sillä onnistunut lopputulos on kiinni siitä, kuinka hyvä arvio suunnitelmasta on. Ketterien projektien arviointimenetelmiä ovat esimerkiksi viiveen aiheuttama kustannus (engl. *Cost of Delay*), painotettu nopein työ ensin (engl. *Weighted Shortest Job First, WSJF*), tarinapisteet (engl. *story points*). (Turetken ym., 2016) Haastateltavista yksi tunnisti, että heillä projekteja tai isoja kehittämiskokonaisuuksia eli Portfolio Epicejä arvioidaan ja priorisoidaan WSJF:n avulla.

”WSJF itse asiassa avasi meidän päätuoteomistajien silmät -- sitä arviointia tehdään juurikin näiltä [WSJF:n] kanteilta.” - Haastateltava H4

Kysyttäessä, käytetäänkö tiimien Epicien arvioinnissa WSJF- tai muita priorisointimenetelmiä, haastateltava vastasi:

”Meillä jokaiselle suunnittelujaksolle on tehty taulukko ja siinä on se tavoite mitä se tuoteomistaja sille tuotteillensä tämän suunnittelu jakson aikana haluaa. Siellä on ne perustelut esimerkiksi, että kuinka monta käyttäjää tästä hyötyy, mitä riskejä tämän tekemisessä ja tekemättä jättämisessä on. Ja niin kuin niiden kohtien perusteella meidän tiimin tuoteomista, --, tutkii näiden kaikkien tuotteiden tuoteomistajan tekemät jutut ja arvottaa ne sitten prioriteettijärjestykseen meidän Confluenceen. - Haastateltava H7

Tiimillä ei ole käytössä virallista WSJF tai CoD-arviointimenetelmää, mutta tiimi on hyödyntänyt omaa WSJF-menetelmää vastaavaa arviointimenetelmää, missä huomioidaan tuotettava asiakasarvo. Kuten alaluvussa 4.4.4. Ketteryyden arviointi ja kiertonopeus aiemmin arviointiin, monella tiimillä story pointit ovat käytössä, mutta niitä hyödynnetään lähinnä tiimitasolla. Havaintoja story pointien tai muiden arviointimenetelmien käytöstä ketterissä projekteissa ei ole. Tämä voi johtua siitä, että varsinkin tiimitasolla työskentelevät Scrum Masterit ovat vähemmän tekemisissä projektitason ketteryyden arviointimenetelmien kanssa. Ketterien projektien arviointia ei ole organisaatiossa käytössä kovinkaan laajasti. Tämä muodostaa empiirisen johtopäätöksen EC32.

EC32: Ketterien projektien arviointi ei ole organisaatiossa laajasti käytössä. (P)

Yhteenvedona suunnittelu ja jatkuva tuottaminen periaatteeseen löytyi yhteensä 17 empiiristä johtopäätöstä. Näiden yhteenvedojen pohjalta muodostettiin ensisijainen empiirinen johtopäätös PEC2.

PEC3: Tulevien töiden parempi suunnittelu ja töiden pilkkominen pienempiin osiin auttaisi kehitystiimejä tuottamaan asiakkaalle nopeammin arvoa.

Suunnittelu ja jatkuvan tuottamisen empiiriset johtopäätökset ovat esiteltynä seuraavassa taulukossa (TAULUKKO 8).

TAULUKKO 8 Suunnittelun ja jatkuvan tuottamisen empiiriset johtopäätökset

Tunniste	Empiirinen johtopäätös
EC16	Yhteissuunnittelut ovat tiimeillä laajasti käytössä. (F)
EC17	Jatkuvaa tuottamista tapahtuu joissain tiimeissä, mutta organisaatiossa säännöllisellä rytmillä tuottaminen ei ole laajasti käytössä. (P)
EC18	Tiimeissä tehdään kahden tason suunnittelua ja seuraamista, mutta joillain tiimeillä pidemmän aikavälin suunnittelu voi olla puutteellista. (L)
EC19	Tiimit arvioivat työmääriään erilaisin menetelmin. (F)
EC20	Työvälineet eivät tue tarpeeksi tiimien tarpeita, ja osalla työvälineiden osaamisen käytössä on puutteita.
EC21	Julkaisun suunnitteluja pidetään, mutta tiimit eivät julkaisun näkökulmasta koordinoi tekemisiään oikeiden tiimien kanssa ja tilaisuudessa ei hallita riippuvuuksia toisiin tiimeihin. (P)
EC22	Riskilähtöiset iteraatiot, jossa sprintti suunnitellaan riskien minimoimiseksi ei tapahdu lähes ollenkaan. (N)
EC23	Asiakkaan tarpeita ilmaistaan sekä ominaisuuksien että tehtävien muodossa. (P)
EC24	Tiekarttoja käytetään organisaatiossa melko laajasti. Kaikilla tiekartoissa ei ole yhteensovitettu liiketoiminnallisia ja teknisiä vaatimuksia. (P)

- EC25 Osa tiimeistä pystyy tuottamaan testattua toimivaa ohjelmistoa yhden sprintin aikana. (P)
- EC26 Portfolion hallinnassa yhtäaikaisen työn määrää ei rajoiteta, mikä saattaa johtua läpinäkyvyyden puutteesta. (N)
- EC27 Ketterän ajattelutavan vastaisesti julkaisujen koot ovat pääosin liian isoja, mutta osa tiimeistä on omaksunut jatkuvan julkaisun käytännön. (P)
- EC28 Ainakin yksi julkaisujuna toimii systemaattisella tavalla, muuten julkaisujunien käyttö ei ole yleistä organisaatiossa ja niitä saatetaan käyttää väärin perustein. (P)
- EC29 Kehittyneimmistä ketterissä tiimeissä muutoksia julkaistaan pienissä erissä ja usein. (P)
- EC30 Adaptiivista suunnittelua on muutamilla tiimeillä käytössä, osalla tiimeistä muuttuviin tilanteisiin reagoiminen aiheuttaa sen, että suunnittelua ei pystytä tekemään ja suunnittelun höydyt jäävät toteutumatta. (P)
- EC31 Organisaatiossa ei tehdä liiketoiminnan arvontuoton ja laadun mittaamista. (N)
- EC32 Ketterien projektien arviointi ei ole organisaatiossa laajasti käytössä. (P)
- PEC3 Tulevien töiden parempi suunnittelu ja töiden pilkkominen pienempiin osiin auttaisi kehitystiimejä tuottamaan nopeammin asiakkaalle arvoa.

4.4 Ihmiskeskeisyys

Ihmiskeskeisyys on kypsyytasomallin kolmas periaate. Tähän alalukuun on koottuna alkuun ihmiskeskeisyyden periaatteeseen liittyviä havaintoja, ja havainnot ovat jaoteltuna tarkempiin arvioitaviin käytänteisiin.

4.4.1 Yhteistyötä tekevät tiimit

Tiimin jäsenet kommunikoivat ja tekevät yhteistyötä keskenään ja muiden tiimien kanssa. (Turetken ym., 2016) Oman tiimin sisällä yhteistyö sujuu pääosin hyvin, mutta merkkejä kahtiajakautuneista tiimeistä löytyy. Kysyttäessä, minkälaista yhteistyö tiimin sisällä on ja toimiiko se, haastateltavat vastasivat:

”Kehittäjät ollut innokkaana mukana, et hekin voisi tulla mukaan testitapauksia laatimaan. Testaajat ottavat vastuuta siitä, että kirjoitetaan testejä Eclippessä vaikka Pythonilla.” – Haastateltava H1

”Jos ihmisillä on vaikka yhdellä kehittäjällä toiselle kehittäjille asiaa, niin hän kirjoittaaakin sen tiimin yhteiselle kanavalle, että mitäs tämä nyt tarkoittaa? Muutkin saa sen tiedon, että tällainen keskustelu käynnissä. -- eli yksityisiä keskusteluja käydään julkisesti.” – Haastateltava H8

”Jossakin se toimii ja jossakin nämä yksiköiden, vaikka IT vs liiketoiminta. Niin se jotenkin paistaa aika usein helposti kaikessa keskustelussa.” – Haastateltava H2

Tällä haastateltava viittaa siihen, että Scrum-tiimit saattavat koostua määrittelijöistä, testaajista, ohjelmoijista, tuoteomistajasta ja Scrum Masterista, joista määrittelijät ja tuoteomistaja kuuluvat liiketoiminnan organisaatioyksikköön. Myös toinen haastateltava nosti esille, että tämä organisaation raja-aita on tuottanut ongelmia nimenomaan tiimien sisällä:

”Sellaisia käytännön ongelmia tulee, kun meidät on jaettu kahteen eri organisaatio yksikköön. Sanotaan että tyhy-rahoista tiimi saa käyttää johonkin yhteiseen juttuunsa. Ne ei voi kutsua sinne tuoteomistaja, koska sillä on eri tyhy-rahat. – Haastateltava H4

Organisaatiossa olevat raja-aidat erityisesti yhden tiimin sisällä saattavat aiheuttaa kitkaa tiimin jäsenten välille ja siten romuttaa tiimijattelua. Merkkejä yhteistyöstä niin tiimi- kuin yksikkörajojenkin yli löytyy kuitenkin paljon. Kysyttäessä, minkälaista yhteistyö organisaatiossa on ja näkykö, että tiimit tekisivät keskenään yhteistyötä, haastateltavat vastasivat:

”Joo ja ei.” – Haastateltava H3

”Raketti-palstaa [Rocket.Chat] jos mä itse ajattelen testiautomaatiopuolta, niin rakettikanava on siellä ihan älyttömän hyvin toiminut tukena.” – Haastateltava H6

”Testausautomaatio-tiimin taholta pidetään -- tällöinen klinikka, missä voi käydä kyselemässä asioita” – Haastateltava H5

Organisaation on onnistunut luomaan yhteistyötä tukevia rakenteita, kuten CoP-toimintaa (engl. Community of Practice) ja klinikkatyöskentelyä. CoP tarkoittaa käytäntöyhteisöä, jossa ryhmä ihmisiä, jotka jakavat saman huolenaiheen, ongelman tai mielenkiinnon kohteen kokoontuvat yhteen jakamaan parhaimpia käytänteitä ja luomaan uutta tietoa aihealueesta (Wenger, 1999).

”Meillä on nyt kehitelty näitä Scrum Master yhteistyökuvioita, että on Scrum Master CoP:in lisäksi myös yksikön laajuinen Scrum Mastereiden kokoontuminen.” – Haastateltava H1

”Meillä on RTE CoP:pia, meillä on Kela tason Scrum Mastereiden CoP ja niin edespäin, yhteistyötä eri yksikön yksikköjen välillä on sitä kautta saatu enemmän -- kontakteja muodostuu helpommin kun meillä on se CoP-toiminta” – Haastateltava H4

Sen lisäksi että organisaatiossa on yhteistyötä tukevia rakenteita, yksi haastateltava nosti erikseen esille, että yhteistyö on ollut kokonaisvaltaisesti toimivaa.

”Itselläni mulla on ollut ilo ja etuoikeus saada jatkuvasti työskennellä yhteisöissä ja tahojen kanssa, jotka ovat avoimia tekemään sitä yhteistyötä yli tiimirajojen” – Haastateltava H6

Kaksi haastateltavista nosti esille myös tilanteita, joissa yhteistyötä ei tehdä, sillä toinen osapuoli ei sitä halua:

”Kyllä mä olen kuullut tarinoita siitä, että jotkut sanoo, että saa suunnilleen näpeilleen, jos uskaltaa toisen tiimin ihmisen tai jotain kysyä, että ei ole tästä taskia Jirassa, en voi keskustella kanssasi.” – Haastateltava H6

”Jossain oli ajateltu, että kun heiltä oli pyydetty apua, ettei he nyt voi, kun tämä sprintti on jo suunniteltu ” – Haastateltava H3

Kaiken kaikkiaan organisaatiossa tehdään paljon yhteistyötä niin tiimin sisällä kuin tiimi- ja yksikkörajojen yli. Joissain tilanteissa yhteistyö ei toimi sekä tiimin sisällä että tiimien välillä. Tämä muodostaa empiirisen johtopäätöksen EC33.

EC33: Yhteistyötä tehdään laajasti yli tiimi- ja yksikkörajojen, kaikki tiimit eivät kuitenkaan tee yhteistyötä muiden tiimien kanssa ja paikoitellen yhteistyö tiimin sisällä tai tiimien välillä ei toimi. (P)

4.4.2 Voimaantuneet ja motivoituneet tiimit

Tiimeillä on aito valta tehdä päätöksiä itse. Auktoriteettiasemassa olevat motivoivat, valtuuttavat ja varustavat tiimit ratkaisemaan ongelmansa ja tehtävänsä itsenäisesti. (Turetken ym., 2016) Kysyttäessä, voivatko tiimit aidosti tehdä itse päätöksiä, haastateltavat vastasivat:

”Se varmaan riippuu siitä, että minkälainen ympäristö siinä on. Tuetaanko sitä ajatusta, että saatte vapaasti miettiä ja tehdä niin kuin näette parhaaksi” – Haastateltava H2

”Olen ihan positiivisesti yllättynyt siitä, että tiimeillä on tosi paljon valtaa, miten tehdään asioita. Millä tehdään, niin siihen ei välttämättä olekaan sananvaltaa, että tavallaan sen, miten tehdään, on myös riippuvainen, millä välineellä me mennään.” – Haastateltava H1

”Ketteryyden kannalta tiimi pystyy tekemään mun mielestä hyvin pienissä raameissa itsellisesti päätöksiä. Aika paljon asiat on annettu. – Haastateltava H5

”Kyllä ja ei, aiheesta riippuen. Joissakin aiheissa tiimi pystyy hyvinkin itsenäisesti, ja olen oikeastaan vähän hämmästynyt siitä, että miten itsenäisesti saadaan nykyään tehdä asioita ja päättää.” – Haastateltava H6

Haastateltava tarkentaa esimerkillä, minkälaisina tekoina se näkyy, että tiimi ei saa itse ratkaista ongelmiaan:

”Kun puhuttiin sprintin kestosta, että meistä tuntuu, että kolme viikkoa on liian pitkä, mahdotonta suunnitella niin pitkälle ajalle, niin todettiin, että meidän pitää mennä samassa sprinttisyklissä, kun [TIIMIN NIMI] menee.” – Haastateltava H6

Epäselvää haastateltavan vastauksesta jää, mikä oli lopulta se taho tai rooli, joka totesi, että sprinttisykliä ei saa muuttaa. Joka tapauksessa tiimi ei itse saanut päättää itseään koskevasta asiasta. Organisaatiossa näyttäisi olevan kipuilua

siitä, missä rajoissa ketterät tiimit saavat päättää itsenäisesti itseään koskevista asioista ja missä kohtaa organisaation tulisi tehdä linjaus:

”Jossain tiimit haluaisivat päättää paljon enemmänkin ja toisaalta jossain haluttaisiin nimenomaan, että tulisi asiat keskitetympään, riippuen asiasta ja riippuen tiimistä.” – Haastateltava H3

”On ihmisiä, joilla on rooli, että heidän kuuluu päättää tällaisista asioista, mikä taas on vastoin sitä, että tiimi itse saa päättää ja itse saa linjata.” – Haastateltava H6

Osa tiimien voimattomuuden tunteensa voisi johtua siitä, että organisaatiossa on aikaisemmin vallinnut perinteinen johtamiskulttuuri, jossa päätöksien odotetaan tulevan muualta ja kyseinen johtamiskulttuuri ei ole vielä poistunut organisaatiosta. Tätä tulkintaa tukee kahden haastateltavan vastaukset:

”Meillä perinteisesti on tosi vahva linjaorganisaatio -- jos ajatellaan, että tuoteomistajien pitäisi pystyä päättää asioita -- niin ei se kyllä ole kauhean pitkällä se asia. Kyllä vahva linja siinä ympärillä puuttuu kaiken maailman asioihin.” – Haastateltava H2

”Oon kuullut jopa semmoisia kommentteja, kun on yritetty laittaa hommaa toimimaan tuoteomistajan kanssa, niin sieltä on linjaesimies tullut sanomaan, että hänen takana on loppupeleissä kaikki priorisointi, hänen kautta pitää mennä kaikki” – Haastateltava H7

Organisaatiokulttuuri, jossa linjaorganisaatio on perinteisesti päättänyt asioista, saattaa toimia osittain esteenä päätöksenteon madaltamiselle. Kysyttäessä linjajohtoon kuuluvalta haastateltavalta, minkälaisin keinoin tai teoin se näkyy, että linjaorganisaatio olisi luopunut päätöksenteonvallasta joissain kohdin, haastateltava vastasi:

”En tiedä mitä päätöksiä tekisin, jotka vaikuttavat sinne tiimien arkeen.” – Yksikön päällikkönä toimiva haastateltava

Toinen haastateltava toteaa, että joissain asioissa linjajohto tukee tiimiä hyvin, mutta joissain tilanteissa linjajohto ei ymmärrä täysin omien tekojensa vaikutuksia:

”Sanotaan, että noudatetaan Scrumia, meillä on sprinttisyklit ja sitten organisaation johto iskee omia infoja ja muita siihen kohtaan sprinttisykliä, kun kaikki todennäköisesti on varanneet sprintin suunnittelupalaverin tai retrot. Tavallaan sieltä linjaorganisaation puolelta tulevissa asioissa ei huomioidakaan yhtään sitä, miten tiimien arki rakentuu ja muodostuu, vaikka se on heidän päätöksellensä ruvennut muodostumaan.” – Haastateltava H6

Näiden haastateltavien vastausten perusteella voisi tehdä sen tulkinnan, että ainakin osittain tiimien voimaantumisen esteenä on linjajohdon ymmärtämättömyys omien päätöksentekojensa vaikutuksista ketteriin tiimeihin. Tämä muodostaa empiirisen johtopäätöksen EC34.

EC34: Tiimien voimaantumisen esteenä on linjajohto ja sen ymmärtämättömyys omien päätöksentekojensa vaikutuksista ketteriin tiimeihin.

Vastauksissa nousee esille myös asian toinen puoli eli joissain tilanteissa tiimillä olisi valta tehdä päätöksiä, mutta tiimeissä ei ole uskallusta tehdä päätöksiä itse:

”Osa on tietysti sitäkin, ettei uskallakaan välttämättä ottaa sitä vastuuta.” - Haastateltava H2

”Osa tiimistä ei ehkä ole vielä henkisesti kasvanut tähän, että me voidaan itse päättää, me saadaan itse päättää, että meillä ainakin tiimissä on vielä sitä henkeä, että tästä pitää olla joku arkkitehti päättämässä” - Haastateltava H6

Ajatusmallin muuttaminen ja sitä kautta käyttäytymisen muuttaminen

”Aikanaan vielä kun tämä oli uutta, niin oli semmoista, että miksei kukaan päättä mitään, mutta nyt on saatu tämmöinen ajatusmalli ihmisten päähän, että kuka sen paremmin päättää, kun me” - Haastateltava H8

Tiimit eivät ole tarpeeksi voimaantuneita uskaltaakseen tehdä itse päätöksiä. Se, miksi tiimit eivät uskalla tehdä itse päätöksiä voi liittyä siihen, että pelätään päätöksenteon mukana tulevaa vastuuta:

”En tiedä kummasta se on enemmän siitä, että eikö uskalleta tehdä päätöstä ja ajatellaan, että siitä tulee se vastuu. Jos meneekin pieleen niin sitten minua syytetään – Vai onko se sitä, meillähän on lähellä myös arkkitehtitiimi ja niin kauan, kun se siinä lähellä pyörii ja on tämmöisiä nimettyjä arkkitehtejä niin ei ehkä myöskään pelätään ehkä, hypätään heidän varpailleen.” - Haastateltava H6

Toisaalta tiimit ja yksilöt eivät ole välttämättä saaneet tarvittavaa opastusta, kannustusta ja tukea siihen, miten olla itseohjautuva. Eli auktoriteettiasemassa olevat henkilöt eivät välttämättä anna tiimeille tarpeeksi tilaa kehittyä päätöksentekijöinä. Tätä tulkintaa tukee haastateltavan vastaus:

”Ymmärtäisivät [AUKTORITEETTIASEMASSA OLEVAT HENKILÖT] sen, että vaikka tulitisiin kysymään päätöstä, että voi sanoa siihen: ’oikeastaan te tiedätte paremmin tämän asian, että voitte päättää itse sen’.” - Haastateltava H2

Päätöksenteon madaltaminen ja johtamiskulttuurin muuttaminen perinteisestä johtamisesta valmentamaan johtamiseen on hankalaa, sillä kaikki eivät ole valmiita luopumaan vanhoista vastuista ja tavoista:

”Siinä varmaan taistelee ne vanhat omat toimintamallit ja -- ei välttämättä kaikki ole edes halukkaita luovuttamaan sitä päätösvaltaa.” - Haastateltava H2

Haastateltavien vastausten perusteella tiimit eivät ole tarpeeksi voimaantuneita uskaltaakseen tehdä itse päätöksiä ja tiimejä ei tueta tarpeeksi, jotta he voimaantuisivat. Todisteita siitä, onko tiimeillä aito valta tehdä päätöksiä itse, on jonkin verran, mutta laajaa tiimien voimaantumista ei ole nähtävillä

haastateltavien vastausten perusteella. Tämä muodostaa empiirisen johtopäätöksen EC35.

EC35: Kaikki tiimit eivät uskalla ja joissain tiimillä ei ole valtaa tehdä päätöksiä itse. Tiimit tarvitsisivat tukea voimaantumisessa. (P)

4.4.3 Määrittely/Ohjelmointi/testaus -tiimi

Tiimissä on kaikki tarvittava osaaminen määrittelystä, ohjelmointiin ja testaamiseen. Moniosaava tiimi on itseorganisoituva, valtuutettu ja itseohjautuva ja se pystyy tuottamaan toimivan kokonaisuuden lyhyessä, rajatussa aikavälissä. (Turetken ym., 2016) Kysyttäessä, löytyykö tiimeissä kaikki tarvittava osaaminen ohjelmiston tuottamiseksi, yksi haastateltava vastasi:

”Ei läheskään kaikista tiimeistä.” - Haastateltava H3

Tässä asian yhteydessä pitää kuitenkin huomioida se, että hyvin harva kehitystiimi tekee pelkkää kehittämistä vaan samat tiimit vastaavat myös esimerkiksi sovelluksien ylläpidosta ja tiimin vastuulla olevien palvelujen tai tuotteiden määrä on iso.

”Tiimissä yhdellä ihmisellä pitää olla kahdenkymmenen vuoden erilainen teknologiakirjo. Jos on viisi tai kuusi järjestelmää, ne on kaikki tehty eri lailla ja pitää osata se substanssi sen päälle, -- et on määrittely, testaus, kehittäjä, sen päälle kolmas ulottuvuus on kaikki se teknologiakirjo.” - Haastateltava H3

”Meilläkin on niin laaja skaala eri teknologioista. Joitakin juttuja tehdään PL/1:sellä, toisessa päässä on konttiratkaisut ja käytetään Robot Frameworkia ja Reactia kohtapuolin” - Haastateltava H1

Koska teknologiakirjo on niin valtava, tiimissä ei välttämättä löydy kykyä sekä kehittää että ylläpitää useamman vuosikymmenen teknologioilla tuotettuja järjestelmiä ja sovelluksia. Arvioitavan käytänteen näkökulmasta tarkoituksena on keskittyä pelkästään uuden tuottamiseen. Alaluvussa 4.2.3. Pienemmät, useammat julkaisut tehtiin johtopäätös ” Pienemmät, useammat julkaisut ovat melko laajasti tiimeillä käytössä, mutta osittain julkaisut ovat vielä isoja, mitä viedään tuotantoon harvoin.”. Eli useat tiimit kykenevät tuottamaan toimivan kokonaisuuden rajatussa aikavälissä. Tästä näkökulmasta katsottuna tiimeistä löytyy pääsääntöisesti tarvittava osaaminen määrittelystä, ohjelmointiin ja testaamiseen.

