

Anttoni Tukia

**IHMISTEN JOHTAMINEN KETTERISSÄ
OHJELMISTOPROJEKTEISSA**
Systemaattinen kirjallisuuskatsaus

Informaatioteknologian ja viestinnän tiedekunta
Kandidaattitutkielma
Syyskuu 2020

TIIVISTELMÄ

Anttoni Tukia: Ihmisten johtaminen ketterissä ohjelmistoprojekteissa – systemaattinen kirjallisuuskatsaus
Kandidaattitutkielma
Tampereen yliopisto
Tietotekniikan tutkinto-ohjelma
Syyskuu 2020

Tämän systemaattisen kirjallisuuskatsauksen tarkoituksena on tarkastella ihmisten johtamista ketterissä ohjelmistoprojekteissa. Katsauksessa pyritään kuvamaan ihmisten ketterää johtamista sekä etsimään yhteisiä teemoja aiemmista aineistoista aiheeseen liittyen. Tuotettu tieto auttaa ymmärtämään ketterää ohjelmistokehitysprojektia ja auttaa lukijaa sekä toiminaan siinä että johtamaan sitä. Tiedosta voivat hyötyä tietotekniikan opiskelijat tai muut tahot, jotka tulevat toimimaan tai toimivat ketterissä kehitysympäristöissä.

Tutkimusaineisto on kerätty systemaattisella kirjallisuuskartoituksella kahdeksasta eri tietokannasta syyskuussa 2020. Aineistoja katsaukseen valittiin yhteensä 20 ja ne koostuivat kirjojen kapaleista ja tutkimusartikkeleista. Aineistojen keskeisimpänä valintakriteerinä oli, että sen tuli käsitellä ihmisten johtamista ketterässä ohjelmistoympäristössä projektinhallinnan näkökulmasta.

Tutkimustulosten mukaan ketterä ihmisten johtaminen edellyttää johtajalta johtajuuden ja päätöksentekovallan jakamista kehitystiimin jäsenille. Tiimin jäseniltä tämä taas edellyttää itseluottamusta ja oma-aloitteisuutta tehdä päätöksiä itsenäisesti ja ottaa tarvittaessa vastuuta johtamisesta. Tällaisessa toimintamallissa korostuu luottamus sekä johtajan ja tiimin välillä että tiimin sisäisesti jäsenien välillä. Vaikka ideaalilanteessa tiimi kykenee toimimaan itsenäisesti, johtajan tulee kuitenkin huolehtia, että ohjelmistoprojekti palvelee sen laajempaa päämäärää. Aineistoista löydetyt yhteiset teemat olivat kollaboraatio ja valtuutus, johtajan rooli fasilitaattorina ja valmentajana, kehitystiimin itseorganisointi, oppimisen tukeminen ja palaute sekä ihmiskeskeinen toiminta.

Avainsanat: ihmisten johtaminen, ketterä ohjelmistokehitys, systemaattinen kirjallisuuskatsaus

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck –ohjelmalla.

Sisällysluettelo

1	Johdanto	1
2	Tutkimusmenetelmä	3
2.1	Tiedonhaku	3
2.2	Sisällönanalyysi	8
3	Tulokset	9
3.1	Ihmisten johtaminen ketterissä ohjelmistoprojekteissa	9
3.2	Yhteiset teemat	10
3.2.1	Kollaboraatio ja valtuutus	12
3.2.2	Johtajan rooli fasilitaattorina ja valmentajana	13
3.2.3	Kehitystiimin itseorganisointi	13
3.2.4	Oppimisen tukeminen ja palaute	14
3.2.5	Ihmiskeskeinen toiminta	15
3.3	Johtopäätökset	15
4	Pohdinta	16
5	Yhteenveto	17
	Lähteet	18
	Liitteet	21

1 Johdanto

Ohjelmistoala on tunnettu sen nopeasta kehityksestä ja dynaamisuudesta. Sekä työkalut, tekijät että toimintatavat muuttuvat jatkuvasti. Tämä edellyttää alan organisaatiolta ketteryyttä muuttaa toimintatapojaan ja oppia uutta, sillä tekniikan kehityksen mukana pysyminen on ohjelmistoalan organisaatiolle elintärkeää.

Perinteiset projektinhallinnan mallit kuten vesiputousmalli, joissa määrittely ja suunnittelu tehdään kokonaan ennen varsinaista toteutusta ovat osoittautuneet alati muuttuvalla ja kehittyvällä ohjelmistoalalla liian kankeiksi. Kuukausien tai jopa vuosia kestäviä ohjelmistoprojekteja on todella vaikea määrittellä ja suunnitella kokonaan ennen toteuttamista, sillä jo projektin aikana ala voi kehittyä valtavasti. Tämä taas saattaa usein johtaa siihen, että projektin alussa tehdyt määrittelyt eivät vastaa enää asiakkaan tarpeita ja tuottaa kehitystiimille paljon turhaa työtä. (Sánchez 2018, s. 6)

Vastauksena edellä mainittuihin perinteisten ohjelmistokehityksen prosessimallien haasteisiin organisaatiot ovat ottaneet käyttöönsä ketterän ohjelmistokehityksen. Sen ideana on toteuttaa projekti pienissä iteraatioissa eli sprinteissä. Määrittelyä ei tehdä kokonaan alussa, vaan sitä toteutetaan ja täydennetään projektin edetessä. Jatkuva määrittely mahdollistaa asiakkaan muuttuvien tarpeiden huomioimisen paremmin sekä parantaa asiakkaan mahdollisuuksia vaikuttaa projektin toteutukseen ja sitä kautta sen lopputulokseen. Lyhyet iteraatiot mahdollistavat myös uusien työkalujen, tekniikoiden ja toimintatapojen käyttöönoton kesken projektin. (Beck et al. 2001)

Ketterän ohjelmistokehityksen periaatteisiin kuuluu myös matala hierarkia ja jaettu vastuu johtamisesta ja päätöksenteosta. Tämä tarkoittaa käytännössä sitä, että ketterässä ohjelmistoprojektissa ei ole yhtä projektipäällikköä johtamassa ja tekemässä päätöksiä, vaan johtaminen ja päätöksenteko delegoidaan kaikille projektin jäsenille tittelistä riippumatta. (Moe et al. 2009, s. 2)

Tämän työn tavoitteena on tutkia ihmisten johtamista ketterissä ohjelmistoprojekteissa systemaattisen kirjallisuuskatsauksen keinoin ja etsiä olemassa olevan kirjallisuuden avulla yhteinen näkemys ihmisten ketterästä johtamisesta ohjelmistoprojekteista. Tutkimuskysymykset ovat seuraavat:

1. Kuinka ihmisiä johdetaan ketterissä ohjelmistoprojekteissa?
2. Mitä yhteisiä teemoja aiemmasta kirjallisuudesta löytyy ihmisten johtamisesta ketterissä ohjelmistoprojekteissa?

Kerätyn tiedon avulla ketterissä ohjelmistoympäristöissä toimivat saavat ymmärryksen siitä, miten ihmisiä johdetaan ketterässä ohjelmistoprojektissa ja millaisia teemoja siihen liittyy. Tämä taas auttaa toimimaan ketterässä kehitysympäristössä ja ymmärtämään paremmin ketterän ohjelmistokehityksen prosessia.

Suomen kielen sana ”johtaminen” voi tarkoittaa kahta englanninkielistä eri asiaa: ”leadership” eli ihmisten johtaminen, sekä ”management” eli asioiden johtaminen. Tämä tutkimus on rajattu pelkästään ihmisten johtamiseen ketterissä ohjelmistoprojekteissa, eikä se suoranaisesti ota kantaa asioiden johtamiseen.

Toisessa luvussa kuvataan katsauksessa tehtyä tiedonhakuprosessia, esitellään katsaukseen valikoituja aineistoja sekä kerrotaan sisällönanalyysistä. Kolmannessa luvussa esitellään tutkimustulokset, johon kuuluu ihmisten ketterän johtamisen kuvaaminen sekä aineistoista löydettyjen yhteisten teemojen esittely. Neljännessä luvussa pohditaan katsauksen onnistumista, sen relevanssia ja mahdollisia jatkotutkimuskohteita. Viimeisessä eli viidessä luvussa kerrataan tutkimuskysymykset ja -tulokset yhteenvedon muodossa.