Käytänteen arvioinnissa tulee arvioida myös sitä, ovatko tiimit itseohjautuvia. Tiimien itseohjautuvuutta käsiteltiin edellisessä alaluvussa 4.4.2. Voimaantuneet ja motivoituneet tiimit. Huomioitavaa on, että osalla tiimeistä on haasteita tehdä päätöksiä itse erilaisien syiden takia. Tiimit eivät välttämättä ole kaikki myöskään itseorganisoituvia siinä mielessä, että ne huolehtisivat kollektiivisesti koko tiimin asioista. Yksi haastateltavista nosti erikseen esille, että

esimerkiksi tiettyjä ohjelmistokehitykseen liittyviä osia saatetaan pitää vähemmän tärkeinä, jolloin ne jäävät tiimissä vähemmälle huomiolle:

”Tämä ei toimi minun mielestäni vielä itseohjautuvasti ja automaattisesti, että saataisiin jo siinä vaiheessa, kun otetaan sinne backlogille joku työ, nii siellä olisi jo tavallaan tehty erillinen story tai sub-task tai muu vastaava, et tässä on testausautomaatio.” - Haastateltava H5

Organisaation tiimit ovat pääsääntöisesti tiimejä, joista löytyy tarvittava määrittely-, ohjelmointi- ja testausosaaminen ohjelmistokehitystä varten, mutta koska tiimeillä on myös ylläpitovastuita, tiimistä ei löydy kaikkea tarvittavaa osaamista. Tämä muodostaa empiirisen johtopäätöksen EC35.

EC35: Organisaation tiimit ovat pääsääntöisesti tiimejä, joista löytyy tarvittava määrittely-, ohjelmointi- ja testausosaaminen ohjelmistokehitystä varten, mutta koska tiimeillä on myös ylläpitovastuita, tiimistä ei löydy kaikkea tarvittavaa osaamista. (P)

4.4.4 Itseohjautuvat tiimit

Tiimi on valtuutettu tekemään itse päätöksiä odottamatta johdon hyväksyntää. Tiimit ovat poikkitoiminnallisia, tiimin jäsenten roolit ja vastuut eivät ole tarkkaan määriteltyjä ja koko tiimi on vastuussa toimivan ohjelmiston toimittamisesta. (Turetken ym., 2016) Datan perusteella vain yksi tiimi on sellainen, missä roolivastuut eivät ole tarkkaan määriteltyjä. Kysyttäessä, onko tiimissä selkeärajaisia rooleja, haastateltava vastasi:

”Ei ole” - Haastateltava H8

Kysyttäessä ovatko tiimit moniosaavia tiimejä, joissa roolit tiimin jäsenten välillä ei ole lyöty lukkoon, haastateltavat vastasivat:

”Pääsääntöisesti eivät ole. Se on yks haaste.” - Haastateltava H2

”Ei kaikki tekee kaikkia asioita.” - Haastateltava H1

Tiimin jäsenten roolit saattavat paikoin olla hyvin tarkkaan määritelty ja yksi haastateltavista nostaa esille, että ongelma alkaa jo rekrytointivaiheessa:

”Me palkataan edelleen määrittelijöitä, toteuttajia ja testaaajia. Eli se moniosaaminen ja sen korostaminen, sitä ei ole tehty ja sen ikävä kyllä huomaa. Se pitäisi saada ennemmin sellaiseksi, että katsottaisiin mitä osaamista tiimissä tarvitaan ja laajennettaisiin siihen osaamiseen mitä tiimissä tarvitaan, että vähän päästäisiin rajo-aidoista pois.” - Haastateltava H4

Jo rekrytointivaiheessa lukkoon lyöty rooli voi aiheuttaa sen, että myöhemmin tehtävänkuvaa ei voi laajeta aidon Scrum-tiimin jäsenen työnkuvan mukaisesti, jossa testaamista voi tehdä esimerkiksi ohjelmoijaksi palkattu henkilö. Organisaatiossa on näkyvissä kuitenkin jonkinasteista osaamisen laajentamista,

mikä mahdollistaa sen, että tiimit olisivat jatkossa moniosaavampia ja henkilöitymisen sijaan tiimi olisi kollektiivisesti vastuussa ohjelmiston tuottamisesta erillisten roolien sijaan.

”Lähinnä se, et on sellaista kiinnostusta laajentaa [OSAAMISTA], kun on alettu kehittämään automaattitestejä, siinä on kehittäjät ollut innokkaana mukana, et hekin voisi tulla mukaan testitapauksia laatimaan.” - Haastateltava H1

Yksi haastateltavista nostaa kuitenkin esille, että osaamisen laajentamista ei tule kuitenkaan tapahtumaan riittävällä nopeudella, sillä yhtäaikaista tekemistä on jatkuvasti liikaa:

”Minun mielestäni lähes kaikki mikä meillä on probleemana tämän tyyppisissä asioissani, johtuu siitä, että meillä on liikaa samanaikaista tekemistä. Ja se pohjautuu priorisointiin.” - Haastateltava H4

Vaikka roolirajat voivat osassa tiimejä olla selkeitä, toiminta voi olla kuitenkin itseohjautuvaa.

”Oon tyytyväinen siihen, miten itseohjautuvaa porukkaa meidän tiimissä on.” - Haastateltava H1

Kuten alaluvussa 4.6.2 Voimaantuneet ja motivoituneet tiimit arvioitiin, suurin osa tiimeistä ei ole vielä joko tarpeeksi voimaantuneet uskaltaakseen tehdä päätöksiä itse tai joissain tapauksissa auktoriteettiasemassa olevat tahot estävät tiimin itsenäisten päätösten tekemisen. Organisaation tiimit eivät ole systemaattisesti moniosaavia ja itseohjautuvia tiimejä ja todisteita käytännön saavutuksista on vähän. Tämä muodostaa empiirisen johtopäätöksen EC36.

EC36: Pääsääntöisesti tiimit koostuvat tarkkarajaisista rooleista ja tiimit eivät ole aina valtuutettu tekemään päätöksiä itse. (P)

4.4.5 Säännöllinen henkilökohtainen vuorovaikutus

Tehokkaan ja toimivan kehitysprosessin luomiseksi tiimin jäsenten välinen säännöllinen kommunikointi on välttämätöntä. (Turetken ym., 2016) Kysyttäessä, miten kommunikointi tiimin sisällä toimii, haastateltavat vastasivat:

”Kommunikaatio pelaa hyvin.” - Haastateltava H1

”Raketti [ROCKET.CHAT] on ihan keskeinen siinä tänä päivänä -- Se on tosi keskeinen, sit on tietysti ihan kokouksia, workshoppeja ja tämmöisiä tehdään ja sitten tietysti kahdenkeskisiä palavereja ja chattailu” - Haastateltava H6

”Raketti on varsinkin siinä toisessa tiimissä tosi hyvässä käytössä. -- useamman henkilön välisiä puheluita käydään.” - Haastateltava H8

Kommunikointi tiimeissä tapahtuu eri työvälineiden avulla, osa kommunikoinnista on henkilökohtaisempaa ja osa enemmän tiedotusluonteista. Monilla tiimeillä on myös dailyt säännöllisin väliajoin käytössä, mikä lisää henkilökohtaista vuorovaikutusta:

”Meillä on karsitut dailyt eli meillä on keskiviikko, perjantai dailyt ja maanantaisin meillä on weekly.” - Haastateltava H6

”Se maanantain weekly, mä haluaisin sen tuoda siihen koska mun mielestä -- tavallaan puuttui se tila, että missä voin informoida ja tiedottaa tai keskusteluttaa tiimillä tällaisia hallinnollisia asioita.” - Haastateltava H6

”Aika lailla on käytössä, vaikka osa pitääkin niitä vain kerran viikossa tai kaksi kertaa viikossa.” - Haastateltava H2

Kaikilla tiimeillä kommunikointi ei ole vuorovaikutteista, ja tulkinnanvaraiseksi jää, onko kommunikointi myöskään säännöllistä:

”Joissakin näkee sellaista porukkaa, että ne ovat tehneet yhdessä, on hyvä meininki ja uskalletaan puhua, ja taas on joitakin [TIIMEJÄ], joissa on kireyttä havaittavissa. Osa [TIIMEISTÄ] ei puhu.” - Haastateltava H2

Organisaatiossa on pääsääntöisesti hyvin monipuolista, säännöllistä ja henkilökohtaista vuorovaikutusta, joissain yksittäisissä tiimeissä vuorovaikutus ei toimi. Tämä muodostaa empiirisen johtopäätöksen EC37.

EC37: Organisaatiossa tiimeillä on pääsääntöisesti hyvin monipuolista, säännöllistä ja henkilökohtaista vuorovaikutusta, joissain yksittäisissä tiimeissä vuorovaikutus ei toimi. (L)

4.4.6 Scrumien scrum

Scrumien scrumilla (eng. *Scrum-of-Scrums* eli SoS) on keino koordinoita tiimien välistä yhteistyötä (Schwaber, 2004). Scrumien scrum tai SoS-tapaamiset ovat organisaatiossa vähän käytössä, sillä moni tiimi ei kuulu esimerkiksi julkaisujunaan, jossa tiimit organisoituvat yhden arvovirran ympärille ja tiimeillä on tarve koordinoita tekemisiään muiden tiimien kanssa. SoS-tapaamiset ovat käytössä kahdella haastateltavalla.

”Kaks kertaa viikossa meillä on tällainen SoS-tapaaminen. Ja sen lisäksi on joka sprintissä tällainen super SoS, missä on tuoteomistajat ja Scrum Masterit eli käydään yleisiä asioita ja seuraavan sprintin juttuja läpi ja kaikkea muutakin, mittareita, riskejä” - Haastateltava H4

”Kerran viikossa meidän alueen Scrum Masterit tapaa toisensa. Lähinnä kerrotaan mitä tapahtuu ja pysytään kartalla, että mitä tiimeissä on ja mitä muutoksia siellä on. Meillä ei ole mitään akuutteja ongelmia esimerkiksi riippuvuuksia muihin” - Haastateltava H8

Vastauksien perusteella scrumien scrum vastaa sisällöllisesti tarkoitustaan vain yhdellä haastateltavista. Jälkimmäisen haastateltavan vastauksessa tulisi huomioida, että tiimi ei kuulu julkaisujunaan. Eli scrumien scrumia käytetään nimellisesti, mutta sisällöllisesti siellä ei koordinoita tekemisiä muiden tiimien kanssa. Tämä muodostaa empiirisen johtopäätöksen EC38.

EC38: Scrumien scrum on organisaatiossa käytössä vain muutamalla tiimillä ja kaikilla tilaisuus ei vastaa sisällöllisesti alkuperäistä tarkoitustaan. (P)

4.4.7 Hajautettujen tiimien hallinta

Hajautettujen tiimien hallinta tulisi hoitaa asianmukaisella viestinnällä sekä tarvittavalla verkko- ja työkalurakenteilla. (Turetken ym., 2016) Organisaation ketteristä tiimeistä suurin osa on hajautettuja ja ne toimivat pääosin etänä.

”Melkein on hajautettuja tiimejä varmaan kaikki.” – Haastateltava H2

”Itseasiassa ei varmasti edes voitaisi olla läsnä yhdessä, koska ollaan eri puolelta Suomea.” – Haastateltava H1

Organisaatio tarjoaa etätöihin tarvittavat työvälineet kommunikoida tiimin kanssa. Eli vaikka ketterä tiimi ei ole fyysisesti samassa paikassa tiedonvaihto on mahdollistettu tarkoituksenmukaisilla verkko- ja työkalurakenteilla.

”Fyysinen sosiaalisuus jää pois, mutta nämä välineet on kehittynyt niin hyvin. On eri välineitä -- on kaikki nämä, niiden kanssa pystyy hyvin pelaamaan.” – Haastateltava H5

”Mun mielestä toimii paremmin etänä tämä kommunikaatio.” – Haastateltava H8

”Tavallaan ei ole jäänyt välineestä kiinni kommunikointi etänäkään.” – Haastateltava H1

Haastateltavien vastausten perustella organisaatio on hoitanut tarvittavat verkko- ja työkalurakenteet tukemaan hajautettujen tiimien hallintaa ja viestintää. Tämä muodostaa empiirisen johtopäätöksen EC39.

EC39: Hajautettujen tiimien hallintaan löytyy tarkoituksenmukaiset työkalut, jotka tukevat tiimien viestintää. (F)

4.4.8 Ihanteellinen fyysinen ympäristö ketterälle kehittämiselle

Ihanteellinen ketterä fyysinen kokoonpano auttaa luomaan oikean ympäristön ketterän ohjelmistokehityksen menestymiseen. Tämän järjestelyn tarkoituksena on se, että tiimi on yhdessä paikkaa ja tietoa voidaan jakaa tiimin jäsenten kesken heti. (Turetken ym., 2016)

”Kyllähän meillä paljon se yhteistyö on sitä, että meidän pitää katsoa samaa näyttöön -- niin olihan se hyvin usein sitä, että kumpikin istui omalla koneella, koska piti jakaa sitä näyttöä -- Istuttiin samassa huoneessa selät vastakkain Skype-kuulokkeet korvilla ja puhuttiin Skypen välityksellä ja jaettiin sitä näyttöä siinä Skypessä.” - Haastateltava H6

Eli kehitystiimit on rakennettu ainakin pääosin osaamisen ei paikkakunnan mukaan. Viimeisimmän haastateltavan vastauksesta epäselväksi jää, eikö fyysinen ympäristö ollut rakennettu ihanteelliseksi ketterälle tiimille vai onko joissain tapauksissa etätyöskentely jopa optimaalisempi ratkaisu ketterälle ohjelmistokehitykselle. Tätä tulkintaa tukee toisen haastateltavan vastaus:

”Välillä haluan silloin kun teen keskittymistä vaativaa työtä, niin mielellään olenkin etänä.” - Haastateltava H5

Etätyöskentelystä huolimatta tiedonvaihto ketterässä tiimissä voi sujua hyvin.

”Itse tykkään, jos on lähitiimi, tavallaan se tiedonvaihto on siinä kaikista nopeinta, mutta ei tämä huono ole näinkään. On siinäkin mielessä positiivisesti yllättynyt, miten kivuttomasti tämä työaika on mennyt.” - Haastateltava H1

”Mun mielestä [TYÖ] sujuu täysin vastaavasti, että päin vastoin ihmiset saa sen työrauhan ja pystyy paremmin suunnittelemaan omaa työpäiväänsä, kun ollaan etänä eikä ole sitä ympärillä olevaa hälinää, ja ehkä ei niin paljon keskeytyksiä, kun jokainen ohikulkija ei pysähdy juttelemaan niitä näitä.” - Haastateltava H6

Etätyössä ja maantieteellisesti hajautetuissa tiimeissä menetetään kuitenkin se etu, että ongelman ilmetessä kynnys kysyä voi olla matalampi:

”Et siinä ei tule sitä ihan samanlaista ketterän toiminnan mallia mihin itse sovelluskehittäjänä esimerkiksi totuin silloin kun olin ketterässä tiimissä. Kun tuli joku ongelma niin saattoi heittää heti käden pystyyn” - Haastateltava H5

Ketterässä ohjelmistokehityksessä maantieteellisesti hajautetut tiimit ja etätyö eivät tue täysin ketterän tiimin toimintaa, mutta joissain tilanteissa fyysisesti eri paikoissa työskentely voi olla optimaalisempaa. Eli vaikka organisaation tiimit ovat maantieteellisesti hajautettuja tiimejä, se ei aina tarkoita sitä, etteikö se olisi ketterälle tiimille joskus paras ympäristö menestyä. Tämä muodostaa empirisen johtopäätöksen EC40.

EC40: Maantieteellisesti hajautetut tiimit ja etätyö eivät rajoita ketterän tiimin toimintaa, mutta joissain tilanteissa siinä menetetään ketterälle kehittämiselle optimaalinen vuorovaikutus eri ohjelmoijien välillä. (P)

4.4.9 Organisaation muuttaminen

Organisaatioiden on tehtävä olennaisia muutoksia saavuttaakseen ketterän hyödyn yrityksen tasolla. Muutokset voivat tarkoittaa muutoksia suunnittelukäytäntöihin, operaatioihin kohdistuvien vaikutusten hallintaan,

uudelleenorganisoinnista jne. Organisaatiomuutoksissa johdon tuki ja sitoutuminen prosessiin sekä esimerkillä johtaminen ovat avaintekijöinä hyötyjen saavuttamisessa. (Turetken ym., 2016) Kysyttäessä, tukeeko johto ketteryyttä, haastateltavat vastasivat:

”Kyllä meidän yksikössä ainakin. Meidän yksikössä tukee lähinnä niin, että me saadaan Scrum Masterit keskenämme tehdä asiat just niin kuin meistä tuntuu parhaiten ketteriltä. -- Mä en tiedä ihan rehellisesti sanottuna, että miten siitä ylöspäin” - Haastateltava H8

”Muistan [LINJA]JOHDOLTA] pyytäneen, että hei, haluaisin, että organisaatio viestisi vahvemmin sitä, että meillä on lupa tehdä asioita toisin.” - Haastateltava H6

”Sanoissa joo, teoissa ei välttämättä. Kun mennään tuotantopuolelle, niin sieltä ei tulekaan tukea. Me oltaisiin tarvittu enemmän näkyvyyttä, että miten he voivat toimia. Meillä olisi ollut lainata heille hetkeksi Scrum Masteria, niin sieltä ihan johdosta kiellettiin, että ei saa, me katsotaan itse kuinka me toimitaan. Me jouduttiin piilossa ja salaa näiden ihmisten kanssa sitten itse tehdä se meidän malli.” - Haastateltava H7

”Nimenomaan ketteryyttä? Ehkä joo ja ei. En tiedä mitä he voisi tavallaan tehdä. Jotain hienoja puheenvuoroja, tehdä jotain hienoja dioja?” - Haastateltava H3

Vastausten perusteella eri yksiköiden johtajat joko tukevat tai eivät tue ketteryyttä. Ylemmän johdon tuki ketteryydelle ei ole datan perusteella saavuttanut kehitystiimejä. Kysyttäessä, tukeeko organisaatio sellaista ajattelua, että tiimien tulisi toimia ketteryuden periaatteiden mukaisesti, mutta viitekehityksen saisi itse valita, haastateltava vastasi:

”Se on selkeästi aiheuttanut tiimissä ahdistusta. Tiimissä on todettu, että tämä Scrumin joku asia ei toimi meillä ja kun on yrittänyt sanoa, että tehdään toisin, no ei voida, kun meiltä edellytetään Scrumia. Et se jotenkin vähän liian vahvasti tuli se Scrum painotteisuus ja tavallaan se kontekstimalli” - Haastateltava H6

Vahva ketterän menetelmän viitekehityksen painottaminen ja johdon tuen puute ketteryydelle voisi johtua siitä, että organisaatiossa ei ole ymmärretty, mitä ketteryys tarkoittaa ja mitä se pitää sisällään. Tätä tulkintaa vahvistaa kahden haastateltavan vastaukset:

”Välillä tuntuu, että heille [JOHDOLLE] on myyty ketteryys ihan sanana ja ketteryyttenä. Toivoisin, että ketteryydestä ei puhuttaisi ketteryyttenä, vaan siitä puhuttaisiin enemmän, että me halutaan olla varmoja, että me tehdään vaan sitä tärkeintä asiaa. -- halutaan sanella, että kuinka sitä tehdään ja sehän menee jo ketteryuden vastaiseksi hommaksi” - Haastateltava H7

”Niin kauan kuin asia ei lähde valumaan ylhäältä alaspäin, sanotaan nyt, että Kelatasoisesti ihan oikeasti ymmärretään mitä testausautomaatio on, mikä sen merkitys on, miten paljon siitä voi olla hyötyä ja siihen resursoidaan -- niin ei se toimi täällä kenttätasollakaan. Testausautomaatio sivuutetaan aina siinä vaiheessa, kun tulee vähänkään kiire.” - Haastateltava H5

Haastateltava puhuu testausautomaation kohdalla johdon tuen puutteesta, mikä on teknisessä mielessä keskeinen osa ketteryyttä. Johdon tuen puute ketteryydelle ja testausautomaatiolle ja paikoittain jopa ketteryuden vastustaminen osoittaa, että organisaatiossa tarvittavia muutoksia ei ole tehty tarpeeksi, jotta ketteryys mahdollistuisi. Tämä muodostaa empiirisen johtopäätöksen EC41.

EC41: Johdon tuen puute ketteryydelle ja testausautomaatiolle osoittaa, että organisaatio ei ole tehnyt merkittäviä muutoksia toimintoihinsa, jotta ketteryys aidosti mahdollistuisi. (N)

Yhteenvedona ihmiskeskeisyyden periaatteeseen löytyi yhteensä 9 empiiristä johtopäätöstä. Näiden yhteenvedojen pohjalta muodostettiin ensisijainen empiirinen johtopäätös PEC3.

PEC4: Organisaatiossa tehdään paljon yhteistyötä, mutta tiimejä pitäisi enemmän kannustaa ja tukea kohti itseohjautuvuutta. Tiimeille pitäisi antaa aito valta ja tuki toimia ketterällä toimintamallilla.

Ihmiskeskeisyyden empiiriset johtopäätökset ovat esiteltyinä seuraavassa taulukossa (TAULUKKO 9).

TAULUKKO 9 Ihmiskeskeisyyden empiirisen johtopäätökset

Tunniste	Empiirinen johtopäätös
EC33	Yhteistyötä tehdään laajasti yli tiimi- ja yksikkörajojen, kaikki tiimit eivät kuitenkaan tee yhteistyötä muiden tiimien kanssa ja paikoitellen yhteistyö tiimin sisällä tai tiimien välillä ei toimi. (P)
EC34	Tiimien voimaantumisen esteenä on linjajohto ja sen ymmärtämättömyys omien päätöksentekojensa vaikutuksista ketteriin tiimeihin.
EC35	Kaikki tiimit eivät uskalla ja joissain tiimillä ei ole valtaa tehdä päätöksiä itse. Tiimit tarvitsisivat tukea voimaantumisessa. (P)
EC36	Organisaation tiimit ovat pääsääntöisesti tiimejä, joista löytyy tarvittava määrittely-, ohjelmointi- ja testausosaaminen ohjelmistokehitystä varten, mutta koska tiimeillä on myös ylläpitovastuita, tiimistä ei löydy kaikkea tarvittavaa osaamista. (P)
EC37	Pääsääntöisesti tiimit koostuvat tarkkarajaisista rooleista ja tiimit eivät ole aina valtuutettu tekemään päätöksiä itse. (P)
EC38	Organisaatiossa tiimeillä on pääsääntöisesti hyvin monipuolista, säännöllistä ja henkilökohtaista vuorovaikutusta, joissain yksittäisissä tiimeissä vuorovaikutus ei toimi. (L)
EC39	Scrumien scrum on organisaatiossa käytössä vain muutamalla tiimillä ja kaikilla tilaisuus ei vastaa sisällöllisesti alkuperäistä tarkoitustaan. (P)
EC39	Hajautettujen tiimien hallintaan löytyy tarkoituksenmukaiset työkalut, jotka tukevat tiimien viestintää. (F)

EC40	Maantieteellisesti hajautetut tiimit ja etätyö eivät rajoita ketterän tiimin toimintaa, mutta joissain tilanteissa siinä menetetään ketterälle kehittämiselle optimaalinen vuorovaikutus eri ohjelmoijien välillä. (P)
EC41	Johdon tuen puute ketteryydelle ja testausautomaatiolle osoittaa, että organisaatio ei ole tehnyt merkittäviä muutoksia toimintoihinsa, jotta ketteryys aidosti mahdollistuisi. (N)
PEC4	Organisaatiossa tehdään paljon yhteistyötä, mutta tiimejä pitäisi enemmän kannustaa ja tukea kohti itseohjautuvuutta. Tiimeille pitäisi antaa aito valta ja tuki toimia ketterällä toimintamallilla.

4.5 Tekninen huippuosaaminen

Tekninen huippuosaaminen on kypsyystasomallin neljäs periaate. Tähän alalukuun on koottuna teknisen huippuosaamisen periaatteeseen liittyviä havaintoja, ja havainnot ovat jaoteltuna tarkempiin arvioitaviin käytänteisiin.

4.5.1 Koodausstandardit

Kehittäjillä on oma yhteinen kieli ja käytännöt koodin kirjoittamisen suhteen. (Turetken ym., 2016) Kysyttäessä onko tiimillä koodausstandardeja olemassa, kaksi haastateltavista vastasi:

”Ymmärtääkseni heillä on [työkalut] mitkä tarkistavat sitä kirjoitusasua ja koodin oikeellisuutta, että kaikki tekisi saman tyylistä tekstiä” – Haastateltava H6

”Joo, kyllä meillä on.” – Haastateltava H7

”No enpä muista nähneeni.” – Haastateltava H8

Yksi haastateltava mainitsee erikseen, että tilanne on aikaisempaan verrattuna huonompi:

”Jotenkin kuvittelen, että on vähän enemmän hunningolla nyt kun oltu aikaisemmin.” – Haastateltava H3

Muita havaintoja koodausstandardeista ei löytynyt datan perusteella. Tähän voi liittyä se, että haastateltavat eivät tiedä, mitä asioita teknisessä mielessä tiimissä tapahtuu.

”Meillä nuo kehittäjät on aika hyvin pitänyt Scrum Masterin ulkona teknisestä työstä. Mä välillä oikein kipuilin asian kanssa.” – Haastateltava H6

”Tuosta en osaa sanoa, en ole koodin kanssa ollut tekemisissä. – Haastateltava H1

Koodausstandardit ovat käytössä, mutta tarkempaa käsitystä siitä, kuinka monella tiimillä ja kuinka laadukkaasti standardeja noudatetaan, ei ole, sillä haastateltavat eivät ole tietoisia tiimin koodauskäytännöistä. Tämä muodostaa empiirisen johtopäätöksen EC42.

EC42: Koodausstandardit ovat organisaatiossa joillakin tiimeillä käytössä, mutta käytön laajuudesta ei ole käsitystä, sillä tiimin koodauskäytännöt eivät ole välttämättä koko tiimille avoimesti näkyvillä. (P)

4.5.2 Tiedon jakaminen

Käytössä on tiedonjakamiseen liittyviä työkaluja, jotka auttavat dokumentoimaan ja ylläpitämään informaatiota ja jakamaan tietoa tiimin jäsenten välillä. (Turetken ym., 2016) Kysyttäessä tarkemmin nimeltä, onko tiimillä käytössä eri viestintävälineitä, haastateltava vastasi:

”Joo, kaikki nuo on käytössä. Et tavallaan ei ole jäänyt välineestä kiinni kommunikointi etänäkään.” - Haastateltava H1

”On eri välineitä, on tosiaan tämä [TYÖVÄLINE], on Raketti [ROCKET.CHAT], on sähköposti, niin kyllä niiden kanssa pystyy hyvin pelaamaan. - Haastateltava H5

”Raketti on ihan keskeinen siinä tänä päivänä -- Se on tosi keskeinen, sit on tietysti ihan kokouksia, workshoppeja ja tällaisia tehdään ja sitten tietysti kahdenkeskisiä palavereja ja chattailua” - Haastateltava H6

”Meillä on Teams tosi tehokkaassa käytössä” - Haastateltava H7

Haastateltavien vastausten perusteella tiimissä on riittävät työkalut tiedon jakamiseen. Ongelmaksi saattaa koitua se, löytyykö oikea tieto, kun tiimillä on niin monta eri työkalua, jolla informaatiota voi jakaa:

”Tiedän kyllä missä on [YKSIKÖN] portfolio Confluencessa, mutta en missä ne kuvaukset on tai missä niitten pitäisi olla” - Haastateltava H4

Tiimeillä on käytössä tiedonjakamiseen liittyviä työkaluja, joilla informaatiota dokumentoidaan ja ylläpidetään hyvin laajasti. Tämä muodostaa empiirisen johtopäätöksen EC43.

EC43: Tiimeillä on käytössä tiedonjakamiseen liittyvät työkalut, joilla informaatiota dokumentoidaan ja ylläpidetään. (F)

4.5.3 Vapaaehtoisuus tehtävissä

Tiimin jäsenet osallistuvat tehtävien tekemiseen vapaaehtoisesti, eivätkä esihenkilöt päättää, kuka tekee minkäkin tehtävän. Jos vapaaehtoisia tehtävän suorittamiseen ei ole, tiimi vastaa kollektiivisesti tehtävän suorittamisesta. Tällä käytännöllä lisätään työtyytyväisyyttä, motivaatiota sekä suorituskyvyn laatua.

(Turetken ym., 2016) Kysyttäessä, miten päätetään tehtävien jako tiimissä, yksi haastateltava vastasi seuraavasti:

”Se menee ihan puhtaasti tiimin kesken keskustellen, joko dailyissa tai Rocketchat -kanavalla.” – Haastateltava H1

”Story kuuluu kokonaisuudessaan tiimille -- tyypillinen tapahan on se, että jollekin se assignoidaan tai joku assignoi sen itselle ja alkaa tehdä sitä” – Haastateltava H8

Yksi havainto on siitä, että toisinaan tehtäviä kiinnitetään liian aikaisessa vaiheessa jo yksittäisille ihmisille.

”Aika pitkälti se tulee sinne sprinttisuunnittelussa monen tehtävän osalta. Välillä siitä vähän kritiikkiä nouseekin tiimissä, että liikaa laitetaan jo sprintti suunnittelussa ihmisille kiinni tehtäviä.” – Haastateltava H6

Epäselväksi jää tässä tapauksessa, kuka tehtävät kiinnittävät henkilöille. Esihenkilöt eivät osallistu sprintin suunnitteluun, joten voisi olla niin, että esimerkiksi Scrum Master tai joku muu tiimin sisältä valitsee tehtävät etukäteen. Havaintoja siitä, että esihenkilö päättäisi, kuka minkäkin tehtävän tekee ei ole. Myöskään siitä ei ole havaintoja, vastaako tiimi kollektiivisesti tehtävän suorittamisesta, jos vapaaehtoisia ei löydy. Tiimeissä tehtävät jaetaan aika pitkälti tiimin jäsenten kesken, eikä ulkopuolisten tahojen tarvitse puuttua siihen, kuka tehtävän suorittaa. Tämä muodostaa empiirisen johtopäätöksen EC44.

EC44: Tiimin jäsenet osallistuvat tehtävien tekemiseen vapaaehtoisesti ja ulkopuolisten ei tarvitse puuttua siihen, kuka tehtävän suorittaa. (L)

4.5.4 Hyväksymistestaus

Hyväksymistestauksessa varmistetaan, että järjestelmä toimii käyttäjätarinan mukaisesti. Testejä automatisoidaan aina kuin mahdollista. (Turetken ym., 2016) Yksi haastateltavista kertoi, että heillä on hyväksymistestaus käytössä:

”Sprintin päätteeksi, kun saadaan asiat valmiiksi, tuoteomistaja tekee sellaisen hyväksyntätestauksen sille. – Haastateltava H7

Huomioitavaa tämän käytänteen arvioinnissa on se, että hyväksymistestaus perustuu käyttäjätarinoiden käyttöön. Käyttäjätarinoiden käyttäminen arvioidaan tulevassa luvussa 4.7.1. Käyttäjätarinat. Hyväksymistestauksen käyttämisestä on hyvin vähän todisteita. Tätä voi selittää se, että käyttäjätarinoita käytetään organisaatiossa vain joissain tiimeissä, jolloin käyttäjätarinoiden hyväksymistestaus ei ole käytössä kuin yksittäisillä tiimeillä. Tämä muodostaa empiirisen johtopäätöksen EC45.

EC45: Käyttäjätarinoiden hyväksymistestausta tehdään organisaatiossa hyvin vähän. (N)

4.5.5 Sovelluskokoonpanon hallinta

SCM (engl. *software configuration management*) -työkalut auttavat sovelluskehityksessä versionhallinnassa. (Turetken ym., 2016) Kysyttäessä, onko tiimillä versionhallintatyökalu olemassa, haastateltavat vastasivat:

”Me käytetään sitä BitBucketia.” - Haastateltava H8

”Joo, mutta me ollaan eri työkaluissa kuin muut.” - Haastateltava H7

Tarvittavat työkalut sovelluksen versionhallintaan löytyy kahden haastateltavan perusteella. Tämä muodostaa empiirisen johtopäätöksen EC46.

EC46: Tiimeillä on tarvittavat sovelluksen versionhallintatyökalut käytössä. (F)

4.5.6 Automaatiotestaus

Automaatiotestausta tulisi hyödyntää aina kuin mahdollista esimerkiksi yksikkö- ja hyväksymistesteissä. (Turetken ym., 2016) Kysyttäessä haastateltavilta, onko tiimillä käytössä automaatiotestausta, haastatteliija vastasi:

”On alettu kehittämään automaattitestejä” - Haastateltava H1

”Nyt ollaan juuri tekemässä testiautomaatiota elikkä me ollaan ottamassa sitä mukaan, mutta se on hyvin alussa vielä” - Haastateltava H7

”Ei pysty, on vanhat systeemit. Ei pysty oikein mitään automaattitestausta käyttää.” - Haastateltava H8

Tarve automaatiotestauksella olisi, sillä järjestelmiä testataan paljon manuaalisesti.

”Voisi käyttää paljon enemmän teknologioita, kun niitä on ja myös testausautomatisointia. Täällä on hirvittävästi manuaalitestausta ympäri Kela” - Haastateltava H7

Kaksi haastateltavaa nostaa esille, että resursoinnin takia automaatiotestausta ei tehdä niin kattavasti kuin tiimeillä olisi halua:

”Meillä on koko aika ollut pahasti aliresursoituna automaatiotestien rakentaminen.” - Haastateltava H6

”Ei vaan kerta kaikkiaan ole resursseja, että se tiedostetaan. Testaajien puolelta on tullut useampaan otteeseen, että he näkevät ja kokevat sen, että se [AUTOMAATIOTESTAUS] olisi hyvä” - Haastateltava H5

Automaatiotestaukseen ei riitä resursseja, mikä saattaisi kertoa siitä, että automaatiotestaaminen ei ole niin tärkeä asia kuin jokin muu, esimerkiksi

liiketoiminnallisten ominaisuuksien kehittäminen. Kysyttäessä, näkyykö se, että organisaatio tukee automaatiotestausta, haastateltavat vastasivat:

”Tuetaan sitä kautta, että siellä on se asiantuntijatiimi keneltä saa tosi hyvin tukea. -- Eihän sille kiinnostusta paljon muualta tule.” - Haastateltava H6

”Eihän tämä ole missään kokonaisarkkitehtuurissakaan. Ei näy missään niin kuin suoranaisesti, että testausautomaatio on yhtenä osana tässä ja sille luotaisiin resursseja tai mahdollisuuksia.” - Haastateltava H5

Haastateltavien vastausten perusteella organisaatio ei anna automaatiotestaukselle tarpeeksi tukea. Organisaation tuen puute näkyy myös siinä, että sovelluskehitystiimit ovat itse yrittäneet implementoida automaatiotestauksen osaksi ohjelmistokehitysprosessia, mutta se on estynyt nimenomaan resurssien puutteiden ja kulttuurin takia.

”Yritin saada sitä asiaa eteenpäin, että testausautomaatio on osa normaalia sovelluskehitystä ja sen pitää nivoutua siihen sovelluskehitysprosessiin. Aluksi oli ihan suoranaista vähättelyä ja muutosvastarintaa, mutta enemmän ja vähemmän vaikeiden vaiheiden kautta ollaan nyt nykytilanteessa, joka mielestäni on valoisampi.” - Haastateltava H5

Tässä tapauksessa yksittäinen henkilö on saanut muutettua yksittäisen tiimin toimintatapaa siihen suuntaan, että automaatiotestaus huomioitaisiin paremmin osana sovelluskehitysprosessia. Samaa yritystä on ollut myös toisella haastateltavalla:

”Me yritettiin aloittaa kehittäjien opastamista automaatiotestien ylläpitoon ja tekemiseen, mutta siihen ei vaan ole ollut aikaa.” - Haastateltava H6

Tuen puute saattaa näkyä esimerkiksi yksittäisissä päätöksenteoissa. Kysyttäessä tarkemmin, miksi uuden opetteluun ei löytynyt tässä tapauksessa aikaa eli mikä nähtiin tärkeämpänä asiana, haastateltava vastasi:

”Tuoteomistajan yläpuolelta sekä IT:n linjaorganisaatiossa ylemmät tahot että liiketoiminnan organisaatiossa ylemmät tahot ovat yhdessä sitä mieltä, että meidän pitää tietyllä aikataululla saada tuotantoon julkaistava, toimiva järjestelmä. Ei niitä kiinnosta, onko siellä automaatiotestejä vai ei.” - Haastateltava H6

Haastateltavien vastauksista voisi tulkita kaksi asiaa: ensinnäkin organisaatiossa ei ole sellaista toimintakulttuuria, jossa automaatiotestaaminen on erottamaton osa ohjelmistokehitysprosessia ja toiseksi organisaatio ei mahdollista automaatiotestaamisen ottamista osaksi toimintamallia. Tämä muodostaa empiirisen johtopäätöksen EC47.

EC47: Organisaation toimintakulttuurissa automaatiotestaaminen ei ole erottamaton osa ohjelmistokehitystä ja tukea organisaatiokulttuurin muuttamiseksi ei ole.

Syy siihen, miksi yksikkö- ja automaatiotestejä jää tällä hetkellä tekemättä, johtuu osittain linjajohdon paineesta. Toisaalta, jos automaatiotestaus olisi erottamaton osa ohjelmistokehitystä, automaatiotestit eivät jäisi välttämättä tekemättä, paineesta huolimatta. Yhteenvetona automaatiotestauksesta haastateltavien vastausten perusteella organisaatiossa on automaatiotestausta käytössä, mutta automaatiotestauksen laajempi käyttö vaatisi sitä, että toiminnalle olisi vahvempi johdon tuki ja tuki organisaatiokulttuurin muuttamiselle. Tämä muodostaa empiirisen johtopäätöksen EC48.

EC48: Organisaatiossa on automaatiotestausta jonkin verran käytössä, mutta laajempi automaatiotestauksen käyttö organisaatiossa vaatisi vahvempaa johdon tukea sekä tukea organisaatiokulttuurin muuttamiseen. (P)

4.5.7 Iteraation edistymisen seuranta

Tiimillä on mahdollisuus mitata kehitystyön edistymistä jokaisen iteraation aikana. (Turetken ym., 2016) Kaikilla kehitystiimeillä on mahdollista käyttää Jiraa työnohjaustyökaluna ja Jirassa työn edistymistä voi mitata erilaisin metriikoin, jotka hyödyntävät joko story pointeja tai tehtävän määriä (Atlassian, 2022).

”Koitan trendejä haistella säännöllisesti, onko meidän vauhti hiipumassa ja mistä se voisi johtua.” – Haastateltava H1

”Dailyt ja siellä käydään tilanne läpi, että millä tavalla me tänään mennään kohti niitä meidän sprintin tavoitteita eli siellä ei käydä edes henkilöitä läpi vaan siellä käydään ne asiat läpi” – Haastateltava H8

”Oletan, että pystyisin Jirasta seuraamaan [TÖIDEN EDISTYMISTÄ]. Välillä käyn jotain katsomassa, mutta en ehkä osaa Jiraa riittävän hyvin käyttää.” – Haastateltava H3

Tiimeillä on mahdollisuus mitata kehitystyön edistymistä jokaisen iteraation aikana. Tämä muodostaa empiirisen johtopäätöksen EC49.

EC49: Tiimeillä on mahdollisuus mitata kehitystyön edistymistä jokaisen iteraation aikana. (L)

4.5.8 Ei isoa suunnittelua etukäteen

Tuotetta kehitetään evolutiivisella tavalla eli kehitystä tapahtuu koko kehitysprosessin ajan. Vesiputousmallisessa kehitysprosessissa ensin suunnitellaan iso määrä tekemistä, jonka jälkeen vasta ohjelmoidaan ja testataan. Tätä kuvataan usein termillä Big Design Up Front eli BDUF. Ketterässä suunnittelussa suunnittelu, ohjelmointi ja testaus kulkee koko kehitysprosessin ajan. (Turetken ym., 2016) Yhdellä tiimillä tehtävät suunnitellaan lähtökohtaisesti niin, että kehitystehtävä ei valmistu ennen kuin esimerkiksi

yksikkötestit on tehty ja sprintit pyritään suunnitella siten, että sprintin päätteeksi valmistuisi jotain, mitä voisi demonstroida.

”Suunnittelupäivänä me suunnitellaan sprintit alustavasti ja viimeksi me taidettiin kirjoittaa sprinttien tavoitteetkin, että me haluttaisiin, että tällä sprintillä valmistuisi nämä asiat.” - Haastateltava H7

”Meillä on Definition of Done eli meillä pitää tietyt kriteerit täytyä, sanotaan, että yksikkötestikattavuus ei saa laskea” - Haastateltava H7

”Me pyritään aina, että meillä olisi jotain demottavaa. - Haastateltava H7

Tiimi saa tuotettua lyhyen iteraation aikana toimivaa ohjelmistoa, mikä on testattu, joten ainakin tämän tiimin tekeminen on ketterien menetelmien mukaista. Myös toisen haastateltavan vastaus tukee sitä tulkintaa, että nykyisin kehittäminen on ketterien menetelmien mukaista, vaikka muutos ketterään kehittämiseen on ollut iso.

”Alussa joku kehittäjä oli huolissaan, että ennen tuli speksit ja hän sitten koodasi. Nyt joudutaan miettimään, miten se tehdään ja määrittelemään ja koodaamaan ja joutuu jopa testausta suunnittelemaan. -- Muutamia henkilövaihdoksia on ollut -- Sit se yhteistyö lähti menemään. Ne [OHJELMOIJAT] piti sitä hyvänä, että eiköhän tehdä näin.” - Haastateltava H8

Toimintamallin muutos oli saatu tässä tiimissä aikaan, mutta kaikissa tiimeissä tilanne ei ole vielä sama:

”Tämän ketterän kehittämisen rinnalla täytyy välillä tapauskohtaisesti soveltaa pikkuisen vesiputousmalliakin jopa, että homma toimii.” - Haastateltava H5

”Täällä on perinteisesti tehty hirveän tarkkoja kuvauksia, vaikka määrittelyyn, että tehdään vähän varastoon. Ja tehdään hirveän tarkalla tasolla, ennen kuin on annettu toteuttajille kommentoitavaksi. Yleisfiilis on, että se on vielä aika tiukassa tämä vanha ajattelu. Mutta kyllä jossain on jo onnistuttu sitä vähän madaltamaan sitä tasoa” - Haastateltava H2

”Ennen kuin on annettu toteuttajille kommentoitavaksi” viittaisi siihen, että vaatimusmäärittelyä annetaan kuitenkin ohjelmoijalle nähtäväksi ja kommentoitavaksi ennen kuin määrittelyt ovat täysin valmiita. Vaikka tuotteen kehitys ei ole täysin ketterien menetelmien mukaista, haastateltavien vastauksista voisi tulkita sen, ettei tuotteen kehitys ole myöskään vesiputousmallista tutun BDUF-ajattelutavan mukaista. Tämä muodostaa empiirisen johtopäätöksen EC50.

EC50: Tuotetta kehitetään entistä evolutiivisemmalla tavalla ja paikoittain yksittäiset tiimit ovat omaksuneet käytänteen hyvin. (L)

4.5.9 Tuotteen kehitysjojo

Tuotteen kehitysjonon (engl. *backlog*) on tulevien töiden varasto. Kehitysjojoa voidaan hyödyntää eri tarkkuustasoilla, mm. tiimi, ohjelma ja portfoliotasoilla. (Turetken ym., 2016) Kysyttäessä, onko tiimillä tai tiimeillä tuotteen kehitysjojo käytössä, kaksi haastateltavaa vastasi:

”Kyllä, joo.” - Haastateltava H6

”Meillä on joku 75 tiimiä -- Pitäisikö meillä olla vähän parempi käsitys siitä, että kuinka monella olisi kunnossa tuollainen asia? - - Ehkä yli puolet on” - Haastateltava H2

Monella tiimillä tuotteen kehitysjojo on käytössä. Ainakin osalla haasteena on tuotteen kehitysjonon pitäminen ajan tasalla.

”Meillä on backlogilla vanhoja ihan suoranaisia bugejakin suuria määriä ollut, niin jossain vaiheessa olen sanonut, ettei näitä kuulkaa ikinä tulla korjaamaan tällä menolla. Me on saatu varmaan parisataa vanhaa tikettiä rejectoitua sieltä.” - Haastateltava H8

”Sanotaanko yhdeksänkymmentä prosenttia niistä tänä päivänä varmaan on ajan tasalla.” - Haastateltava H6

”Pystyy katsomaan toisten tiimien Jiraa, vaikka backlogia, että mitä siellä on.” - Haastateltava H1

Tiimitasolla tuotteen kehitysjonot ovat melko kattavasti ja laadukkaasti käytössä, mutta tiimistä ylemmillä tasoilla tuotteen kehitysjonon käyttö on vajavaista.

”Monestikin saatan tietää, että tässä on samalla asialla muitakin, miksei ne ole linkannut itsensä tähän Portfolio Epiciin tai sieltä ei löydy paljon mitään sieltä Portfolio Epistä, että mihin tämä nyt liittyy.” - Haastateltava H8

”Tiedän kyllä missä on [NIMI] portfolio, mutta en missä ne kuvaukset on tai missä niitten pitäisi olla. Eli pitäisi olla selkeämpää tai jotenkin läpinäkyvämpää, helpommin ymmärrettävää, että missä siellä mennään” - Haastateltava H4

Haastateltavat puhuvat siitä, että kehityksen aikainen tietojen päivittäminen jää tekemättä. Tästä voisi päätellä sen, että jos tiedot eivät ole ajan tasalla kehitysvaiheessa, missä työtä jo tehdään, toimiiko tuotteen kehitysjojokaan aidosti tulevien töiden varastona? Eli tuotteen kehitysjojo saattaa olla teknisesti olla olemassa, mutta sisällöllisesti tuotteen kehitysjojoa ei käytetä ylemmillä organisaation tasoilla tulevien töiden varastona tai käytössä on puutteita. Tiimitasolla tuotteiden kehitysjonot ovat melko laajasti käytössä. Tämä muodostaa empiirisen johtopäätöksen EC51.

EC51: Tiimitasolla tuotteen kehitysjonot ovat melko laajasti käytössä, ylemmillä tasoilla kehitysjojojen käytössä on puutteita. (L)

4.5.10 Jatkuva integraatio

Tiimin jäsenten työt integroidaan mahdollisimman usein. Jokainen integraatio olisi hyvä varmentaa automatisoidulla rakennustyökalulla, jotta mahdolliset integroimisvirheet huomataan mahdollisimman nopeasti. (Turetken ym., 2016). Kysyttäessä miten tiimin jäsenten töiden integroiminen hoidetaan, haastateltava vastasi:

”Näenhan mä sieltä tiketeiltä, että siellä on committeja tyypillisesti enemmän kuin yksi eli mitenkä mahtavat sen tehdä ryhmätyönä? Mä en osaa sanoa miten sinne pääsee koko Gitiin” - Haastateltava H8

”Meillä nuo kehittäjät on aika hyvin pitänyt Scrum Masterin ulkona teknisestä työstä.” - Haastateltava H6

Tarkempaa tietoa siitä, miten ohjelmoijien työt integroidaan keskenään ei datassa ole. Tähän voi liittyä se, että kaikki haastateltavat henkilöt eivät ole koodin kanssa tekemisissä, tekninen ymmärrys puuttuu tai roolinsa vuoksi henkilö jätetään teknisten toimintatapojen ulkopuolelle.