2 Tutkimusmenetelmä

2.1 Tiedonhaku

Opinnäytetyö on systemaattinen kirjallisuuskatsaus, jonka tarkoituksena on etsiä ja yhdistellä tietoa ihmisten johtamisesta ketterissä ohjelmistoprojekteissa. Tiedonlähteinä käytettiin valikoituja tieteellisiä lähteitä, joita yhdistelemällä pyrittiin etsimään yhteisiä teemoja ihmisten johtamisesta ketterissä ohjelmistoprojekteissa. Systemaattista kirjallisuuskatsausta tehdessä sovellettiin prosessimallina Finkin (2014) mallia, jonka pohjalta katsaus tehtiin seuraavia vaiheita noudattaen:

1. Tutkimuskysymysten asettaminen
2. Tietokantojen valinta
3. Hakusanaryhmien valinta
4. Seulojen asettaminen
 - Kieli
 - Otsikko
 - Tiivistelmä
5. Katsauksen suorittaminen
6. Tulosten synteesointi.

Tiedonhaku suoritettiin syyskuussa 2020. Katsaukseen valitut tietokannat olivat seuraavat:

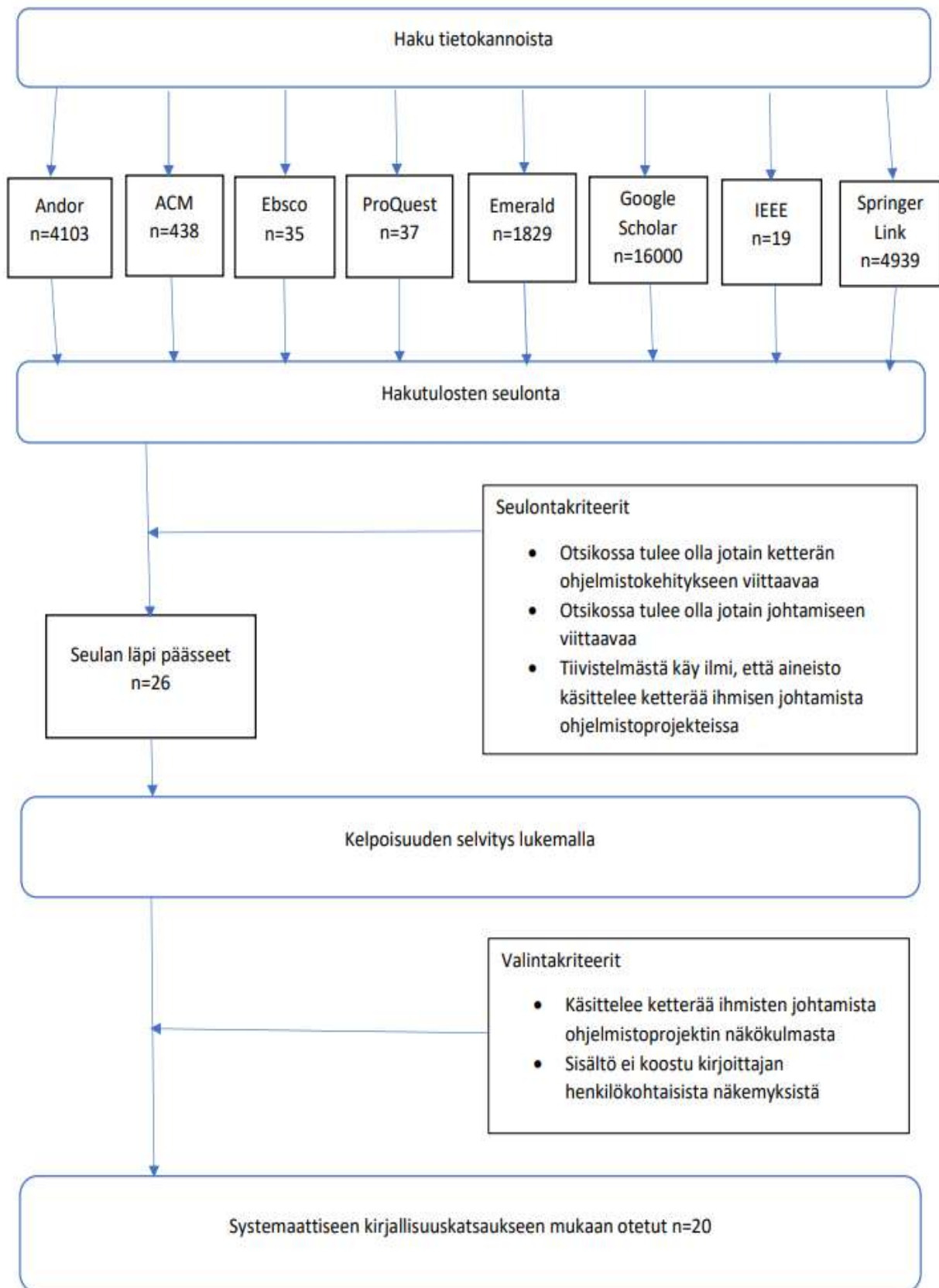
- ACM Digital Library
- Andor
- Applied Science & Technology Source (Ebsco)
- Computer Science Database (ProQuest)
- Emerald Insight
- Google Scholar
- IEEE/IET Electronic Library
- SpringerLink.