Tätä käytäntöä arvioidessa tulisi huomata, että jatkuva integraatiota käytetään usein DevOps-toimintamallin yhteydessä. Tulevassa alaluvussa 4.7.4. DevOps arvioidaan, että DevOps-toimintamalli ei ole organisaatiossa vielä laajasti käytössä. Toinen huomioitava asia on yksikkötestien käyttö. Käytäntö arvioidaan tulevassa alaluvussa 4.6.12. Yksikkötestit, ja arvioinnin lopputulemana on se, että yksikkötestejä käytetään jonkin verran. Kolmas huomioitava asia on automaatiotestauksen käyttö: organisaatiossa automaatiotestausta käytetään harvakseltaan. Tarkemmat perustelut johtopäätökselle löytyvät tulevasta aliluvusta 4.6.6. Automaatiotestaus.

Vaikka tarkempaa dataa jatkuvasta tuottamisesta ei löytynyt, jatkuva integraatio ei voi olla kovin laajasti organisaatiossa käytössä, sillä DevOps-toimintamallin, yksikkötestien ja automaatiotestauksen käyttö ei ole yleistä. Kuitenkin jotkut tiimit hyödyntävät DevOps-toimintamallia, yksikkö- ja automaatiotestausta, mikä voisi osoittaa sen, että jatkuva integraatio olisi käytössä yksittäisillä tiimeillä. Tämä käytäntö jätetään arvioimatta sillä perusteella, että dataa aiheesta ei ole riittävästi.

4.5.11 Jatkuva parantaminen (refaktorointi)

Refaktoroinnilla tarkoitetaan koodin uudelleenkirjoittamista siten, että sisällöllisesti koodi pysyy samana, mutta rakenteellisesti koodin rakennetta parannetaan ja on sitä kautta jatkossa ylläpidettävämpi. Yleensä refaktorointi keskittyy koodin päällekkäisyyden poistamiseen. (Turetken ym., 2016) Neljän vastauksen perusteella koodia parannellaan jatkuvasti. Kysyttäessä, refaktoroidaanko koodia, haastateltavat vastasivat:

”Mielestäni joskus tosi paljon” - Haastateltava H8

”Hirveän paljon meillä on tehty refaktorointia, että yritetään noudattaa eri osissa järjestelmää samanlaista tyyliä, et on tavallaan refaktoroitu, että olis koko järjestelmä samalla tavalla rakennettu.” – Haastateltava H6

”Ollaan sovittu, että jos on joku pieni juttu mikä voidaan siinä paikassa korjata samalla, vaikka se ottaisi vähän kauemmin se homma. Se kuitenkin nopeuttaa jatkossa.” – Haastateltava H7

”Osittain teknisen velan kautta sitä varmasti joudutaan tekemään.” – Haastateltava H2

Koodin jatkuva parantamista tehdään paljon, mikä voi johtua siitä, että osa järjestelmästä on vanhoja ja ne on rakennettu tavalla, joita voi olla vaikea ylläpitää ja kehittää. Tätä tulkintaa haastateltavan vastaus:

”Ehkä vähemmän on semmoisia tapauksia, että ihan päin seiniä tehty. Joskus semmoisiakin on. Kyllä se on joskus tehty sen ajan parhaan käsityksen mukaan ja sitten se korjataan” – Haastateltava H8

Jatkuvaa koodin parantamista tapahtuu paljon. Tämä muodostaa empiirisen johtopäätöksen EC52.

EC52: Jatkuvaa koodin parantamista tiimeissä tehdään paljon. (L)

4.5.12 Yksikkötestit

Yksikkötesteillä varmistetaan, että lähdekoodin yksittäiset osat toimivat oikein. Suositeltavaa on, että yksikkötestit on automatisoitu. (Turetken ym., 2016) Kysyttäessä, onko tiimeillä yksikkötestit käytössä, kolme haastateltavaa vastasi:

”Olen ymmärtänyt näin, että ei kaikilta osin ole niin laajasti kuin mitä pitäisi olla. Joissain osa-alueissa on kai kohtuullisen hyvinkin.” – Haastateltava H5

”Tällä hetkellä ne siellä ymmärtääkseni aika kattavasti on käytössä.” – Haastateltava H6

”Jotain pientä yksikkötestausta, mutta voi sanoa, että ei ole.” – Haastateltava H8

Havaintoja yksikkötestien käyttämisestä on kolmen haastateltavan perusteella. Yksi haastateltava nosti erikseen esille, että vaikka yksikkötestit ovat kattavasti käytössä, ne on rakennettu ohjelmistoon jälkikäteen.

”Tuli se pilotti tuotantokäyttöön, niin meillä ei ollut yksikkötestejä. Et se oli teknistä velkaa, josta silloin me jouduttiin karsimaan taas aikataulusyistä pois, että me saatiin yleensäkin järjestelmä käyttöön.” – Haastateltava H6

Tästä voisi tulkita sen, että yksikkötestien tekeminen ei ole rakennettu osaksi koko organisaation ohjelmistokehitystä vaan testien tekeminen on irrallinen osa laadunvarmistusta. Toisen haastateltavan vastaukset tukevat tätä tulkintaa:

”Meillä on Definition of Done eli meillä pitää tietyt kriteerit täytyä, vaikka sanotaan, että yksikkötestikattavuus ei saa laskea” – Haastateltava H7

Kysyttäessä, ovatko laatustandardit tulleet tiimin sisältä, haastateltava vastasi:

”Kyllä. Olettaisin, että ne tulisi jostain linjauksista ehkä, mutta ne linjaukset on muutenkin hankalia kun ne tulee ulkopuolelta tavallaan. – Haastateltava H7

Tässä kyseisessä tiimissä yksikkötestit on saatu rakennettua osaksi ohjelmistokehitysprosessia, mutta tiimi on ne itse itseltään ne sinne vaatineet. Eli riippuen tiimistä ja itseohjautuvuudesta yksikkötestit on saatu rakennettua osaksi tiimin toimintakulttuuria. Havaintoja systemaattisesta yksikkötestien käytöstä on vain yhden tiimin kohdalta, kahden tiimin osalta yksikkötestit ovat käytössä, mutta yksikkötestien käyttöä ei ole rakennettu osaksi tiimien ohjelmistokehitysprosessia. Tästä voisi tulkita sen, että yksikkötestien tekeminen ei ole osa koko organisaation ohjelmistokehitysprosessia ja -kulttuuria. Tämä muodostaa empiirisen johtopäätöksen EC53.

EC53: Yksikkötestejä käytetään jonkin verran, mutta yksikkötestien tekeminen ei ole osa koko organisaation ohjelmistokehitysprosessia ja -kulttuuria. (P)

4.5.13 30% tason 2 ja tason 3 ihmisiä

Cockburnin mukaan henkilön kokemuksen määrä alalla liittyy henkilön ymmärtämistasoon. Ymmärtämistasoja on Cockburnin mukaan kolme. Turetkenin kyspysymallin kolmannella tasolla tarvitaan tason 2 ja tason 3 osaajia. (Turetken ym., 2016) Tarkemmat määritelmät tasoille löytyvät alaluvusta 2.6.3. Cockburnin henkilöstötasot. Kysyttäessä, minkä tasoista osaamista ohjelmoijilla on, haastateltavaa vastasivat:

”Meillä on ehkä moninaista, meillä on jopa semmoista, jotka ei ole kehittynyt niin paljon. Tuntuu, että ne ei ikinä pääse sinne medium-tasolle jostain syystä. Meillä on enemmän medium-tasoa, mutta sitten meillä on pari semmoista huippuosaaja.” – Haastateltava H7

”Koodarit on kaikki jo kohta kaksi vuotta olleet pelkästään tässä meidän tiimissä ja jokainen on tehnyt koodaustyötä jo sitä ennen. Jos mä heidän osaamistaan [ARVIOIN], vähintään mediumtaso, osa on medium plussaa jo siellä.” – Haastateltava H6

”Mä lasken siihen osaamiseksi sen asenteen millä tavalla sitä työtä tehdään, että ei tehdä vaan ensimmäistä helpointa ratkaisua vaan mietitään oikeasti sitä, että mikä on fiksuin ratkaisu.” – Haastateltava H6

Kysyttäessä, onko ohjelmointiosaaminen ohjelmoijilla enemmän junior vai senior tasolla, haastateltava vastasi:

”Puolet ja puolet. Tästä talosta löytyy aika paljon meiltä niitä kokeneita.” – Haastateltava H3

Tiimien ohjelmoijien taso voisi kuulua Cocburnin osaamistason mukaisesti vähintään tasoille 1B – 2. Myös toisen haastateltavan vastaus tukisi tätä tulkintaa, että osaaminen on tiimissä vähintään tasolla 1B.

”Meillä on nyt ollut muutaman kuukauden harjoittelija. Hän rupesi tekemään tosi hyvää tulosta ja se jo määrittelee itsellensä, että seuraavaksi hän tekee tämmöistä ja kertoilee vaan meille muille” – Haastateltava H8

Jopa harjoittelija voisi kuulua Cocburnin osaamistason mukaisesti tasolle 1A. Vastauksista voisi tehdä sen tulkinnan, että näissä tiimeissä 30% tason 2 ja tason 3 ihmisistä täyttyy. Tason -1 määritelmään kuuluu se, että henkilö on kykenemätön tai haluton tekemään yhteistyötä tai noudattaa jaettuja menetelmiä (Cockburn, 2002; Boehm & Turner, 2003). Myös tason -1 osaajia on ainakin aiemmin ollut organisaatiossa yhden haastateltajan perusteella:

”Muutamia henkilövaihdoksia on ollut, muutama pahempi jäää ehkä lähtenyt pois.”
– Haastateltava H8

Epäselväksi jää, lähtivätkö henkilöt pois pelkästään tiimistä vai koko organisaatiosta. Vastauksen perusteella ohjelmoijien taso voisi olla siis aina tasolta -1 tasolle 2 asti. Tässä käytänteessä olennaista on, täyttääkö 30% ohjelmoijista tason 2 ja osa tason 3 kriteerit. Koska tason 3 osaajista ei ole dataa, käytäntöä ei ole saavutettu ainakaan täysin. Noin puolet ohjelmoijista voisi kuulua vähintään medium-tasolle, mikä tarkoittaa Cocburnin asteikolla tasoa 1B tai 1A. Ohjelmoijien osaamistason määrittely on jo lähtökohtaisesti vaikeaa, tämä käytäntö jätetään arvioimatta perustuen siihen, että riittävästi luotettavaa dataa aiheesta ei ole. Huomioitavaa on kuitenkin se, että osalla ohjelmoijista osaaminen ei kehity. Tämä muodostaa empiirisen johtopäätöksen EC54.

EC54: Organisaatiossa ohjelmoijien osaamistaso on melko hyvä, kaikilla ohjelmoijilla osaaminen ei kuitenkaan kehity.

4.5.14 Tarkoituksenmukainen arkkitehtuuri

Arkkitehtuuri on joukko tarkoituksenmukaisia, suunniteltuja arkkitehtonisia aloitteita ratkaisujen suunnittelun, suorituskyvyn ja käytettävyyden parantamiseksi, ja se tarjoaa ohjeita ryhmien väliseen suunnitteluun ja toteutuksen synkronointiin. (Turetken ym., 2016) Datassa arkkitehtuurinäkökulmaa on käsitelty neljässä vastauksessa siihen tyyliin, että arkkitehtuuri ei ole ollut aikaisemmin tarkoituksenmukainen, mutta sitä kohti ollaan menossa.

”Kyllä se on joskus sen ajan parhaan käsityksen mukaan ja sitten se korjataan, mutta kyllähän tämä tämmöinen voi sanoa, että himmeleistä koostuva yhdistelmä koko tämä meidän työalue täällä.” – Haastateltava H8

Haastateltava puhuu esimerkissään testausautomaatiosta ja sen huomioimisesta arkkitehtuurissa:

”Se vaatisi paremman koordinoinnin ja koordinoijan, suunnittelun ja suunnittelijan arkkitehtuuritaholle ja sieltä nähtäisiin, et jos me halutaan tätä, niin se vaatii tätä ja jotta se toimisi, niin minun näkemykseni on se, että silloin se täytyy tulla ylhäältä päin niin sanotusti annettuna, että nyt me halutaan näin” - Haastateltava H5

Esimerkiksi testausautomaatiota ei haastateltavan perusteella huomioida tällä hetkellä arkkitehtuurissa, mutta tarkoituksenmukaista se ei välttämättä ole. Järjestelmäkehityksessä ainakin jatkossa tarkoituksenmukainen arkkitehtuuri on tarkoitus huomioida entistä paremmin:

”Me saatiin ratkaisuarkkitehti, joka oli meidän teknisen arkkitehtuuriin keskittyvä ihminen ja teknisestä velasta huolissaan. Hän rakensi meille roadmapit ja toi sen asian näkyväksi” - Haastateltava H4

”Meidän tiimin sisällä on henkilö, joka voisi olla hyvin arkkitehti. -- Kyllä sen pitäisi tavallaan olla, että kaikissa tiimeissä olisi semmoinen syväosaaja” - Haastateltava H7

Ensimmäisen haastattelijan vastaus ”me saatiin” ja toisen haastattelijan vastaus ”pitäisi” voisi viitata siihen, että nykyisellään organisaatiossa arkkitehtuuriosaamista ei välttämättä ole riittävästi. Arkkitehtuuriosaamisen puute voi puolestaan johtaa siihen, että järjestelmäkehityksessä arkkitehtuuria ei pystytä kehittämään tarkoituksenmukaiseksi. Yksi haastateltava tuo esille myös toisen syyn sille, miksi arkkitehtuuri ei välttämättä ole tarkoituksenmukainen.

”Tavoite kehittämistyössä on ollut se, että pitäisi luoda kopioitavia malleja ja ratkaisuja. Kun seuraava järjestelmäkehitys alkaa, niin pystyisi tavallaan ottamaan pohjia sieltä ratkaisuista. -- Saataisiin ratkaisuja, mitkä helpottaisivat ylläpitoa ja näkyisi pitkällä tähtäimellä. Mutta ne ei ole sitä suoraan näkyvää toiminnallisuutta siinä järjestelmässä, mikä liiketoiminnan on joskus vaikea ymmärtää, että sekin on heidän parhaaksi ja eduksi, kun sitä tehdään.” - Haastateltava H6

Järjestelmän tai sovelluksen arkkitehtuuri ei ole aina tarkoituksenmukainen, mikä voi johtua ymmärtämisen ja osaamisen puutteesta. Tämä muodostaa empiirisen johtopäätöksen EC55.

EC55: Järjestelmän tai sovelluksen arkkitehtuuri ei ole aina tarkoituksenmukainen, mikä voi johtua ymmärtämisen ja osaamisen puutteesta. (P)

4.5.15 Päivittäiset edistymisen seurantakokoukset

Tiimissä iteraation edistymistä seurataan päivittäin. (Turetken ym., 2016) Kysyttäessä, onko tiimeillä dailyt käytössä, haastateltavat vastasivat:

”Aika lailla on käytössä, vaikka osa pitääkin niitä vain kerran viikossa tai kaksi kertaa viikossa.” - Haastateltava H2

"Joka aamuhan meillä on se daily" - Haastateltava H8

"Meillä on karsitut dailyt eli meillä on periaatteessa keskiviikko, perjantai dailyt ja maanantaisin meillä on weekly." - Haastateltava H6

Kysyttäessä tarkemmin, mitä daillyssä käydään läpi, haastateltavat vastasivat:

"Keskiviikko ja perjantain dailyt on perus dailyn tyyppisiä eli pyritään varttitunnissa käymään läpi se, mitä kukakin on tehnyt ja mitä tulee tekemään seuraavaksi, oliko ongelmia" - Haastateltava H6

"Käydään läpi ne tehtävät. Siinä on erittäin hyvää teknistä, nopeata keskustelua ja suunnittelua, kuka tekee mitä" - Haastateltava H8

Sisällöllisesti dailyt vastaavat tarkoitustaan eli ne toimivat edistymisen seurantakokouksina, mutta kaikki tiimit eivät pidä dailya joka päivä, joten iteraation edistymistä ei tarkastella tiiminä joka päivä. Tämä muodostaa empiirisen johtopäätöksen EC56.

EC56: Iteraation edistymistä seurataan daillyissa, mutta suurin osa tiimeistä ei seuraa iteraation edistymistä joka päivä. (P)

4.5.16 Testilähtöinen kehittäminen

Testi Driven Development eli TDD on ohjelmistokehitystekniikka, jossa testit kirjoitetaan ensin, jonka jälkeen tarvittava koodi kirjoitetaan testien läpäisemiseksi. (Turetken ym., 2016) Kysyttäessä, onko tiimeissä testilähtöinen kehittäminen käytössä, haastateltavat vastasivat:

"Ehkä meillä toistaiseksi ideaali, että siitä paljon kyllä puhutaan, useamman kerran viikossa nousee keskustelussa esiin." - Haastateltava H1

"En osaa sanoa." - Haastateltava H2

"Ei ole pystynyt olemaan. Tämmöistä kohti voitais mennä -- Mutta en usko, että varsinainen TDD ikinä tulee tähän." - Haastateltava H8

"No ei välttämättä." - Haastateltava H7

Haastateltava tarkensi, mitä hän tarkoitti vastauksellaan:

"Me ei koskaan tehdä niin, että tehdään jotain valmiiksi ja kirjoitetaan testit, jos ehditään, vaan meillä ei mikään tule valmiiksi ennen kuin ne testit on siinä." - Haastateltava H7

Eli haastateltavan vastauksen perusteella tiimillä ei ole testilähtöistä ohjelmistokehitystekniikkaa sellaisenaan käytössä, vaikka testit tuleekin tehtyä. Testilähtöisestä kehittämisestä ei noussut haastatteludatan kautta havaintoja. Yhdessä tiimissä on kiinnostusta, mutta käytännön näyttöä periaatteen

omaksumisesta ei ole. Vastauksien perusteella testilähtöistä kehittämistä tapahtuu joko erittäin vähän tai ei ollenkaan. Tämä muodostaa empiirisen johtopäätöksen EC57.

EC57: Testilähtöistä kehittämistä tapahtuu organisaatiossa joko erittäin vähän tai ei ollenkaan. (N)

4.5.17 Tason 1B tai -1 ihmisiä vähän tai ei ollenkaan tiimissä

Cockburnin mukaan henkilön kokemuksen määrä alalla liittyy henkilön ymmärtämistasoon. Ymmärtämistasoja on Cockburnin mukaan kolme. Turetkenin kypsyysmallin viidennellä tasolla tason 1B tai -1 ihmisiä tulisi tiimissä olla vähän tai ei ollenkaan. (Turetken ym., 2016) Tarkemmat määritelmät tasoille löytyvät alaluvusta 2.6.3. Cockburnin henkilöstötasot. Alaluvussa 4.6.14 30% tason 2 ja tason 3 ihmisiä arvioitiin haastattelujen perusteella ohjelmoijien osaamistasoa, mutta puutteellisen datan puuttuessa käytäntö jätettiin arvioimatta. Myös tämä käytäntö jätetään arvioimatta, sillä käytäntöä ei pysty arvioimiaan luotettavasti.

4.5.18 Rinnakkainen testaaminen

Käytäntö sisällyttää kehitystestaus-, testausautomaatio- ja testauksen suorituskäytännöt iteraation aikana. Samanaikainen testaus sisältää useita ketterän testauksen strategioita, kuten: yksikkötestaus, hyväksyntätestaus, komponenttitestaus, järjestelmä-, suorituskyky- ja luotettavuustesti. (Turetken ym., 2016) Kuten alaluvussa 4.5.9. arvioitiin, havaintoja iteraation hallitsemisesta oli vähän eli tiimit eivät saa sprintin aikana tuotettua testattua toimivaa ohjelmistoa. Näin ollen tiimit eivät saa suoritettua myöskään rinnakkaista testaamista iteraation aikana.

Yhden haastateltavan tilanne on lähimpänä rinnakkaisen testaamisen periaatetta, sillä tiimissä työn valmiin määritelmään kuuluu yksikkötestit. Eli sprintin aikana, jos jotain saadaan valmiiksi, ohjelma yksikkö- sekä hyväksyntätestataan.

"Sprintin päätteeksi, kun saadaan asiat valmiiksi, tuoteomistaja tekee sellaisen hyväksyntätestauksen sille. – Haastateltava H7

"Meillä on Definition of Done eli meillä pitää tietyt kriteerit täytyä, vaikka sanotaan, että yksikkötestikattavuus ei saa laskea – nyt ollaan ottamassa integraatiotestejä mukaan, kun päästään tiettyyn pisteeseen, niin täytyy olla integraatiotesti mukana." – Haastateltava H7

Tulevaisuudessa, ainakin tämän tiimin kohdalla, rinnakkainen testaaminen on varmasti mahdollista, sillä tiimissä ollaan ottamassa testiautomaatio paremmin käyttöön.

”Sen sijaan, että me vaan korjattaisi se bugi, nyt ollaan juuri tekemässä siihen testiautomaatiota elikkä me ollaan ottamassa sitä mukaan, mutta se on hyvin alussa vielä.” - Haastateltava H7

Havainnot rinnakkaisesta testaamisesta ei ole riittävästi haastateltavien vastausten perusteella, jotta rinnakkaista testaamista tapahtuisi organisaatiossa. Tätä voi ainakin osittain selittää se, että suurimmassa osassa sovelluskehitystiimejä DevOps-toimintamalli ei ole vielä laajasti käytössä, kuten alaluvussa 4.8.4. arvioitiin. Tämä muodostaa empiirisen johtopäätöksen EC58.

EC58: Rinnakkaista testaamista ei ole organisaatiossa käytössä, sillä tiimit eivät hallitse vielä iteraatioita ja testausautomaatio ei ole vielä laajasti käytössä. (N)

Yhteenvedona teknisen huippuosaamisen periaatteeseen löytyi yhteensä 17 empiiristä johtopäätöstä. Näiden yhteenvedojen pohjalta muodostettiin ensisijainen empiirinen johtopäätös PEC5.

PEC5: Ohjelmistokehitystä lähestytään työkalu- ja ohjepainotteisesti, jolloin yhteiset toimintatavat ja prosessit eli ohjelmistokehityskulttuuri jää kehittymättä.

Teknisen huippuosaamisen empiiriset johtopäätökset ovat esiteltyinä seuraavassa taulukossa (TAULUKKO 10).

TAULUKKO 10 Teknisen huippuosaamisen empiirisen johtopäätökset

Tunniste	Empiirinen johtopäätös
EC42	Koodausstandardit ovat organisaatiossa joillakin tiimeillä käytössä, mutta käytön laajuudesta ei ole käsitystä, sillä tiimin koodauskäytänteet eivät ole välttämättä koko tiimille avoimesti näkyvillä. (P)
EC43	Tiimeillä on käytössä tiedonjakamiseen liittyvät työkalut, joilla informaatiota dokumentoidaan ja ylläpidetään. (F)
EC44	Tiimin jäsenet osallistuvat tehtävien tekemiseen vapaaehtoisesti ja ulkopuolisten ei tarvitse puuttua siihen, kuka tehtävän suorittaa. (L)
EC45	Käyttäjätarinoiden hyväksymistestausta tehdään organisaatiossa hyvin vähän. (N)
EC46	Tiimeillä on tarvittavat sovelluksen versionhallintatyökalut käytössä. (F)
EC47	Organisaation toimintakulttuurissa automaatiotestaaminen ei ole erottamaton osa ohjelmistokehitystä ja tukea organisaatiokulttuurin muuttamiseksi ei ole.
EC48	Organisaatiossa on automaatiotestausta jonkin verran käytössä, mutta laajempi automaatiotestauksen käyttö organisaatiossa vaatisi vahvempaa johdon tukea sekä tukea organisaatiokulttuurin muuttamiseen. (P)
EC49	Tiimeillä on mahdollisuus mitata kehitystyön edistymistä jokaisen iteraation aikana. (L)
EC50	Tuotetta kehitetään entistä evolutiivisemmalla tavalla ja paikoittain yksittäiset tiimit ovat omaksuneet käytänteet hyvin. (L)

- EC51 Tiimitasolla tuotteen kehitysjonot ovat melko laajasti käytössä, ylemmillä tasoilla kehitysjonojen käytössä on puutteita. (L)
- EC52 Jatkuvaa koodin parantamista tiimeissä tehdään paljon. (L)
- EC53 Yksikkötestejä käytetään jonkin verran, mutta yksikkötestien tekeminen ei ole osa koko organisaation ohjelmistokehitysprosessia ja -kulttuuria. (P)
- EC54 Organisaatiossa ohjelmoijien osaamistaso on melko hyvä, kaikilla ohjelmoijilla osaaminen ei kuitenkaan kehity.
- EC55 Järjestelmän tai sovelluksen arkkitehtuuri ei ole aina tarkoituksenmainen, mikä voi johtua ymmärtämisen ja osaamisen puutteesta. (P)
- EC56 Iteraation edistymistä seurataan dailyissa, mutta suurin osa tiimeistä ei seuraa iteraation edistymistä joka päivä. (P)
- EC57 Testilähtöistä kehittämistä tapahtuu organisaatiossa joko erittäin vähän tai ei ollenkaan. (N)
- EC58 Rinnakkaista testaamista ei ole organisaatiossa käytössä, sillä tiimit eivät hallitse vielä iteraatioita ja testausautomaatio ei ole vielä laajasti käytössä. (N)
- PEC5 Ohjelmistokehitystä lähestytään työkalu- ja ohjepainotteisesti, jolloin yhteiset toimintatavat ja prosessit eli ohjelmistokehityskulttuuri jää kehittymättä.

4.6 Asiakasyhteistyö

Asiakasyhteistyö on kypsyystasomallin viides ja viimeinen kategoria. Tähän alalukuun on koottuna asiakasyhteistyön periaatteeseen liittyviä havaintoja, ja havainnot ovat jaoteltuna tarkempiin arvioitaviin käytänteisiin. Tämän kategorian yhteydessä on ensin määriteltävä tarkemmin, mitä asiakkaalla tarkoitetaan Kelassa, jotta käytänteitä ja niiden omaksumista olisi mahdollista arvioida. Kuten aiemmin alaluvussa 3.2 Kohdeorganisaation esittely esiteltiin, IT-palvelujen tulosityksikkö vastaa IT:n kehittämisestä, ylläpidosta sekä tuotantopalvelujen tuottamisesta Kelalle ja Kelan muille tulosityksiköille (Kela, 2022). Tämä muodostaa Kelassa sisäisen tilaaja-tuottaja-asetelman, jossa asiakasta ja joissain tapauksissa käyttäjiä edustaa tuoteomistaja. Erityisesti tämän periaatteen, asiakasyhteistyön, kohdalla nousi ilmi, että Kelassa, ainakaan kehitystiimeissä ei ole selkeää käsitystä siitä, kuka asiakas tai sidosryhmä on missäkin tilanteessa.

”Täälläkin on palveluita, joita vähän kaikki tarvitsee. Asiakkaan määrittelemisen voi olla vähän vaikeaa.” – Haastateltava H1

”Meidän asiakas on ehkä se Kelan IT himmeli eli sen IT himmelin pitää toimia joka päivä, joka hetki. Se ei henkilöidy mihinkään yksittäiseen henkilöön tai johonkin lautakuntaan tai mihinkään, se on oikeasti koko Kela.” – Haastateltava H8

”Mulla itsellä on täällä muistilappukin, että ota selvää, että kuka on tämä meidän ‘stakeholders’ kummallekin tiimille” – Haastateltava H8

Haasteita asiakkaan määrittelyyn tuo se, että asiakkaita voi olla useita ja ne voivat sijaita toisiinsa nähden hierarkkisesti "asiakkaan asiakas"-tyylillä.

"Meillä on ehkä useampia asiakkaita. Eli on monia tahoja ihan talon sisällä" - Haastateltava H1

"Kuka on asiakas, minun asiakkaani? Puhutaanko me Kelan asiakkaasta vai puhutaanko me kaikista kansalaisista, jotka käyttävät meidän asiointipalvelua ja puhutaanko meidän toimistotyön väestöstä?" - Haastateltava H3

"Meillä on tavallaan kahden tason asiakkaita" - Haastateltava H7

Ongelmaksi "asiakkaan asiakas" -tilanteessa tai tilanteessa, jossa asiakkaita on useita, nousee se, kuka asiakkaista on kaikkein tärkein? Kenen asiakasarvoa pyritään luomaan eniten?

"Mutta se, että miten ne just arvottaa, kumpi [ASIAKKAISTA] on sitten tärkeämpi" - Haastateltava H4

Yksi tiimeistä oli lähtenyt itse selkiyttämään sitä, kuka asiakas on ja kenelle kehitystiimi tekee töitä.

"Meidän kehitystiimin asiakas on meidän tuotteiden tuoteomistajat. Niin me se ajatellaan." - Haastateltava H7

"Tiimin tuoteomistaja on käytännössä meillä aika pitkälle fasilitoiva rooli" - Haastateltava H7

"Meidän tiimin asiakas on tavallaan Kelan sisäiset työntekijät" - Haastateltava H7

Tällä haastateltava tarkoittaa siis sitä, että koodaavat henkilöt tiimissä palvelevat fasilitoivan tiimin tuoteomistajan avulla tuotteiden tuoteomistajia, ja tämän tiimin palveluja käyttävät lopulta Kelan sisäiset työntekijät. Tämän perusteella arvioitaessa esimerkiksi käytäntöä "asiakas sitoutuu työskentelemään kehitystiimin kanssa", asiakkaalla viitataan yhteistyöhön nimenomaan tuotteiden tuoteomistajien kanssa, ei esimerkiksi tiimin tuoteomistajan kanssa. Tuotteiden tuoteomistajat edustavat asiakkaita yksittäisen tuotteen osalta, mutta epäselväksi jää, edustaako kukaan vai edustaako tiimin tuoteomistaja lopulta kaikkien tuotteiden asiakkaita tässä tapauksessa. Toisilla tiimeillä ei ole tällaista asetelmaa kuin tuotteen tuoteomistaja ja tiimin tuoteomistaja:

"Tiimillä ei ole tuoteomistajaa ollut tänä vuonna ollenkaan" - Haastateltava H8

Kaikilta tiimeiltä ei siis löydy edes tuoteomistajaa, joka priorisoi tehtävät sen mukaisesti, mikä kehityskohde tuottaa eniten asiakasarvoa. Tämä herättää monia kysymyksiä, kuten kuka vastaa siitä, että tuotetaan asiakasarvon kannalta tärkeimpiä asioita? Keskittyykö kehitystiimi tuottamaan asiakasarvon kannalta tärkeimpiä kohteita ja tunnustetaanko edes, mitä tärkeimmät kehityskohteet

ovat? Tämän tiimin kohdalla tuoteomistajan puuttuminen ei välttämättä ole kovin ongelmallista, sillä järjestelmä ei ole enää elinkaarensa alkupuolella.

”Me joudutaan korjaamaan sysivanhoja järjestelmiä, modernisoimaan niitä, niin me tiedetään suurin piirtein itse tiimissä, että nämä kiireellisimmät pitää tehdä ensin” - Haastateltava H8

Joka tapauksessa, ohjasi tiimin työn prioriteettia tiimin tuoteomistaja, tuotteiden tuoteomistajat tai kehitystiimi itse, on edelleenkin epäselvää, mistä tahoista puhutaan, kun puhutaan asiakasyhteistyöstä tai asiakkaasta. Kuten yläolevat tilanteet osoittavat, Kelassa on haasteita asiakkaan määrittelemisessä. Tämä muodostaa empiirisen johtopäätöksen EC59.

EC59: Kehitystiimeille ei ole selvää, kuka heidän asiakkaansa on tai kuka heidän asiakkaistaan on kaikkein tärkein.

Tämä aiheuttaa haasteita asiakasyhteistyöhön liittyvien käytänteiden, kuten ”asiakas sitoutuu työskentelemään kehitystiimin kanssa”, ”vaikutus asiakkaisiin ja toimintoihin”, ”asiakas on heti käytettävissä” ja ”asiakas sitoutuu yhteistyöhön” arviointiin ja paikoittain arviointi onkin melko mahdotonta, jos asiakasta ei pystytä määrittelemään. Kuten yksi haastateltavista totesi, asiakasarvosta vastaa kuitenkin lopulta liiketoiminnan edustaja, yleensä tuoteomistaja.

”Kyllähän siinä aika iso vastuu on nimenomaan siellä liiketoiminnalla tuoteomistajilla ja kehitysvastaavilla” - Haastateltava H3

”Siitä on jonkunnäköinen varmaan peruskäsitys, että asiakas on se Kelan käsittelijä. Musta itsestäni tuntuu, että välillä unohtuu, että lopulta sitä järjestelmää käyttää myös asiakaspalvelun henkilöstö. Se välillä unohtuu jopa tuoteomistajaltakin.” - Haastateltava H6

Huolestuttavaa onkin sen, jos tuoteomistaja, asiakkaan tai asiakkaiden edustajat eivät muista edustaa jotain käyttäjäryhmää. Yksi haastateltavista totesikin, että asiakkaan selkeyttäminen olisi selkeä kehityskohde Kelassa.

”Tää on mun mielestä yks ehkä suurimmista heikkouksista ainakin meidän tiimissä, ja Kelassa ylipäättään. Se on jotenkin vähän häivytetty se asiakkaan rooli tässä” - Haastateltava H1

Toisaalta toinen haastateltava toteaa, että asiakkaan määrittelemisellä ei saavuteta mitään lisäarvoa, kunhan työt sujuvat:

”Me tehdään meidän työt mahdollisimman hyvin tehokkaasti ja pystytään tuottamaan tavallaan niitä mitä meiltä tilataan mahdollisimman laadukkaasti.” - Haastateltava H3

Yksi haastateltavista nosti erikseen esille, että ongelma ei ole siitä, etteikö tiimi tietäisi, kenelle tuotetta tuotetaan, vaan sisäiset asiakkaat nähdään toisarvoisina.

”Kyllä selvillä ollaan siitä, kenelle tehdään, ja miksi tehdään.” – Haastateltava H5

”Aina välillä olen törmännyt vähän yllätyksekseni siihen, että -- saatetaan vähätellä sitä asiakasroolia tässä tilanteessa eli, että no kyllä ne tietää, mitä ne tekee, vaikka siellä järjestelmä ei toimisikaan kovin loogisesti tai siellä olisi heikkouksia tai jopa suoranaisia virheitä.” – Haastateltava H5

Riippuen siitä, kenelle tuotetta tuotetaan vaikuttaa siihen, kuinka hyvin asiakasnäkökulma huomioidaan ohjelmistokehityksessä. ”No kyllä ne tietää, mitä ne tekee” voisi viitata siihen, että nurinkurisesti ajateltuna asiakkaiden tai käyttäjien ammattitaitoon luotetaan niin paljon, että ohjelmistokehityksessä ei aina kannata ajatella esimerkiksi helppokäyttöisyyttä tai järjestelmän toimivuutta. Sama haastateltava haastaakin tätä ajattelutapaa: järjestelmiä tehdään asiakkaille, oli asiakas kuka hyvänsä.

”Mun mielestä asiakas on asiakas, oli se sisäinen asiakas tai ulkoinen asiakas” – Haastateltava H5

Kaikista yllä olevista vastauksista voisikin päätellä sen, että Kelan ohjelmistokehityksessä asiakasnäkökulmaa ei arvosteta tarpeeksi. Tämä tulkinta perustuu siihen, että moni kehitystiimeistä ei tiedä tai tunnista, kuka tai keitä heidän asiakkaansa ovat, tiimeistä saattaa puuttua tuoteomistaja ja joissakin tiimissä vähätellään sitä, kenelle järjestelmää kehitetään. Tämä muodostaa empiirisen johtopäätöksen EC60.

EC60: Organisaation ohjelmistokehityksessä ei huomioida ja arvosteta tarpeeksi asiakaslähtöisyyttä eli kenelle järjestelmiä kehitetään.

4.6.1 Käyttäjätarinat

Ketterä käytäntö, jossa ohjelmistovaatimukset muodostetaan parilla lauseella käyttäjän kielellä. (Turetken ym., 2016). Yksi haastateltavista vastasi:

”Me ei käytetä käyttäjätarinoita ” – Haastateltava H8

Tässä tiimissä käyttäjätarinoiden käyttäminen ei ole yleistä, sillä tiimin töiden pääpaino on ylläpidossa, ei ohjelmistokehityksessä. Tämä sama tilanne saattaa päteä myös muihin Kelan tiimeihin, ainakin osittain. Toinen haastateltavista mainitsi, että käyttäjätarinoiden sijaan käytetään käyttötapauksia:

”Käyttötapaukset ovat olleet se edellinen tapa määritellä, siksi niitä varmaan on enemmän, mutta kyllä käyttäjätarinoita aika yllättävästi jo näkee.” – Haastateltava H2

Haastateltavien vastausten perustella todisteita käyttäjätarinoiden käytöstä on vähän ja systemaattista käyttöä ei ole. Tämä muodostaa empiirisen johtopäätöksen EC61.

EC61: Käyttäjätarinoita käytetään organisaatiossa joissain tiimeissä, yleisempi tapa on hyödyntää käyttäjätarinoita raskaampia käyttötapauksia tai määrittelydokumentteja. (P)

Käyttäjätarinat eivät ole vielä kovin laajassa käytössä. Se, miksi käyttötapaukset ovat käyttäjätarinoita enemmän käytössä voisi johtua siitä, että monilla kehitystiimeillä tiimeihin kuuluu määrittelijöitä, joiden työnkuvaan kuuluu ohjelmistovaatimusten määrittely ja heillä saattaa olla perinteisempiä toimintamalleja käytössä, jossa vaatimuksen kirjoitetaan ensin ja sen jälkeen annetaan vasta näytettäväksi ohjelmoijalle.

”Tehdään hirveän tarkalla tasolla, ennen kuin on annettu toteuttajille kommentoitavaksi ja yleisfiilis on, että se on vielä aika tiukassa täällä tämä vanha ajattelu.” - Haastateltava H2

Haastateltavan vastauksesta voisi tulkita sen, että ohjelmistovaatimusten muodostaminen tapahtuu irrallaan ohjelmoijista ja ohjelmoijien osallistaminen tapahtuu myöhäisessä vaiheessa. Se, että ohjelmoijia ei osallisteta aikaisessa vaiheessa määrittelyyn, voisi nostaa riskiä, että ohjelmistokehityksen aikaisia virheitä tulee enemmän, kehittäminen on hidasta tai ohjelma ei vastaa käyttäjän todellisia tarpeita. Muiden haastateltavien perusteella organisaatiossa tapahtuu myös ketterien menetelmien mukaista määrittelyä. Kysyttäessä onko tiimillä määrittelijärooleissa toimivia henkilöitä tiimissä, haastateltavat vastasivat:

”Meillä on aika hyvin ketterää määrittelyä, meillä on nimellisesti yksi määrittelijä. Vaatimusmäärittelyssä kehittäjät on paljon mukana, testaajat on mukana” - Haastateltava H1

”Ei” - Haastateltava H7

Jatkokysymyksenä haastateltavalta, joka vastasi ”ei”, kysyttiin, miten tiimi toimii ilman määrittelijöitä, haastateltava vastasi:

”Meille tulee käyttötapauksuvaukset. Meillä on Definition of Ready pohja, minkä perusteella me halutaan ne Epicit sinne Jiraan. Niissä on meidän tiimin määrittelemät kysymykset niin, että me pyritään ymmärtämään mistä on kysymys. Ja kun me saadaan niistä tietää mistä on kysymys, me kerrotaan, että miten se voidaan tehdä.” - Haastateltava H7

Haastateltavien vastausten perusteella voisi tehdä sen päätelmän, että tiimit, jotka hyödyntävät määrittelijöitä vähemmän, käyttävät käyttötapausten sijaan käyttäjätarinoita. Tähän liittyen yksi haastateltavista nosti erikseen esille määrittelyyn ja määrittelijärooliin liittyvän epäkohdan:

”Määrittely täällä Kelassa on mun mielestä ihan käsittämätöntä, täällä saatetaan määritellä vielä jotain asiaa puoli vuotta sen jälkeen, kun kehitystiimi on tehnyt sen. Itse näkisin, että määrittelijäkerros pitäisi keventää” - Haastateltava H7

Organisaatiossa tehdään paljon turhaa ja liian tarkkaa määrittelyä siitä näkökulmasta, että ohjelmoijat pystyisivät hyvin paljon kevyemmällä määrittelyllä ohjelmoimaan vaatimusten mukaisesti. Myös toisen haastateltavan vastaus tukee tätä tulkintaa:

”Täällä on perinteisesti tehty hirveän tarkkoja kuvauksia määrittelyn osalta.” – Haastateltava H2

Eli jos vaatimusmäärittelyä tehtäisiin uudemmilla ketterien menetelmien mukaisilla käytänteillä, kuten käyttäjätarinoilla, organisaatio voisi säästää paljon resursseja. Organisaatiossa tehdään paljon turhaa ja liian tarkkaa määrittelyä, mikä saattaa johtua vanhentuneista käytänteistä. Tämä muodostaa Tämä muodostaa empiirisen johtopäätöksen EC62.

EC62: Organisaatiossa tehdään paljon turhaa ja liian tarkkaa määrittelyä, mikä saattaa johtua vanhentuneista käytänteistä.

4.6.2 Asiakas sitoutuu työskentelemään kehitystiimin kanssa

Asiakkaan on sitouduttava toimimaan kehitystiimin kanssa, jotta yhteistyö olisi sujuvaa. (Turetken ym., 2016) Tämän käytänteen arvioinnissa on hyödynnetty luvussa 4.7 Asiakasyhteistyössä esitettyjä haastattelijoiden vastauksia ja niistä tehtyjä empiirisiä johtopäätöksiä. Asiakkaan ja kehitystiimin välistä yhteistyötä korostettiin yhden haastateltavan vastauksissa:

”Asiakas on se, kenen tuoteomistajan jutut on valikoitunut siihen [SPRINTILLE]. Me ollaan tosi aktiivisesti juuri sen kanssa tekemisissä.” – Haastateltava H7

Muita havaintoja sujuvasta yhteistyöstä ei datassa ollut. Tämän käytänteen arvioinnissa tulisi huomioida jo tehty empiirinen johtopäätös EC55: Kehitystiimeille ei ole selvää, kuka heidän asiakkaansa on. Eli jos kehitystiimit eivät tunnista asiakastaan, ei yhteistyön sujuvuutta asiakkaan ja kehitystiimin välillä pystytä arvioimaan. Tämä muodostaa empiirisen johtopäätöksen EC63.

EC63: Yhteistyön sujuvuutta asiakkaan ja kehitystiimin välillä ei pystytä arvioimaan, sillä kehitystiimit eivät tunnista, kuka heidän asiakkaansa on. (N)

Tämä johtopäätös ei huomioi sitä, että kehitystiimin ja tuoteomistajan välinen yhteistyö voi hyvinkin olla tiivistä ja sujuvaa. Kysyttäessä, miten tiimi suhtautuu tuoteomistajaan eli mielletäänkö tuoteomistaja asiakkaan edustajana, yksi haastateltava vastasi:

”Toisessa tiimissä on tuoteomistaja, niin kyllä se on yksi tiimiläisistä.” – Haastateltava H8

Tässä tiimissä tuoteomistaja mielletään enemmän osana kehitystiimiä eikä asiakkaan edustajana, mikä voisi osittain aiheuttaa sen epäselvyyden, kuka on

tiimin asiakas. Kuten aiemmin todettu, tämän käytännön luotettava arviointi on haastavaa, jos kehitystiimeille ei ole selkeää, kuka asiakas on.

4.6.3 Asiakassopimus heijastaa evolutiivista kehitystä

Asiakas ymmärtää evolutiivisen ohjelmistokehityksen luonteen, jonka seurauksena asiakassopimuksessa vältetään määrittelemästä päivämäärät, jolloin virstanylväät on saavutettava. (Turetken ym., 2016) Tämän käytännön omaksumisen arvioinnissa on hyödynnetty Kehittämismallin sanallista kuvausta. Asiakassopimus vastaa organisaatiossa IT-kehittämisessä puitesopimista, jonka perustana on yhteisen sopimisen kautta rullaavasti päivittyvät kehittämisen tavoitteet (Kehittämismalliryhmä, 2022). Eli asiakassopimus on luonteeltaan sellainen, että se mahdollistaa ketterän kehittämisen ilman tarkkoja päivämääriä.

”Ketterien tyylien vastaisesti tai Scrumin vastaisesti meille on sanottu deadlinepäivämäärät, tuossa menette tuotantoon.” – Haastateltava H6

Haastateltavan vastaus osoittaa, että puitesopimusta ei välttämättä noudateta yksittäisissä tiimeissä. Tämä muodostaa empiirisen johtopäätöksen EC64.

EC64: Asiakassopimus mahdollistaa evolutiivisen ohjelmistokehityksen, missä päivämääriä ei määritellä tarkkaan, mutta sopimusta ei välttämättä noudateta. (L)

4.6.4 DevOps

Ketterä käytäntö, jolla varmistetaan nopeampi arvon kulku käyttäjälle integroimalla tiiviimmäksi ohjelmistokehitys sekä ylläpitoon liittyvät toiminnot. (Turetken ym., 2016) Kysyttäessä onko tiimeillä DevOps-toimintamallia käytössä, haastateltavat vastasivat:

”Kyllähän sitä on jo vähän aloiteltu.-- Siinäkin ollaan aika alkumetreillä.” – Haastateltava H2

”Ollaan pikkuhiljaa alettu ikään kuin sen oppimisen kautta. Me ollaan varattu oppimiseen liittyviä asioita, niin me ollaan otettu DevOps maailmaa [KÄYTTÖÖN] ja nyt ollaan päästy siihen vaiheeseen, että meillä on jonkinlainen pohja.” – Haastateltava H7

”Ei pysty, on vanhat systeemit” – Haastateltava H8

”Tavallaan nyt suunnataan tässä toimintamallissa siihen, että tiimi tekee tosiaan kaiken suunnilleen itse” – Haastateltava H6

DevOps-toimintamalli ei siis ole nykyisellään kovin laajasti käytössä, tosin tähän suuntaan on tarkoitus kehittää toimintamallia. Yksi haastateltavista nosti huolen esille toimintamallimuutokseen liittyen:

”Ei nämä ihmiset kerkeä koodaamaan sitä vaadittua järjestelmääkään, niin saati sitten oppimaan ja ottamaan haltuun nämä kaikki uudet välineet ja uudet osaamisalueet. Eikä me olla ainut tiimi, että me oltiin ensimmäiset, jotka siitä rupesi meteliä pitämään” - Haastateltava H6

Haastateltavan vastauksesta voisi tulkita sen, että organisaatio on muuttamassa toimintamalliaan perustavanlaatuisesti, mikä vaatii tiimeiltä paljon uuden opettelua ja ajattelumallin muutosta. Huomionarvoista on se, että erityisesti organisaation johdolla ei ole välttämättä tarvittavaa ymmärrystä siitä, mitä DevOps-toimintamalliin siirtyminen tiimeiltä vaatii. Tätä tulkintaa tukee haastateltavan vastaus, kun häneltä kysyttiin, onko tiimeillä DevOps-toimintamalli käytössä:

”Jossain vaiheessa, kun meillä sitä määriteltiin, mikä se DevOps sen jälkeen tipahdin ihan täysin, en enää tiedä mitä meillä haettiin” - Yksikön päällikkönä toimiva haastateltava

Jos organisaation linjajohdossa ei ole tarvittavaa ymmärrystä koko toimintamallin muutoksesta, se voisi tarkoittaa sitä, että tiimit eivät saa tarvitsemaansa tukea uuden toimintamallin käyttöönottoon. Yksi haastateltavista huomautti, että DevOps-toimintamalli pitäisi tilata tiimiltä, jotta se tulisi käyttöön:

”Ollaan odotettu, että se DevOps ikään kuin tilaus tulisi jostain, että joku pyytäisi meiltä sitä. Meidän mallihan tavallaan vaatii sen, että kaikki pitää tulla jostain.” - Haastateltava H7

Yhteenvetona DevOps-toimintamalli ei ole organisaatiossa vielä laajasti käytössä, mutta tarkoituksena on, että toimintamallia kehitetään siihen suuntaan, mutta linjajohdossa ei löydy välttämättä tarvittavaa ymmärrystä siitä, mitä toimintamallin muutos tiimeiltä vaatii. Tämä muodostaa empiirisen johtopäätöksen EC65.