Tiedonhakuun käytetyt hakusanaryhmät olivat kaikissa tietokannoissa samat. Katsauksessa käytetyt hakusanaryhmät olivat seuraavat:

- leadership AND “agile software development”
- “agile leader*” AND “software development”.

Katsaukseen mukaan otetut aineistotyypit olivat tutkimusartikkeleja ja kirjoja. Artikkelien osalta rajauksen ensimmäisenä ehtona oli, että artikkelin yhteys ketterään johtamiseen sekä ohjelmistoalaan tuli käy ilmi sen otsikossa. Toisena ehtona oli, että tiivistelmässä kerrotaan artikkelin käsittelevän ihmisten johtamista ketterissä ohjelmistoprojekteissa projektinhallinnan näkökulmasta. Sekä artikkelien että kirjojen osalta seulontaa tehtiin myös hakemalla sanaa ”leadership”, jonka avulla pystyttiin suhteellisen nopeasti arvioimaan aineiston relevanssi katsauksen osalta.

Aineiston kielen tuli olla suomi tai englanti, mutta käytännössä kaikkien aineistojen ollessa hakusanoilla englanninkielisiä, kielelliselle seulonnalle ei ollut tarvetta. Seulonnan jälkeen aineistojen kelpaavuus arvioitiin vielä lukemalla aineisto ja todennettiin, että käsitteleekö se haluttua aihepiiriä. Tiedonhakuprosessia on havainnollistettu alla kuvassa 1.



Kuva 1. Tiedonhakuprosessi.

Tiedonhaku toteutettiin tietokanta kerrallaan ja hakusanoja yhdisteltiin AND-komenolla. Haku aloitettiin Tampereen yliopiston Andorista, jonka 4103 hakutuloksesta kaksi tutkimusartikkelia ja neljä kirjaa otettiin katsaukseen mukaan. Vastaavasti Google Scholarin yli 16000 hakutuloksesta mukaan otettiin vain kolme tutkimusta, Emeraldin 1829 tuloksesta 2 ja ACM:n 428:n joukosta vain yksi. Huomioon otettavaa on, että mikäli tietokannasta löytyi kyseenomaisilla hakusanoilla yli 100 tulosta, hakutuloksista huomioitiin vain 100 ensimmäistä.

Muutamasta tietokannasta löytyi otsikon ja tiivistelmän perusteella lupaavia aineistoja, jotka kuitenkin lukemisen jälkeen hylättiin. Hylkäyksen syitä olivat:

- Aineisto ei käsittele ihmisten johtamista vaan asioiden johtamista
- Aineisto ei käsittele ihmisten johtamista projektinhallinnan näkökulmasta
- Aineisto käsittelee kirjoittajan henkilökohtaisia mielipiteitä.

Yllä mainituista syistä muun muassa kaikki Ebscosta ja ProQuestistä löytyneet aineistot hylättiin. Myös IEEE:n viidestä alkuseulan läpi päässeestä artikkelista 4 hylättiin ja vain yksi otettiin katsaukseen mukaan. Vastaavasti SpringerLinkin kahdeksasta seulan läpi päässeestä aineistosta yksi hylättiin ja seitsemän otettiin katsaukseen mukaan.

Valintakriteerien seurauksena katsaukseen valikoitui yhteensä 20 tutkimusartikkelia ja kirjaa. Valitut artikkelit löytyivät Tampereen yliopiston Andorista, ACM:stä, IEEE:stä, Google Scholarista, SpringerLinkistä ja Emeraldista. Katsaukseen valitut aineisto olivat julkaistu aikavälillä 2008–2020. Niistä 7 oli kirjoja ja 13 tutkimusartikkeleita. Katsaukseen valikoidut kirja on esitelty taulukossa 1 ja katsaukseen valikoidut tutkimusartikkelit liitteessä 1.