EC65: DevOps-toimintamalli ei ole toistaiseksi organisaatiossa käytössä kuin muutamilla tiimeillä ja tarkempi ymmärrys siitä, mitä toimintamallin muutoksella tarkoitetaan saattaa puuttua linjajohdolta. (P)

4.6.5 Visio, ominaisuudet

Visio on sidosryhmien näkemys kehitettävästä ratkaisusta sidosryhmien tarpeiden ja ehdotettujen ominaisuuksien perusteella. Siinä huomioidaan suunnitellun ratkaisun korkean tason ominaisuuksien, ei-toiminnallisten vaatimusten ja suunnittelusta johtuvien rajoitteiden muodossa ja annetaan yleiskatsaus kehitettävästä järjestelmästä. (Turetken ym., 2016) Kysyttäessä onko tiimillä visiota, mitä kautta näkee, mihin suuntaan järjestelmää tulisi kehittää, haastateltava vastasi:

”Kyllä meillä viime ajat on, viimeiset pari kolme vuotta se on ollut.” – Haastateltava H6

”Periaatteessa kyllä. Kelassa on järjestelmiä, jotka on tehty 20 vuotta sitten, niillä on nyt näkyvissä sellaista roadmappia, että niitä pitää ruveta uudistamaan” – Haastateltava H8

”Joo ja ei. -- jos miettii esimerkiksi mitä kehityksen painopisteet on, niin ne ei näy siellä” – Haastateltava H3

”Joo, tavallaan. Me tehdään vuosisuunnitelmataulukkoon kolmen kuukauden suunnittelujaksoja ja sinne tuoteomistajat käy etukäteen laittamassa toiveita mitä ne haluaa, niin tavallaan me nähdään jo sieltä mitä on tulossa. -- On tarkoitus, että tuoteomistaja tekisi vielä toisenlaisiakin roadmappeja, mut toi on meille aika lailla riittävä.” – Haastateltava H7

Sanoista ”toisenlaisia roadmappeja” voisi tulkita se, että tietylle tuotteelle ei löydy selkeää pidemmän aikavälin visiota, vaan näkyvyys jää suunnittelujakso tasolle eli 3 kuukaudelle. Yksi haastateltavista oli sitä mieltä, että visioita tai tiekarttoja ei ole.

”Mä en ole nähnyt selkeätä mitään roadmappia ylipäänsä tän sovelluksen tai järjestelmän kehittämisestä. Mutta koska tää on iso laaja olemassa oleva järjestelmä niin musta tuntuu, että tässä enemmänkin se kehittäminen on niiden kulloistenkin tarpeiden vaatimaa kehittämistä. – Haastateltava H5

Se miksi visioita ei käytetä voi johtua siitä, että tiimit kehittävät jo olemassa olevia järjestelmiä ja suurempia kehityskohteita ei välttämättä enää ole. Visioitahan voi olla, mutta ainakaan niitä ei ole jaettu kehitystiimien kanssa. Haastateltavien vastausten perusteella visioita käytetään joissain tiimeissä. Tämä muodostaa empiirisen johtopäätöksen EC66.

EC66: Vain harvalla tuotteella tai järjestelmällä on visio, joka on jaettu kehitystiimin kanssa. (P)

4.6.6 Vaikutus asiakkaisiin ja toimintoihin

Kehitysmallin muutosten vaikutuksia myyntiin, toimintaan ja asiakkaaseen tulisi mitata. (Turetken ym., 2016) Tämän käytänteen arvioinnissa on hyödynnetty luvussa 4.7 Asiakasyhteistyössä esitettyjä haastattelijoiden vastauksia ja niistä tehtyjä empiirisiä johtopäätöksiä. Kuten aiemmin osoitettiin, kehitystiimeissä ei ole selkeää käsitystä siitä kuka asiakas on, mikä heijastuu myös tämän käytännön omaksumisen arviointiin. Näyttää siltä, että organisaatiossa mitattaisiin kehitysmallin muutosten vaikutuksiin, toimintaan tai asiakkaaseen, ei löytynyt. Tämä voi johtua siitä, että tätä organisaatiossa ei mitata muutosten vaikutuksia tai haastateltavat eivät edusta sellaista joukkoa, jossa tällaista tehtäisiin. Tämä muodostaa empiirisen johtopäätöksen EC67.

EC67: Kehitysmallin muutosten vaikutusten mittaamista myyntiin, toimintaan tai asiakkaaseen ei löydetty datan perusteella. (N)

4.6.7 Asiakas on heti käytettävissä

Asiakas on yhteistyöhön osallistuva, edustava, valtuutettu, sitoutunut ja asiantunteva. Tässä käytännössä olennaista ei ole asiakkaan fyysinen sijainti, vaan se, onko asiakas heti käytettävissä. (Turetken ym., 2016) Tämän käytänteen arvioinnissa on hyödynnetty luvussa 4.7 Asiakasyhteistyössä esitettyjä haastattelijoiden vastauksia ja niistä tehtyjä empiirisiä johtopäätöksiä. Asiakas on heti käytettävissä ainakin yhdellä haastattelijalla:

”Jos tarvitaan käyttäjätukitestausta tai selvennyksiä, niin on tosi aktiivisesti mukana.”
- Haastateltava H7

”Meillä on Teamsissa yhteistyökanavat tuotteiden tuoteomistajien kanssa, minne he voi laittaa kysymyksiä ja kuka tahansa tiimistä voi käydä vastaamassa ja sillä tavalla saadaan tieto ikään kuin leviämään joka paikkaan” - Haastateltava H7

Muita havaintoja siitä, että asiakas olisi heti käytettävissä, sekä edustava, valtuutettu, sitoutunut ja asiantuntija, ei datan perusteella ollut. Tähän voi vaikuttaa hyvin vahvasti se, että kaikille haastatelijoille ei ole selvää, kuka heidän asiakkaansa on. Yhden haastateltavan vastaus tukee kuitenkin sitä tulkintaa, että asiakas ei ole valtuutettu tekemään päätöksiä.

”Tuoteomistajien pitäisi pystyä päättää asioita” - Haastateltava H2

Sanasta ”pitäisi” voisi tulkita sen, että tuoteomistajat eivät kaikki ole välttämättä sellaisessa asemassa, että he olisivat valtuutettuja tekemään päätöksiä. Yhteenvedon voisi todeta sen yllä olevien vastausten perusteella, että koska kehitystiimeille ei ole selkeä se, ketä tarkoitetaan heidän asiakkaallansa, asiakas ei ole kaikilla tiimeillä heti käytettävissä. Tämä muodostaa empiirisen johtopäätöksen EC68.

EC68: Kaikilla tiimeillä asiakas ei ole heti käytettävissä johtuen kehitystiimien epäselvyydestä siitä ketä tai keitä asiakkaalla tarkoitetaan. (N)

4.6.8 Asiakasyhteys pyörii yhteistyöhön sitoutumisen ympärillä

Asiakas suostuu sopimaan yhteistyön eikä vaatimusten ja ominaisuuksien määräästä. (Turetken ym., 2016) Tämän käytänteen arvioinnissa on hyödynnetty luvussa 4.7 Asiakasyhteistyössä esitettyjä haastattelijoiden vastauksia ja niistä tehtyjä empiirisiä johtopäätöksiä. Datasta ei löytynyt käytänteitä siitä, minkälaista asiakasyhteys on luonteeltaan. Tämä käytäntö jätetään arvioimatta siihen perustuen, että luotettavaa dataa aiheesta ei löytynyt.

4.6.9 Toistuva kasvokkain tapahtuva vuorovaikutus kehittäjien ja käyttäjien välillä

Ihanteellista olisi, että ei vaan kehittäjät, vaan myös asiakkaat ja käyttäjät tekisivät yhteistyötä fyysisesti samoissa tiloissa. (Turetken ym., 2016) Tämän käytänteen arvioinnissa on hyödynnetty luvussa 4.7 Asiakasyhteistyössä esitettyjä haastattelijoiden vastauksia ja niistä tehtyjä empiirisiä johtopäätöksiä. Kuten aiemmin kävi ilmi, kehitystiimeissä ei ole selkeää käsitystä siitä kuka asiakas tai käyttäjä on, joten näissä tiimeissä ei tapahdu toistuvaa kasvokkaista vuorovaikutusta kehittäjien ja käyttäjien välillä. Yksi haastatteliijoista kuitenkin mainitsi, että heidän tiimissään entinen käyttäjä on nykyinen testaaja.

”Me saatiin tavallaan testaajiksi suoraan käyttäjät.” – Haastateltava H6

Yhteenvedona voisi tulkita, että kehittäjien ja käyttäjien välillä ei ole nähtävissä toistuvaa kasvokkaista tapahtuvaa vuorovaikutusta. Tämä muodostaa empiirisen johtopäätöksen EC69.

EC69: Kehittäjät ja käyttäjät eivät tee toistuvasti kasvokkaista tapahtuvaa yhteistyötä. (N)

Yhteenvedona asiakasyhteistyön periaatteeseen löytyi yhteensä 11 empiiristä johtopäätöstä. Näiden yhteenvedojen pohjalta muodostettiin ensisijainen empiirinen johtopäätös PEC4.

PEC6: Ohjelmistokehityksessä olisi tärkeää huomioida sujuva asiakasyhteistyö ja se, kenelle järjestelmiä kehitetään, mutta kehitystiimeille ei ole selkeää, kuka heidän asiakkaansa on tai kuka heidän asiakkaistaan on kaikkein tärkein.

Asiakasyhteistyön empiiriset johtopäätökset ovat esiteltynä seuraavassa taulukossa (TAULUKKO 11).

TAULUKKO 11 Asiakasyhteistyön empiirisen johtopäätökset

Tunniste	Empiirinen johtopäätös
EC59	Kehitystiimeille ei ole selvää, kuka heidän asiakkaansa on tai kuka heidän asiakkaistaan on kaikkein tärkein.
EC60	Organisaation ohjelmistokehityksessä ei huomioida ja arvosteta tarpeeksi asiakaslähtöisyyttä eli kenelle järjestelmiä kehitetään.
EC61	Käyttäjätarinoita käytetään organisaatiossa joissain tiimeissä, yleisempi tapa on hyödyntää käyttäjätarinoita raskaampia käyttötapauksia tai määrittelydokumentteja. (P)
EC62	Organisaatiossa tehdään paljon turhaa ja liian tarkkaa määrittelyä, mikä saattaa johtua vanhentuneista käytännöistä.

- EC63 Yhteistyön sujuvuutta asiakkaan ja kehitystiimin välillä ei pystytä arvioimaan, sillä kehitystiimit eivät tunnista, kuka heidän asiakkaansa on. (N)
- EC64 Asiakassopimus mahdollistaa evolutiivisen ohjelmistokehityksen, missä päivämääriä ei määritellä tarkkaan, mutta sopimusta ei välttämättä noudateta. (L)
- EC65 DevOps-toimintamalli ei ole toistaiseksi organisaatiossa käytössä kuin muutamilla tiimeillä ja tarkempi ymmärrys siitä, mitä toimintamallin muutoksella tarkoitetaan saattaa puuttua linjajohdolta. (P)
- EC66 Vain harvalla tuotteella tai järjestelmällä on visio, joka on jaettu kehitystiimin kanssa. (P)
- EC67 Kehitysmallin muutosten vaikutusten mittaamista myyntiin, toimintaan tai asiakkaaseen ei löydetty datan perusteella. (N)
- EC68 Kaikilla tiimeillä asiakas ei ole heti käytettävissä johtuen kehitystiimien epäselvyydestä siitä ketä tai keitä asiakkaalla tarkoitetaan. (N)
- EC69 Kehittäjät ja käyttäjät eivät tee toistuvasti kasvokkain tapahtuvaa yhteistyötä. (N)
- PEC6 Ohjelmistokehityksessä olisi tärkeää huomioida sujuva asiakasyhteistyö ja se, kenelle järjestelmiä kehitetään, mutta kehitystiimeille ei ole selkeää, kuka heidän asiakkaansa on tai kuka heidän asiakkaistaan on kaikkein tärkein.

4.7 Yhteenveto

Tässä luvussa esiteltiin empiirisen aineiston analyysi sekä sen tulokset. Empiiristä johtopäätöksiä muodostettiin yhteensä 69 ja ensisijaista empiiristä johtopäätöstä yhteensä kuusi. Alla olevassa taulukossa (TAULUKKO 12) on koottuna yhteen ensisijaiset empiiriset johtopäätökset, joita hyödynnetään luvussa 5 Tulosten pohdinta.

TAULUKKO 12 Ensisijaiset empiirisen johtopäätökset

Tunniste	Empiirinen johtopäätös
PEC1	Kelan organisaatiokulttuurin ja -toimintaan kuuluu erilaisia asioita, jotka toimivat ketteryyden omaksumisen esteenä.
PEC2	Asiakaspalautteen kautta toiminnan kehittäminen ja asiakasarvon tuottaminen eivät ohjaa ohjelmistokehitystyötä.
PEC3	Tulevien töiden parempi suunnittelu ja töiden pilkkominen pienempiin osiin auttaisi kehitystiimejä tuottamaan asiakkaalle nopeammin arvoa.
PEC4	Organisaatiossa tehdään paljon yhteistyötä, mutta tiimejä pitäisi enemmän kannustaa ja tukea kohti itseohjautuvuutta. Tiimeille pitäisi antaa aito valta ja tuki toimia ketterällä toimintamallilla.
PEC5	Ohjelmistokehitystä lähestytään työkalu- ja ohjepainotteisesti, jolloin yhteiset toimintatavat ja prosessit eli ohjelmistokehityskulttuuri jää kehittymättä.
PEC6	Ohjelmistokehityksessä olisi tärkeää huomioida sujuva asiakasyhteistyö ja se, kenelle järjestelmiä kehitetään, mutta kehitystiimeille ei ole selkeää, kuka heidän asiakkaansa on tai kuka heidän asiakkaistaan on kaikkein tärkein.

5 TULOSTEN TULKINTA JA POHDINTA

Tämän tutkielma tavoitteena oli arvioida Kelan ketteryyden arvojen ja periaatteiden omaksumista Turetkenin ym. (2016) SAFe MM -kypsyystasomalliin perusteella. Toisena tutkittavana asiana oli tunnistaa ne organisaation osa-alueet, jotka vaativat kehittämistä kypsyystasomallin perusteella. Kolmantena tutkielman tavoitteena oli tutkia sitä, kuinka hyvin Turetkenin ym. (2016) SAFe MM kypsyystasomalli soveltuu ketteryyden kypsyystason mittaamiseen.

Tutkielman empiirisessä osuudessa hyödynnettiin laadullista tutkimusta, jossa aineisto kerättiin teemahaastattelujen avulla. Tutkimuksen kohteena oli julkishallinnollinen organisaatio, Kela.

5.1 Kelan kypsyystasomallin tulokset

Turetkenin ym. (2016) SAFe MM -mallissa on viisi eri kypsyystasoa. Ensimmäisellä tasolla arvioidaan yhteistyötä, toisella tasolla arvioidaan ohjelmistokehitysprosessin evolutiivista lähestymistapaa, kolmannella tasolla arvioidaan laadukkaan ohjelmistokehityksen tehokkuutta, neljännellä tasolla arvioidaan kykyä reagoida muutokseen palautteen perusteella ja viidennellä eli korkeimmalla tasolla arvioidaan kaiken kattavan ympäristön luomista ketteryyden ylläpitämiseksi. (Turetken, ym., 2016) Jokaisen käytännön perässä on kirjain N, L, P tai F, mikä kuvastaa sitä, onko periaatteen saavuttamisesta käytännön näyttöä vähän, jonkin verran, paljon tai hyvin paljon Kelassa. Tarkemmat kirjainten kuvaukset löytyvät alaluvusta 2.6.5. SAFe MM mallin käyttäminen.

Ensimmäisellä tasolla arvioidaan yhteistyön sujuvuutta. Empiirisen aineiston perusteella Kela on onnistunut saavuttamaan tason 1 tavoitteen eli yhteistyön periaatteen osittain. Taseon liittyvät käytänteet ja niiden arvosanat ovat lueteltuna tarkemmin alla olevassa taulukossa (TAULUKKO 13). Tason 1 osalta käytänteet, jotka on saavutettu täysin ovat ”yhteissuunnittelu sekä tiedon

jakaminen” ja isoimmat puutteet löytyivät käytänteestä ”hyväksymistestaus”. Tältä tasolta yksi käytäntö jätettiin arvioimatta.

TAULUKKO 13 Organisaation ketteryyden kypsyyt: Taso 1

	Periaate				
	Muutoksen omaksuminen asiakasarvon tuottamiseksi	Ohjelmistojen suunnittelu ja toimittaminen usein	Ihmiskeskeisyys	Tekninen huippu-osaaminen	Asiakasyhteistyö
Taso 1: Yhteistyössä tehty	Prosessin reflektointi ja parantaminen (L)	Yhteis-suunnittelu (F)	Voimaantuneet ja motivoituneet tiimit (P) Yhteistyötä tekevät tiimit (P)	Koodaus-standardit (P) Tiedon jakaminen (F) Vapaaehtoisuus tehtävissä (L) Hyväksymistestaus (N)	Käyttäjätarinat (P) Asiakas sitoutuu työskentelemään kehitystiimin kanssa (ei arvioitu)

Alla olevassa taulukossa (TAULUKKO 14) tason 1 saavutetut tulokset ovat jaoteltuna vielä prosentuaalisesti osoittamaan sitä, missä arvosanassa pääpaino on tasolla 1. Suuri painopiste (44%) on arvosanassa osittain saavutettu (P). Huomioitavaa on, että vain 11% käytänteistä sai tällä tasolla arvosanan ei saavutettu (N).

TAULUKKO 14 Organisaation ketteryyden tulokset ensimmäisen kypsyystason osalta

Kypsyystaso	(F) Täysin saavutettu	(L) Laajasti saavutettu	(P) Osittain saavutettu	(N) Ei saavutettu	Käytänteitä yhteensä /arvioitu (kpl)
Taso 1	22%	22%	44%	11%	10/9

Toisella tasolla arvioidaan sitä, onko ohjelmistokehitysprosessin lähestymistapa evolutiivinen. Empiirisen aineiston perusteella Kela on saavuttanut tason 2 tavoitteen eli evolutiivisuuden laajasti. Tasoon liittyvät käytänteet ja niiden arvosanat ovat lueteltuna tarkemmin alla olevassa taulukossa (TAULUKKO 15). Tällä tasolla kaikki käytänteen on arvioitu.

TAULUKKO 15 Organisaation ketteryyden kypsyyt: Taso 2

	Periaate				
	Muutoksen omaksuminen asiakasarvon tuottamiseksi	Ohjelmistojen suunnittelu ja toimittaminen usein	Ihmiskeskeisyys	Tekninen huippuosaaminen	Asiakasyhteistyö

Taso 2: Evolutiivisuus	Vaatimusten evolutiivisuus (P) Pienemmät, useammat julkaisut (L) Vaatimusten löytäminen (L)	Jatkuva tuottaminen (P) Kahden tason suunnittelu ja seuraaminen (L) Ketteryyden arviointi ja kiertonopeus (F) Julkaisun suunnittelu (P)	Määrittely/ohjelmointi/testaus -tiimi (P)	Sovellus-kokoonpanon hallinta (F) Automaatiotestaus (P) Iteraation edistymisen seuranta (L) Ei isoa suunnittelua etukäteen (L) Tuotteen kehitysjojo (L)	Asiakas-sopimus heijastaa evolutiivista kehitystä (L)
-----------------------------------	---	--	---	---	---

Alla olevassa taulukossa (TAULUKKO 16) tason 2 saavutetut tulokset ovat jaoteltuna arvosanan mukaisesti. Suurin osa (50%) käytänteistä on saanut arvosanan laajasti saavutettu (L). Yksikään käytäntö ei saanut arvosanaa ei saavutettu (N).

TAULUKKO 16 Organisaation ketteryyden tulokset toisen kypsyystason osalta

Kypsyystaso	(F) Täysin saavutettu	(L) Laajasti saavutettu	(P) Osittain saavutettu	(N) Ei saavutettu	Käytänteitä yhteensä /arvioitu (kpl)
Taso 2	7%	50%	36%	0%	14/14

Kolmannella tasolla arvioidaan ohjelmistokehitysprosessin tehokkuutta tuottaa toimiva, laadukas ohjelmisto. Empiirisen aineiston perusteella Kela on saavuttanut tason 3 tavoitteen eli tehokkuuden tavoitteen osittain. Tasoon liittyvät käytänteet ja niiden arvosanat ovat lueteltuna tarkemmin alla olevassa taulukossa (TAULUKKO 15). Tällä tasolla arvioimatta jätettiin yhteensä kaksi käytäntöä.

TAULUKKO 17 Organisaation ketteryyden kypsyys: Taso 3

	Periaate				
	Muutoksen omaksuminen asiakasarvon tuottamiseksi	Ohjelmistojen suunnittelu ja toimittaminen usein	Ihmiskeskeisyys	Tekninen huippuosaaminen	Asiakas-yhteistyö

Taso 3: Tehokas	Säännöllinen reflektointi ja sopeutuminen (P)	Riskilähtöiset iteraatiot (N) Ominaisuuksien, eikä tehtävien suunnittelu (P) Tiekartta (P) Iteraation hallitseminen (P) Kanbanin hyödyntäminen (N) PSI (engl. <i>potentially releasable increment</i>)/julkaisu (P) Julkaisujuna (engl. <i>Agile Release Train</i>) (P)	Itseohjautuvat tiimit (P) Säännöllinen henkilökohtainen vuorovaikutus (L) Scrumien Scrum (P)	Jatkuva integraatio (ei arvioitu) Jatkuva parantaminen (refaktorointi) (P) Yksikkötestit (P) 30% tason 2 ja tason 3 ihmisistä (ei arvioitu)	DevOps (P) Visio, ominaisuudet (N) Vaikutus asiakkaisiin ja toimintaan (N)
---------------------------	---	---	--	--	--

Alla olevassa taulukossa (TAULUKKO 18) tason 3 saavutetut tulokset ovat jaoteltuna arvosanan mukaisesti. Eri osa-alueiden käytänteet on saavutettu eri tasoisesti, mutta suurin osa (69%) käytänteistä on saanut arvosanan osittain saavutettu (P). Tällä tasolla yhtäkään käytäntöä ei saavutettu täysin ja toisaalta (25%) käytänteistä ei saavutettu ollenkaan.

TAULUKKO 18 Organisaation ketteryyden tulokset kolmannen kypsyystason osalta

Kypsyystaso	(F) Täysin saavutettu	(L) Laajasti saavutettu	(P) Osittain saavutettu	(N) Ei saavutettu	Käytänteitä yhteensä /arvioitu (kpl)
Taso 3	0%	6%	69%	25%	18/16

Neljännellä tasolla arvioidaan kykyä reagoida muutokseen palautteen perusteella eli adaptiivisuutta. Empiirisen aineiston perusteella Kela ei ole saavuttanut tason 4 tavoitetta eli adaptiivisuuden tavoitetta ollenkaan. Tasoon liittyvät käytänteet ja niiden arvosanat ovat lueteltuna tarkemmin alla olevassa taulukossa (TAULUKKO 19).

TAULUKKO 19 Organisaation ketteryyden kypsyys: Taso 4

	Periaate
--	-----------------

	Muutoksen omaksuminen asiakasarvon tuottamiseksi	Ohjelmistojen suunnittelu ja toimittaminen usein	Ihmiskeskeisyys	Tekninen huippuosaaminen	Asiakasyhteistyö
Taso 4: Adaptiivinen	Asiakaslähtöiset iteraatiot (N) Jatkuva asiakastyytyväisyys -palaute (P) Leanin vaatimukset skaalautuvassa mittakaavassa (P)	Pienemmät, useammat julkaisut (N) Adaptiivinen suunnittelu (N) Liiketoiminnan tehokkuuden arvioiminen (N)	Hajautettujen tiimien hallinta (F)	Tarkoituksen mukainen arkkitehtuuri (P) Päivittäiset edistymisen seuranta-kokoukset (P)	Asiakas on heti käytettävissä (N) Asiakasyhteistyö pyörii yhteistyön sitoutumisen ympärillä (ei arvioitu)

Alla olevassa taulukossa (TAULUKKO 20) tason 4 saavutetut tulokset ovat jaoteltuna arvosanan mukaisesti. Suurin osa (50%) käytänteistä on saanut arvosanan ei saavutettu (N). Erikoisena asiana on se, että yksikään käytäntö ei ole saanut arvosanaa laajasti saavutettu, mutta yksi on saanut arvosanan täysin saavutettu.

TAULUKKO 20 Organisaation ketteryyden tulokset neljännen kypsyystason osalta

Kypsyystaso	(F) Täysin saavutettu	(L) Laajasti saavutettu	(P) Osittain saavutettu	(N) Ei saavutettu	Käytänteitä yhteensä /arvioitu (kpl)
Taso 4	10%	0%	40%	50%	11/10

Viidennellä tasolla arvioidaan kykyä vakiinnuttaa kaiken kattavan ympäristön luomista ketteryyden ylläpitämiseksi. Empiirisen aineiston perusteella Kela ei ole saavuttanut tason 5 tavoitteen eli kattavuuden tavoitetta. Taseen liittyvät käytänteet ja niiden arvosanat ovat lueteltuna tarkemmin alla olevassa taulukossa (TAULUKKO 21).

TAULUKKO 21 Organisaation ketteryyden kypsyys: Taso 5

	Periaate				
	Muutoksen omaksuminen asiakasarvon tuottamiseksi	Ohjelmistojen suunnittelu ja toimittaminen usein	Ihmiskeskeisyys	Tekninen huippuosaaminen	Asiakasyhteistyö

Taso 5: Kattava	Matala prosessiseremonia (P) Jatkuva SAFe kyvykkyksien parantaminen (ei arvioitu)	Ketterien projektien arviointi (P)	Ihanteellinen fyysinen ympäristö ketterälle kehittämiselle (P) Organisaation muuttaminen (N)	Testilähtöinen kehittäminen (N) Tason -1 tai 1B ihmisiä vähän tai ei ollenkaan tiimissä (ei arvioitu) Rinnakkainen testaaminen (N)	Kehittäjien ja käyttäjien välinen vuorovaikutus (N)
----------------------------	--	------------------------------------	---	--	---

Alla olevassa taulukossa (TAULUKKO 22) tason 5 saavutetut tulokset ovat jaoteltuna arvosanan mukaisesti. Eri osa-alueiden käytänteet on saavutettu eri tasoisesti, mutta suurin osa (57%) käytänteistä on saanut arvosanan ei saavutettu (N).

TAULUKKO 22 Organisaation ketteryyden tulokset viidennen kypsyystason osalta

Kypsyystaso	(F) Täysin saavutettu	(L) Laajasti saavutettu	(P) Osittain saavutettu	(N) Ei saavutettu	Käytänteitä yhteensä /arvioitu (kpl)
Taso 5	0%	0%	43%	57%	9 / 7

Yhteenvedon kaikkien eri tasojen perusteella

Alla olevassa taulukossa (TAULUKKO 23) on koottuna yhteen kaikki SAFe MM-mallin käytänteiden arvosanojen suhde eri kypsyystasoittain sekä arvosanojen osuus kaikkiin käytänteisiin suhteutettuna. Kela on saanut eniten (47%) arvosanoja "osittain saavutettu". Tämän jälkeen yleisin arvosana on "ei saavutettu" (29%). Vain muutama (16%) käytäntö on onnistuttu saavuttamaan laajasti ja täysin saavutettuja on hyvin vähän (8%).

TAULUKKO 23 Organisaation ketteryyden tulokset

Kypsyystasot	(F) Täysin saavutettu	(L) Laajasti saavutettu	(P) Osittain saavutettu	(N) Ei saavutettu	Käytänteitä yhteensä /arvioitu (kpl)
Taso 5	0%	0%	43%	57%	9/7
Taso 4	10%	0%	40%	50%	11/10
Taso 3	0%	6%	69%	25%	18/16
Taso 2	7%	50%	36%	0%	14/14
Taso 1	22%	22%	44%	11%	10/9
Yhteensä	8%	16%	47%	29%	62/56

SAFe MM -mallin perusteella Kela on saavuttanut kypsyystaso 1 osittain, ja tason 2 kypsyystason laajasti. Kelan kypsyystaso on siis kypsyystason 1 ja 2 välissä. Turetkenin ym. (2016) mukaan ylempien kypsyystasojen käytänteiden omaksuminen ei ole tehokasta, ellei alempien tasojen käytänteitä ole saavutettu. Eli joissain tapauksissa ylemmillä tasoilla hyvän arvosanan saanut käytäntö ei välttämättä ole

Tuloksista kävi ilmi, että organisaation osa-alueista eniten kehittämistä vaatisivat asiakkaan tunnistaminen, asiakkaan ymmärtäminen sekä asiakasarvon luominen mahdollisimman usein. (PEC2: Asiakaspalautteen kautta toiminnan kehittäminen ja asiakasarvon tuottaminen eivät ohjaa ohjelmistokehitystyötä; PEC3: Tulevien töiden parempi suunnittelu ja töiden pilkkominen pienempiin osiin auttaisi kehitystiimejä tuottamaan asiakkaalle nopeammin arvoa; PEC6: Ohjelmistokehityksessä olisi tärkeää huomioida sujuva asiakasyhteistyö ja se, kenelle järjestelmiä kehitetään, mutta kehitystiimeille ei ole selkeää, kuka heidän asiakkaansa on tai kuka heidän asiakkaistaan on kaikkein tärkein.)

Myös muita kehitettäviä asioita tunnistettiin ja ne ovat esimerkiksi johtaminen, yksilöiden ja tiimien valmentaminen. Esimerkiksi yksilöiden valmentaminen pitää sisällään myös linjajohtoon kuuluvien henkilöiden opastaminen siitä, mitä ketteryys on ja miten ketterässä toimintamallissa johtajuus näkyy. (PEC4: Organisaatiossa tehdään paljon yhteistyötä, mutta tiimejä pitäisi enemmän kannustaa ja tukea kohti itseohjautuvuutta. Tiimeille pitäisi antaa aito valta ja tuki toimia ketterällä toimintamallilla.)

Tutkielman puitteissa tunnistettiin myös ketteryyden omaksumista hidastavia tai estäviä kulttuurillisia tekijöitä, jotka ovat johtaminen, asenteet toisia työntekijöitä kohtaan sekä työympäristössä vallitsevat arvot. (PEC 1: Kelan organisaatiokulttuurin ja -toimintaan kuuluu erilaisia asioita, jotka toimivat ketteryyden omaksumisen esteenä).

Organisaatiokulttuuriin liittyvien asioiden kehittämistä pitäisi tutkia tarkemmin, jotta olisi mahdollista muodostaa parempi käsitys yksittäisistä kehittämiskohteista. Datan perusteella moni ketteryyteen liittyvästä asiasta, kuten itse ketteryys, automaatiotestaus, DevOps-toimintamalli ja kulttuurin muutosmatka kohti ketteryyttä vaatisi organisaatiolta syvempää ymmärrystä siitä, mistä näissä asioissa on kyse ja miten asioita voisi kehittää. Uudenlainen lähestymistapa olisi tarpeen, kun organisaatiossa mietitään uusien toimintatapojen ja teknologioiden käyttöönottoa: ensin kannattaa paneutua yhteisiin toimintatapoihin ja prosesseihin, sitten vasta tarkemmin työkaluihin ja ohjeisiin. (PEC5: Ohjelmistokehitystä lähestytään työkalu- ja ohjepainotteisesti, jolloin yhteiset toimintatavat ja prosessit eli ohjelmistokehityskulttuuri jää kehittymättä.)

Asiat, jotka Kelassa toimivat liittyvät muun muassa yhteissuunnitteluun, tiedon jakamiseen, hajautettujen tiimien hallintaan, vapaaehtoisuuteen tiimeissä ja säännölliseen henkilökohtaiseen vuorovaikutukseen. Ohjelmistokehityksen julistuksen arvoihin peilaten Kela on onnistunut luomaan yhteistyötä korostavia

foorumeita, mutta yksilöiden korostamisessa ja arvostamisessa on vielä kehitettävää. Osassa tiimejä saadaan tuotettua toimivaa ohjelmisto lyhyessä aikavälissä, mutta dokumentoinnin taso voi olla väärä siinä mielessä, että dokumentaatiota saatetaan tuottaa liikaa vaatimuksiin nähden. Asiakasyhteistyön kohdalla Kelalla on eniten kehittämistä, sillä kuten aiemmin empiirisitä johtotuloksista kävi ilmi, moni kehitystiimi ei tunnista omaa asiakastaan. Kelassa myös reagoidaan hyvin nopeaan muutokseen, joskus jopa sellaisella tahdilla, että suunnitelmia ei ehditä luomaan ja suunnitelmien hyödyt jäävät saamatta.

5.2 SAFe MM kypsyysmallin arviointi

Mallissa tulokset laskevat odotetusti alemmille arvosanoille, mitä korkeammalle kypsyystasoa mennään muutamia poikkeuksia lukuun ottamatta. Erityisesti alemmilla kypsyystasoilla arvosanojen suhteen on epäloogisuuksia, mutta ylemmillä tasoilla, tasoilla 3,4 ja 5 arvosanat laskevat odotetusti. Osittain arvosanojen epäloogisuutta voi selittää se, että kaikkia käytänteitä ei arvioitu, kuten esimerkiksi tasolla 2 yksikään käytäntö ei ollut saanut arvosanaa "ei saavutettu", mutta sitä alemmalla tasolla, tasolla 1 arvosanan oli kuitenkin saanut 11%. Luku olisi voinut olla eri, jos tasolta 1 olevat käytänteet olisi kaikki arvioitu.

Tasolla 4 käytänteistä 10% on saavutettu täysin, vaikka sitä alemmalla tasolla 2 täysin saavutettuja käytänteitä ei ollut ollenkaan. Kyseinen käytäntö liittyy hajautettujen tiimien hallintaan, jossa arvioitavana asiana on asianmukainen viestintä sekä tarvittavat verkko- ja työkalurakenteiden käyttö. Useissa organisaatioissa ja tiimeissä toimitaan nykyisin hajautetuissa tiimeissä, joten voi olla, että käytännön tulisi sijaita alemmalla kypsyystasolla, esimerkiksi tasolla 3. Tulosten perusteella voisi vahvistaa, että käytänteet ovat mallissa oikealla kypsyystasoilla.

Tason 1 ja tason 2 välillä on kuitenkin enemmän epäloogisuuksia, sillä odotus oli, että arvosanat laskisivat sen mukaisesti, mitä korkeammalle kypsyystasossa mennään. Näin ei kuitenkaan käynyt tason 1 ja taso 2 välillä, vaan arvosanojen painopiste nousi korkeammalle, kun tasolta 1 siirryttiin tasolle 2. Osittain tätä voi selittää se, että tasolla 1 jätettiin yksi käytäntö arvioimatta. Epäselväksi jää, onko kypsyysmallin tasoilla 1 ja 2 sellaisia käytänteitä, jotka kuuluisivat kypsyysmallissa toiselle tasolle. Se, että ylempien tasojen käytänteitä omaksutaan ilman, että alempien tasojen käytänteitä on saavutettu, johtaa siihen, että käytänteet eivät ole tehokkaita (Turetken ym., (2016). Kela on siis voinut yrittää omaksua ylempillä tasolla olevia käytänteitä ilman, että alempana olevat käytänteet on saavutettu. Toisaalta epäloogisuus voisi johtua myös siitä, että organisaatioissa on hyvin eri kypsyystasoilla olevia tiimejä, jolloin tulokset ovat vääristyneet niiden myötä ja itse SAFe MM -mallissa ei ole vikaa.

SAFe MM kypsyystasomalli sopii soveltaen julkishallinnollisen organisaation kypsyystason arviointiin. Huomioitavaa viitekehysten käytössä

on se, että haastattelujen pitäminen ja aineiston analysointi voi olla raskasta. Tällä perusteella malli ei sovi välttämättä kaikkiin arviointitilanteisiin, esimerkiksi jos arviointiprosessiin ei pysty sitoutumaan. Toinen huomioitava asia mallin käytöstä on se, että kaikista käytänteistä ei löytynyt tarkkaa kuvausta. Esimerkiksi käytänteistä ”jatkuva SAFe kyvykkyyksien parantaminen” ja ”automaatiotestaus” käytännön kuvaus puuttui kokonaan. Kolmas huomioitava asia on se, että monen käytännön kuvaus on hyvin lähellä toista kuvausta. Esimerkiksi käytäntöjen ”pienemmät ja useammin toistuvat julkaisut”, ”jatkuva tuottaminen” sekä ”iteraation hallitseminen” kuvaukset ovat hyvin lähellä toisiaan. Neljäs huomio liittyy mallin sopivuuteen nimenomaan julkishallinnollisen organisaation kypsyiden arviointiin. Kuten sanottua SAFe MM -mallia voi käyttää soveltaen julkishallinnollisen organisaation kypsyystason arviointiin, mutta tiettyjen periaatteiden ja käytänteiden arvioiminen on haastavaa organisaatiokontekstista johtuen. Esimerkiksi asiakasyhteistyö -periaatteeseen kuuluu monia sellaisia käytänteitä, joiden arvioiminen voi olla vaikeaa tai jopa mahdotonta. Esimerkiksi asiakkaan määrittäminen voi olla haastavaa, sillä asiakkaita on usein monia eri tahoja, eri asiakkaiden tahtotila kehitettävän ohjelmiston suhteen voi olla eri ja organisaatio ei voi välttämättä itse päättää, kuka asiakkaita on kaikkein tärkein.

5.3 Tutkimuksen rajoitteet

Tässä tutkielmassa on monta tutkimusrajoitetta, jotka tulisi huomioida. Ensinnäkin kohdeorganisaation ketteryyden arvioiminen rajautuu organisaation IT-liitännäiseen tekemiseen, eli arvioinnissa ei ole huomioitu koko organisaatiota. Lisäksi organisaation sisäinen tilaaja-toimittajamalli saattaa vaikuttaa siihen, että tuloksia ei voi välttämättä soveltaa esimerkiksi muihin julkishallinnollisiin organisaatioihin. Tutkimus keskittyy luomaan yleiskuvan ketteryyden omaksumisen tasosta julkishallinnollisessa organisaatiossa, eli tutkimus ei ota tarkemmin kantaa siihen, kuinka hyvin organisaatio on omaksunut oman ketterän viitekehyksen tai muiden tunnettujen ketterien menetelmien, kuten Scrumin, Kanbanin tai SAFen viitekehyksiä.

Tutkielmassa on pyritty arvioimaan ketterien arvojen ja periaatteiden omaksumista Turetkenin ym. (2016) SAFe MM -mallin pohjalta, mutta mallia ei ole suunniteltu julkishallinnollista organisaatiota varten. Kuten luvussa 2.4. ”Julkishallinnon erityispiirteet ja haasteet ketterissä menetelmissä” on esitelty, julkishallinnollisiin organisaatioihin liittyy ominaispiirteitä, jotka vaikuttavat ketterien menetelmien omaksumiseen. Eli lähtökohtaisesti tutkimuksen tulokset eivät ole välttämättä sovellettavissa kaikkiin yritysmaailman organisaatioihin.

Tutkimusmenetelmän valinta ja käyttö luovat tutkimukselle myös rajoitteita. Tässä tutkielmassa tapaustutkimuksen käyttäminen antaa kuvan yhden organisaation ketteryyden omaksumisesta, mutta tutkielman tulosten yleistettävyyden tulee ennemminkin SAFe MM -mallin käytettävyydestä. Tosin

tässäkin pitää huomioida, että kypsyystasomalli soveltuu eri organisaatiokonteksteihin eri tavalla.

Aineistonkeruumenetelmänä käytettiin teemahaastattelua. Haastateltavien määrän ollessa kahdeksan yli 800 hengen IT-palvelujen organisaatiossa, vastaukset eivät voi millään edustaa kaikkia IT-palvelujen tiimejä, joten tämä tulisi huomioida tutkimuksen tuloksia lukiessa. (Kela, 2022d). Lisäksi kaikista IT-palvelujen kymmenestä yksiköstä ei ole haastateltu vähintään yhtä henkilöä (Kela, 2022b). Toisaalta koko IT-palvelujen tulosityksikkö vastaa IT:n kehittämisestä, ylläpidosta sekä tuotantopalvelujen tuottamisesta, mikä pitää sisällään ohjelmistokehityksen ja -tuotannon ulkopuolisia töitä (Kela, 2022c). Tällä perusteella kaikista yksiköistä ei olekaan välttämättä järkevää tutkia ketterän kehittämisen arvojen ja periaatteiden omaksumista. Haastateltavien määrän kasvattaminen olisi voinut antaa laajemman ja tarkemman kuvan organisaation ketteryuden omaksumisesta, mutta pro gradu -tutkielman rajoissa tätä ei nähty järkevänä.

Haastateltavien valinta on vaikuttanut olennaisesti tutkimuksen tuloksiin, sillä haastateltaviksi valittiin sellaisia henkilöitä, joiden ajateltiin ymmärtävän, minkälaisia onnistumisia ja haasteita Kelalla on ketteryuden suhteen. Tämä tarkoittaa sitä, että haastateltavat edustavat todennäköisesti sellaisia tiimejä, joissa ainakin haastateltava on ymmärtänyt ketteryuden syvimmän olemuksen. Haastateltavien valinta on voinut vääristää tuloksia kypsyystason kannalta parempaan suuntaan, sillä haastateltavien tiimit voivat edustaa organisaatiossa ketteryuden omaksujissa edelläkävijöitä. Tuloksia lukiessa tulisikin huomioida se, että tulokset edustavat näiden yksittäisten haastateltavien havaintoja ja huomioita organisaatiossa tietyssä ajan hetkessä. Aineiston analysoinnin ja tulosten yhteenvedon välissä ei pidetty haastateltaville jälkihaastatteluja, jotta löydökset olisi voitu vielä vahvistaa.

5.4 Jatkotutkimusaiheita

Organisaation ketteryys, ketterien menetelmien ja skaalautuvien ketterien menetelmien käyttö sekä itsessään myös ketteryuden kypsyystasomallin sopivuus käytäntöön ovat kaikki aiheina hyvin laajoja. Sen vuoksi aiheita voisi tutkia vielä syvällisemmin, esimerkiksi yhden organisaatioyksikön tai julkaisujunan suhteen. Kiinnostavaa olisi tutkia vielä tarkemmin Kelan organisaatiokulttuurin ja ketteryuden arvojen ja periaatteiden suhdetta toisiinsa. Yhtenä tarkempaan jatkotutkimusaiheena voisi olla sen tutkiminen, mitkä tekijät vaikuttavat siihen, että ketteryuden arvojen ja periaatteiden omaksuminen on onnistunut Kelassa vain osittain? Yhtenä omana jatkotutkimusaiheena voisi olla Kelan organisaatiokulttuurin tutkiminen, sillä sen on kuitenkin todettu vaikuttavan ketteryuden omaksumiseen (Tolfo ym., 2009).

Lisäksi SAFe MM -mallin soveltuvuutta muuhun kuin julkishallinnolliseen organisaatioon pitäisi tutkia tarkemmin. Tämä tutkimusaihe voisi antaa paremman käsityksen siihen, miten malli soveltuu erilaisiin

organisaationkonteksteihin ja onko julkishallinnollisilla organisaatioilla sellaisia erityispiirteitä, joita SAFe MM -malli ei huomioi. Jokainen edellä esitetyistä jatkotutkimusaiheista voisi antaa paremman käsityksen siitä, miten ketterä ajattelutapa näyttäytyy organisaatioissa ja kuinka hyvin kehitetyt ketterien menetelmien viitekehukset toimivat käytännössä.

6 YHTEENVETO

Tämän tutkielman tavoitteena oli arvioida julkishallinnollisen organisaation, Kelan, ketteryyden arvojen ja periaatteiden omaksumista. Arvioinnissa hyödynnettiin Turetkenin ym. (2016) SAFe MM -kypsyystasomallia. Tarkemmat tutkimuskysymykset, joihin tutkielma pyrki vastaamaan, olivat:

- Kuinka hyvin Kela on onnistunut omaksumaan ketterien menetelmien arvot ja periaatteet Turetkenin ym. (2016) kypsyystasomallin perusteella?
- Mitkä organisaation osa-alueet vaativat kehittämistä kypsyystasomallin perusteella?
- Soveltuuko Turetkenin ym. (2016) SAFe MM -malli ketteryyden kypsyystason mittaamiseen?

Ketteryyttä ja ketteriä menetelmiä lähestytään usein työkalu ja prosessinäkökulman kautta, vaikka ketteryyden omaksumisessa olisi paljon tärkeämpää ymmärtää ketterän ajatustavan omaksuminen. Ketterän ajattelutavan omaksuminen vaatii organisaatiokulttuurin, joka tukee uudenlaista ajattelutapaa. Tämän ymmärtäminen on ollut keskeistä tutkimuksen tuloksia analysoitaessa.

Empiirinen aineisto kerättiin temahaastattelujen avulla ja aineiston analyysissä hyödynnettiin suunnattua sisällönanalyysiä. Sisällönanalyysin teoreettisena viitekehystenä käytettiin Turetkenin ym. (2016) SAFe MM -kypsyystasomallia. Empiirisen aineiston pohjalta tehtiin yhteensä 69 empiiristä johtopäätöstä ja 6 ensisijaista empiiristä johtopäätöstä, jotka kiteyttävät eri teemojen empiiriset johtopäätökset yhteen.

SAFe MM malliin perustuen Kela on saavuttanut ketteryyden omaksumisessa tason 1 tai 2. Ohjelmistokehityksen julistuksen arvoihin peilaten Kela on onnistunut luomaan yhteistyötä korostavia foorumeita, mutta yksilöiden korostamisessa ja arvostamisessa on vielä kehitettävää. Osassa tiimejä saadaan tuotettua toimivaa ohjelmisto lyhyessä aikavälissä, mutta dokumentaatiota saatetaan tuottaa liikaa vaatimuksiin nähden. Asiakasyhteistyön kohdalla Kelalla on eniten kehittämistä, sillä moni kehitystiimi ei tunnista omaa

asiakastaan. Kelassa muutoksiin reagoiminen on nopeaa, joskus jopa niin nopeaa, että suunnitelmat jäävät luomatta ja suunnitelmien hyödyt jäävät saamatta. Organisaation osa-alueista eniten kehittämistä vaatisivat asiakkaan tunnistaminen, asiakkaan ymmärtäminen sekä asiakasarvon luominen mahdollisimman usein. Myös yksilöiden ja tiimien valmentaminen kohti ketteryyttä tunnistettiin tärkeänä kehittämiskohteena ja tämä pitää sisällään myös linjajohdon valmentaminen siitä, mitä johtajuus tarkoittaa ketterässä toimintamallissa.

Tutkielman puitteissa tunnistettiin myös ketteryyden omaksumista estäviä kulttuurillisia tekijöitä, joita ovat Kelassa johtaminen, asenteet toisia työntekijöitä kohtaan sekä työympäristössä vallitsevat arvot. Ymmärryksen lisääminen esimerkiksi ketteryydestä, automaatiotestauksesta, DevOps-toimintamallista sekä muutosjohtamisesta tunnistettiin myös yhtenä kehitettävänä asiana. Näissä oleellista on huomioida se, että edellä mainittuja asioita kannattaisi kehittää kulttuurinäkökulma edellä eli ensin kannattaa paneutua yhteisiin toimintatapoihin ja prosesseihin, sitten vasta tarkemmin työkaluihin ja ohjeisiin.

Tämän tutkielman perusteella SAFe MM kypsyystasomalli sopii soveltaen julkishallinnollisen organisaation kypsyystason arviointiin, vaikka mallissa tunnistettiinkin muutamia epäkohtia.

LÄHTEET

- Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). Agile software development methods: Review and analysis. arXiv preprint arXiv:1709.08439.
- Alqudah, M., & Razali, R. (2016). A Review of Scaling Agile Methods in Large Software Development. *International Journal on Advanced Science, Engineering and Information Technology*, 6(6), 828-837.
- Atlassian. (6.6.2022). *Tracking your progress*.
<https://confluence.atlassian.com/jirasoftwareserver/tracking-your-progress-938845110.html>
- Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). Manifesto for agile software development.
- Boehm, B., Turner, R. (2003) Rebalancing Your Organization's Agility and Discipline. Teoksessa Maurer, F., Wells, D. (toim.) *XP/Agile Universe 2003*. LNCS, vol. 2753, 1–8. Springer, Heidelberg.
- Cockburn, A. (2002). *Agile Software Development*. Pearson Education, Indianapolis.
- Conboy, K. (2009). Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development. *Information Systems Research*, 20(3), 329-354
- Conboy, K., Coyle, S., Wang, X., & Pikkarainen, M. (2011). People over process: Key challenges in agile development. *IEEE Software*, vol. 28, 48–57.
- Hirsjärvi, S. & Hurme, H. (2008). Tutkimushaastattelu: teemahaastattelun teoria ja käytäntö. Helsinki: Gaudeamus. Hirsjärvi, S., Remes, P. & Sajavaara, P. (2009). Tutki ja kirjoita. (15. uud. painos). Helsinki: Tammi.
- Hogan, R. & Kaiser, R. (2005). What We Know About Leadership. *Review of General Psychology*, 9(2), 169-180
- Hsieh, H. F., & Shannon, S. E. (2005). Three approaches to qualitative content analysis. *Qualitative health research*, 15(9), 1277-1288
- Iveroth, E. (2010), Inside Ericsson: A Framework for the Practice of Leading Global IT-enabled Change. *California Management Review* 53(1), 136-153.
- Kansaneläkelaitos. (2022a). Tarkoitus ja arvo. Haettu 4.6.2022 osoitteesta:
<https://www.kela.fi/tarkoitus-ja-arvot>
- Kansaneläkelaitos. (2022b). Tulosityksiköt. Haettu 15.5.2022 osoitteesta:
<https://www.kela.fi/tulosityksikot>
- Kansaneläkelaitos. (2022c). IT-palvelut. Haettu 29.5.2022 osoitteesta:
<https://www.kela.fi/it-palvelut>

- Kansaneläkelaitos. (2022d). IT-työssä Kelassa. Haettu 29.5.2022 osoitteesta: <https://www.kela.fi/uramahdollisuuksia-it-tyo>
- Kehittämismalliryhmä. (2022). Kehittämismalli: Toiminnan kehittäminen IT-ratkaisuja hyödyntämällä). Kela.
- Laanti, M., & Kettunen, P. (2019). SAFe Adoptions in Finland: A Survey Research. Lecture Notes in Business Information Processing. *International Conference on Agile Software Development*, 364, 81–87.
- Leffingwell, D. (2011a) Agile software requirements: lean requirements practices for teams, programs, and the enterprise, Addison-Wesley Professional.
- Leffingwell, D. (2011b) Scaling Agile Framework (SAFe). Haettu 9.3.2021 osoitteesta <https://www.scaledagileframework.com/>
- Leppänen M. (2013). A Comparative Analysis of Agile Maturity Models. In: Pooley R., Coady J., Schneider C., Linger H., Barry C., Lang M. (eds) *Information Systems Development*. 329-343.
- Measey, P. (2015). The foundations of agile. Teoksessa: Agile Foundations: Principles practices and frameworks, 11-25. Measey, Peter & Radtac. United Kingdom: BCS, The Chartered Institute for IT.
- Moreira, M. (2017). The Agile Enterprise: Building and Running Agile Organizations. Berkeley, California: Apress.
- Myers & Newman. (2007). The qualitative interview in IS research: Examining the craft.
- Myers, M. D. (1997). Qualitative Research in Information Systems. *AISWorld Section on Qualitative Research in Information Systems*. Haettu 29.5.2022 osoitteesta www.qual.auckland.ac.nz
- Nurdiani, I., Börstler, J., Fricker, S., Petersen, K., & Chatzipetrou, P. (2019). Understanding the Order of Agile Practice Introduction: Comparing Agile Maturity Models and Practitioners' Experience. *Journal of Systems and Software*. 156. 10.1016/j.jss.2019.05.035.
- SAFe. (2022). SAFe 5 for Lean Enterprises Framework® 5.1 Haettu 25.5.2022 osoitteesta <https://www.scaledagileframework.com>
- Schwaber, K. (2004). Agile Project Management with Scrum. *Microsoft Press, Redmond, Washington, USA*.
- Tuomi, J. & Sarajärvi, A. (2009). Laadullinen tutkimus ja sisällönanalyysi (6. uud. laitos.). Tammi.
- Turetken, O., Stojanov, I. & Trienekens. (2016). Assessing the adoption level of scaled agile framework: a maturity model for Scaled Agile Framework.
- Wenger, E. (1999). *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press, 1999.

LIITE 1 SAFE MM VIITTAUKSET

Dokumentti (50 kpl)	Dokumenttityyppi	Viittauspaikka
Schuh, G., Rebentisch, E., Dölle, C., Mattern, C., Volevach, G. & Menges, A. (2018). Defining scaling strategies for the improvement of agility performance in product development projects	Konferenssijulkaisu	Taustoituskappale. 2.1. Scalability of Agile Project Management
Razzak, M.A., Noll, J., Richardson, I., Canna, C.N, & Beecham, S. (2017). Transition from Plan Driven to SAFe ®: Periodic Team Self-Assessment	Konferenssijulkaisu	Taustoituskappale. 2 2, Mikä SAFe on?
Nurdiani, I., Börstler, J., Fricken, S., Petersen K. & Chatzipetrou, P. (2019). Understanding the order of agile practice introduction: Comparing agile maturity models and practitioners' experience	Tieteellinen julkaisu	Kypsyystasomallien vertailu
Laanti, M. & Kettunen, P. (2019). SAFe adoptions in Finland: a survey research	Konferenssijulkaisu	Johdantokappale
Razzak, M.A., Richardson, I., Noll, J., Canna, C.N. & Beecham, S. (2018). Scaling agile across the global organization: an early stage industrial SAFe self-assessment	Konferenssijulkaisu	Taustoituskappale 2.3 Scaled Agile Framework (SAFe)
Yürüm, O.R., Demirörs, O. & Rabhi, F. (2018). A comprehensive evaluation of agile maturity self-assessment surveys	Konferenssijulkaisu	Viittaus lähdeluettelossa, ei viittauksia tekstissä.
Almeida, F. & Simões, J. (2019). Moving from waterfall to agile: Perspectives from IT Portuguese companies	Konferenssijulkaisu	Taustoituskappale. 2. Challenges on moving to agile
Putta, A. (2018). Scaling agile software development to large and globally distributed large-scale organizations	Konferenssijulkaisu	Viittaus lähdeluettelossa, ei viittauksia tekstissä.
Kasem, H.M., Hung, K.W. & Jiang, J. (2018). Revised spatial transformer network towards improves image super-resolutions	Konferenssijulkaisu	Virheellinen viittaus, ei viittauksia tekstissä tai lähdeluettelossa.
Marnewick, C. & Marnewick A. (2019). Insights into managing project teams for industry 4.0	Kirja	Ei pääsyä aineistoon.
Mucambe B., Tereso, A.P., Faria, J.M.P. & Mateus, T. (2019). Large-Scale Agile Frameworks-dealing with interdependences	Konferenssijulkaisu	Taustoituskappale. SAFe - Scaled Agile Framework
Kil, T., Koo, H.I. & Cho, N.I. (2018). Scene text rectification using glyph and character alignment properties	Konferenssijulkaisu	Virheellinen viittaus, ei viittauksia tekstissä tai lähdeluettelossa.

Almeida, F. & Monteiro, J.A. (2018). UX challenges and best practices in designing web and mobile solutions	Kirja	Ei pääsyä aineistoon.
Limaj, E. & Bernroider, E.W.N. (2019). A Systematic Analysis and Synthesis of Case Study Based Agile Scaling Research in the Context of Digital Transformations	Konferenssijulkaisu	Ei pääsyä aineistoon.
Nurdiani, I. (2018). Introducing of Agile Practices: Strategies and Impacts	Väitöskirja	Yhteenveto kypsyystasomalleista
Kiruba, N.R., Uma Devi, S. & Mohamed, S. (2020). A Proposal on Developing a 360° Agile Organizational Structure by Superimposing Matrix Organizational Structure with Cross-functional Teams	Tieteellinen julkaisu	Taustoituskappale. SAFe®: Scaled Agile Framework
Männikkö, A. (2020). Agile Development model in Multi-project Environment	Pro gradu - tutkielma	Taustoituskappale. 2.4.9. Scaled Agile Framework (SAFe)
Kasauli, R.N. (2020). Requirements Engineering that Balances Agility of Teams and System-level Information Needs at Scale	Väitöskirja	Taustoituskappale. 1.1.3. Large-scale Agile Systems Development
Sands, O. & Selling, A. (2020). Implementing customer behaviour data analysis in a scaled agile context. A case study of software development challenges at SKF	Pro gradu - tutkielma	Taustoituskappale. 2.4.4 Scaled agile
Silva, D.G.O.A. (2020). Application of SAFe® to an Aircraft Maintenance	Väitöskirja	SAFea koskeva kirjallisuus
Hussain, W., Shahin, M., Hoda, R., ... Oliver, G. (2021). Assessing the Adoption Level of Agile Development Within Software Product Lines: The AgiPL-AM Model	Konferenssijulkaisu	Kypsyystasomallien vertailu
Tuncel, D., Körner, C. & Plösch, R. (2020). Comparison of Agile Maturity Models: Reflecting the Real Needs	Konferenssijulkaisu	Kypsyystasomallien vertailu
Haidar, H., Kolp, M. & Wautelet, Y. (2019). Assessing the Adoption Level of Agile Development Within Software Product Lines: The AgiPL-AM Model	Konferenssijulkaisu	Kypsyystasomallien vertailu
Trienekens, J., Kusters, R., Himawan, H.B. & van Moll, H. (2018). Customer Involvement in the Scaled Agile Framework	Konferenssijulkaisu	Johdantokappale
Hinterberg, L. & Hoffman, F. (2018). Exploring the Scaled Agile Framework in a Virtual Team Setting	Pro gradu - tutkielma	Johdantokappale. Taustoituskappaleet: 2.3. Scaling agile frameworks, 2.4 The Scaled Agile Framework (SAFe), 2.6 Benefits of agile frameworks, 5.2

		Theme 2: Perceived benefits
Korraprolo, S.A. (2018). Evaluation of the Relevance of Agile Maturity Models in the Industry: A Case Study	Pro gradu tutkielma	- Kypsyystasomallien vertailu
Salma, F. & Gómez, J.M. (2021). Challenges and Trends of Agile	Kirja	Ei pääsyä aineistoon
Al-Matouq, H., Mahmood, S., Alshayeb, M. & Niazi, M. (2020). A Maturity Model for Secure Software Design: A Multivocal Study	Tieteellinen julkaisu	Taustoituskappale: D. Software Maturity Models
Sievers, J. (2017). Agile Control and Collaboration in Large Organizations	Pro gradu tutkielma	- Taustoituskappale: 3.6 Scaled Agile
Kärki, A. (2020). The Impact of Education and Training in Large-scale Agile Transformation	Pro gradu tutkielma	- Taustoituskappale: 2.3 Large-scaled agile transformation success and failure factors
Pinciroli, M. (2019). Assessing the impact of business agility model on smart attitude of people: an empirical analysis.	Pro gradu tutkielma	- Taustoituskappale: 1.4.4 Boundaries and Challenges
Blewden, A.J. (2020). Agile Transformation for Business Teams within New Zealand Organisations	Pro gradu tutkielma	- Taustoituskappale: 2.8.2. ore recent Scaled Empirical Studies and Frameworks
Kuiper, C.J. (2019). Relationship of Transformational Leadership and Organizational Change During Enterprise Agile and DevOps Initiatives In Financial Service Firms	Väitöskirja	Taustoituskappaleet: Reduction of Gaps, Entering Scale ja Reflections
Collins, W. (2020). Understanding Scaled Agile Framework Coordination Methods for Reducing Failure Rates	Väitöskirja	Taustoituskappale: Large scale to scalability
Petit, Y. & Marnewick, C. (2021). Strategic Alignment of Information Technology Initiatives in a Scaled Agile Environment	Tieteellinen julkaisu	Ei pääsyä aineistoon
Aghajani, M. (2019). A Qualitative Study of Success Factors for Portfolios of Agile Projects in Large Organizations	Väitöskirja	Taustoituskappaleet: 1.1.3. Scaling agile methods to PPM domain, 1.2 The statement of the problem, 2.2.2 Agile scale-up frameworks, 6.6 Agile scale-up practices
Paajoki, A. (2020). Best practices for and benefits from implementing ISPMAs SPM framework	Pro gradu tutkielma	- Taustoituskappale: 2.5.1 Scaled Agile Framework (SAFe)
Wahl, E. (2020). Reflecting and adjusting in large-scale Agile software development: A case study	Pro gradu tutkielma	- Taustoituskappale: Related work
Kapic, C. (2019). Managing the Agile Scalability to implement Agile Project	Pro gradu tutkielma	- Johdantokappale

Portfolio Management: A Case Study within the Automotive Industry		
Wright, D. (2018). Best Practices for Large-Scale Agile Transformations	Pro gradu - tutkielma	Taustoituskappale: Problem statement, Challenges of Large-Scale Agile Implementations, Best Practices of Large-Scale Agile Implementations
Murthy, S.S.M. (2020). Qualitative Study of Scaled Agile Framework: An Organizational Approach	Väitöskirja	Johdantokappale, taustoituskappale: business problem
Sun, Q., Jin, X. & Kang, Y. (2018). Assessment of Cybersecurity Effectiveness in Serving Maqasid Al-Sariah	Tieteellinen julkaisu	Taustoituskappale: B.Cybersecurity capability maturity model
Pountney, P. (2017). Agile Beyond the Sweet Spot Insights and Recommendations.	Tieteellinen julkaisu	Johdantokappale, taustoituskappale: 3.2.1. Project Scale and Complexity
Nilssons, T. (2020). A Transparent Agile Change: Predicting a Transparent Organizational Change from Change Recipients' Beliefs and Trust in Management	Pro gradu - tutkielma	Johdantokappale
Ward, R. & Chang, C.K. (2019). In Search of Scientific Agile	Konferenssijulkaisu	Taustoituskappale: C. Practice Variations
Rode, A.L.G., Svejvig, P., Holm, C.H. & Rosenlund, S.L. 2021. Project Half Double.		7.1.3. Diffusion of agile project management
Rasch, L. & Thun, V. (2020). The Road to Become Agile: A case study of agile transformations in the retail market, including organization development approach	Pro gradu - tutkielma	Taustoituskappale: 3.1 Maturity Model Practices
Ölander, S. (2019). DevOpsin kypsyyden mittaust	Opinnäytetyö	Taustoituskappale: 3.4 Ketterät menetelmät
Akhter, T. & Åkerlind, T. (2018). Agile IT Infrastructure Transformation: A Case Study of a Nordic Incumbent Telco	Pro gradu - tutkielma	Taustoituskappale: 2.4 Scaled Agile Framework (SAFe)
Petersson, U. (2020). Metrics for agile software development: Support for the agile team to continuously evaluate and improve its performance	Pro gradu - tutkielma	Taustoituskappale: 2.4 Scaled Agile Framework (SAFe)

LIITE 2 SAFE MM -MALLI: TASOT, PERIAATTEET JA KÄYTÄNNÖT (TURETKEN YM., 2016)

Periaatteet		
	Muutoksen omaksuminen asiakasarvon tuottamiseksi	Ohjelmistojen suunnittelu ja ihmiskeskeisyys toimittaminen usein
	Tekninen huippuosaaminen	Asiakasyhteistyö
Taso 5	L5P1: Matala prosessieronomia L5P2: Jatkuva SAFe kyvykkyyksien parantaminen	L5P3: Ketterien projektien arviointi L5P4: Ihanteellinen fyysinen ympäristö ketterälle L5P5: Organisaation muuttaminen
	L5P6: Testilähtöinen kehittäminen	L5P7: Tason -1 tai 1b ihmisiä tapahtuva vuorovaikutus kehittäjien ja käyttäjien välillä
	L5P9: Toistuva kasvokkain	L5P8: Rinnakkainen testaaminen
Taso 4	L4P1: Asiakaslähtöiset iteraatiot L4P2: Jatkuva asiakastytyväisyyspalaute L4P3: Leanin vaatimukset skaalautuvassa mittakaavassa	L4P4: Pienemmät, useammat julkaisut L4P5: Adaptiivinen suunnittelu L4P6: Liiketoiminnan tehokkuuden arvioiminen
	L4P7: Hajautettujen tiimien hallinta	L4P8: Tarkoituksenmukainen arkkitehtuuri
	L4P9: Päivittävät edistymisen seurantakokoukset	L4P10: Asiakas on heti käytettävissä L4P11: Asiakasyhteys pyörii yhteistyön sitoutumisen ympärillä
Taso 3	L3P1: Säännöllinen reflektointi ja sopeutuminen	L3P2: Riskilähtöiset iteraatiot L3P3: Ominaisuuksien, eikä tehtävien suunnittelu L3P4: Tiekartta L3P5: Iteraation hallitseminen L3P6: Kanbanin hyödyntäminen L3P7: PSI / julkaisu L3P8: Julkaisujuna
	L3P9: Itseohjautuvat tiimit	L3P10: Säännöllinen henkilökohtainen vuorovaikutus L3P11: Scrum
	L3P12: Jatkuva integraatio	L3P13: Jatkuva parantaminen (refaktorointi)
	L3P14: Yksikkötestit	L3P15: 30 % tason 2 ja tason 3 henkilöitä
	L3P16: DevOps	L3P17: Visio, ominaisuudet L3P18: Vaikutus asiakkaisiin ja toimintaan
Taso 2	L2P1: Vaatimusten evolutiivisuus L2P2: Pienemmät, useammat julkaisut L2P3: Vaatimusten löytäminen	L2P9: Sovellus-kokoonpanon hallinta L2P10: Automaatiotestaus L2P11: Iteraation edistymisen seuranta L2P12: Ei isoa suunnittelua etukäteen L2P13: Tuotteen kehitysjono
	L2P4: Jatkuva tuottaminen	L2P14: Asiakassopimus heijastaa evolutiivista kehitystä
	L2P5: Kahden tason suunnittelu ja seuraaminen	L2P15: Iteraation edistymisen seuranta
	L2P6: Ketterä arviointi ja kiertonopeus	L2P16: Ei isoa suunnittelua etukäteen
	L2P7: Julkaisun suunnittelu	L2P17: Tuotteen kehitysjono
Taso 1	L1P1: Prosessin reflektointi ja parantaminen	L1P2: Yhteissuunnittelu L1P3: Voimaantuneet ja motivoituneet tiimit L1P4: Yhteistyötä tekevät tiimit
	L1P5: Koodausstandardit	L1P6: Tiedon jakaminen
	L1P7: Vapaaehtoisuus	L1P8: Hyväksymistestaus
	L1P9: Käyttäjätarinat	L1P10: Asiakas sitoutuu työskentelemään kehitystiimin kanssa

LIITE 3 TIETOA TUTKIMUKSESTA HAASTATELTAVILLE

Tietoa tutkimuksesta

Tutkimuksen tarkoituksena on kartuttaa ymmärrystä siitä, kuinka hyvin Kela on omaksunut ketteryiden arvot ja periaatteet. Tutkimuksessa keskitytään tarkastelemaan Kelan ketteryyden omaksumista kokonaislaajuisesti organisaation eri tasoilla. Tutkimus on rajattu Kelan IT-liitännäiseen tekemiseen.

Tutkimuksen aineistona toimivat teemahaastattelut, joita kerätään vuosien 2021-2022 aikana. Haastateltaviksi on valittu henkilöitä, joiden katsotaan ymmärtävän, minkälaisia onnistumisia ja haasteita Kelalla on ketteryyden suhteen.

Kunkin haastattelun kesto on arvioilta noin tunnin, riippuen kuitenkin siitä kuinka paljon kerrottavaa kullakin haastateltavalla on. Haastattelut saattavat kestää myös alle tunnin tai venyä yhden tunnin. Kerätty haastatteluaineisto litteroidaan ja aineistoa säilytetään, kunnes lopputyö on hyväksytty. Aineistoa ei käytetä lisätutkimuksiin ilman asianomaisten erillistä suostumusta.

Haastatelluilla on mahdollisuus lukea tutkimus ennen sen julkaisua ja korjata mahdollisia väärinymmärryksiä, joita heidän sanomisistaan on voinut tulla. Haastatelluille annetaan mahdollisuus lukea myös valmis tutkimus.

Tutkittavilla on tutkimuksen aikana oikeus kieltäytyä tutkimuksesta ja keskeyttää tutkimukseen osallistuminen missä vaiheessa tahansa ilman, että siitä aiheutuu heille mitään seuraamuksia. Tutkimuksen järjestelyt ja tulosten raportointi ovat luottamuksellisia. Tutkimuksesta saatavat tutkittavien henkilökohtaiset tiedot tulevat ainoastaan tutkittavan käyttöön ja tulokset julkaistaan tutkimusraporteissa siten, ettei yksittäistä tutkittavaa voi tunnistaa. Tutkittavilla on oikeus saada lisätietoa tutkimuksesta missä vaiheessa tutkimusta tahansa.

Tutkimus tehdään Lotta Kallosen pro gradu -lopputyönä Kelalle.

LIITE 4 SUOSTUMUSLOMAKE HAASTATELTAVILLE**Kirjallinen suostumus tutkimukseen osallistumisesta****Suostumus**

Suostun vapaaehtoisesti osallistumaan Lotta Kallosen pro gradu -tutkimukseen. Minulle on kerrottu ketkä aineistoa käsittelevät, miten aineistoa käsitellään ja miten aineisto säilytetään. Suostun haastattelun tai haastatteluiden nauhoittamiseen.

Minulle on kerrottu mitä varten aineistoa kerätään.

Voin keskeyttää osallistumiseni missä vaiheessa tahansa.

Annan suostumukseni tutkimuksen tekemiseen, haastattelujen nauhoittamiseen sekä litteroinnin hyödyntämiseen tutkimuksessa.

Aika ja paikka

Osallistujan allekirjoitus

Nimenselvennys