Taulukko 1. Systemaattiseen kirjallisuuskatsaukseen valikoituneet kirjat

Viite	Kuvaus
Appelo (2010)	Antaa ohjeita ja näkökulmia ketterän johtamisen käytännön toteuttamiseen
Crowder ja Friess (2014)	Auttaa ymmärtämään ketterää ihmisten johtamista ja sen psykologiaa
Dubinsky ja Hazzan (2008)	Kertoo ihmisten johtamisesta yleisesti sekä ihmisten johtamisesta ketterissä ohjelmistoprojekteissa
Levine (2020)	Auttaa ketterässä ohjelmistokehitysympäristössä työskenteleviä oppimaan ketteriksi johtajiksi
Medinilla (2012)	Tarjoaa kokonaisvaltaisen ymmärryksen, mitä ketteryys tarkoittaa johtotehtävissä työskentelevälle sekä kuinka ketteriä kehitystiimejä johdetaan ja motivoidaan
Stober ja Hansmann (2009)	Kertoo ketterästä ohjelmistokehityksestä ja sen hyvistä käytännöistä
Whitaker (2010)	Tarjoaa tietoa johtamisen hyvistä käytännöistä projektipäälliköille ja muille ohjelmistoprojekteissa johtotehtävissä työskenteleville

2.2 Sisällönanalyysi

Systemaattisen kirjallisuuskatsauksen viimeisenä vaiheena tehdään aineistojen analyysin pohjalta synteesi. Aineistojen sisällölle tehtiin alustavat analyysit jo valitsemisvaiheessa, mutta tarkempi sisällönanalyysi aineistoille tehtiin niiden valikoitumisen jälkeen. Sisällönanalyysi on aineistojen analyysitapa, jonka tarkoituksena on tuoda esiin aineistojen keskeisimmät merkitykset (Weber 1990, s. 2). Katsauksessa sisällönanalyysi on tarkoituksenmukainen, sillä aineistojen keskeisimmät asiat halutaan kuvata tiivistetysti sekä sanallisesti että visuaalisesti.

Analyysin keinon löydettyjä yhteneväisyyksiä, yhteisiä ilmaisuja sekä ihmisten johtamisen piirteitä ryhmiteltiin yhteisiksi teemoiksi. Huomionarvoista on, että tähän katsaukseen kerättiin enemmän tietoa tutkimusartikkeleiden kirjallisuuskatsauksista, kuin itse tutkimusten tuloksista. Luvussa kolme näitä teemoja esitellään, sekä käytetään katsauksen tulosten esittämisessä.

3 Tulokset

3.1 Ihmisten johtaminen ketterissä ohjelmistoprojekteissa

Ketterään ohjelmistokehitykseen kuuluu ketterä ihmisten johtaminen. Ketterä ihmisten johtaminen voidaan nimensä mukaisesti ajatella ketterämmäksi kuin perinteiset johtamistavat, sillä ketterässä johtamisessa päätöksiä ei tee pelkästään johtaja. Vastuu päätöksenteosta on projektin nimellisen vastuuhenkilön (kuten projektipäällikkö tai scrummaster) sijaan kaikilla projektiorganisaation jäsenillä, ja ideaalitulanteessa päätöksen tekee se jäsen, kenellä on kyseenomaisesta aiheesta paras asiantuntemus. (Karhatsu et al. 2010, s. 2)

Jaettu vastuu päätöksenteosta ja johtamisesta edellyttää projektiorganisaation jäseniltä halukkuutta osallistua päätöksentekoon ja valmiutta kantaa tarvittaessa vastuuta johtamisesta. Tämä voi kuitenkin aiheuttaa ongelmia, sillä projektin jäsenet voivat olla haluttomia ottamaan tarvittaessa vastuuta päätöksenteosta tai johtamisesta, koska he eivät yksinkertaisesti uskalla. Pelko siitä, että vastuu jäisi pysyvästi vastuunottajalle on usein syynä tähän ongelmaan (Moe et al. 2009, s. 5–6). Tämän takia ketterän johtajan tulee toimia tässä asiassa ikään kuin valmentajana ja luoda kehittäjiin uskoa omaan ammattitaitoon sekä kannustaa heitä osallistumaan päätöksentekoon ja johtamiseen (Dubinsky & Hazzan 2008, s. 264).

Jaettu vastuu päätöksenteosta ja johtamisesta edellyttää sitä, että ketterä johtaja luottaa tiimin jäsenien kykyyn tehdä itsenäisesti tarvittavia toimenpiteitä ohjelmistoprojektin edistämiseksi (Crowder & Friess 2015 s. 13). Se edellyttää myös sitä, että tiimin jäsenet ovat johtajan luottamuksen arvoisia. Appelo (2010, luku 7) esittää, että luottamus muodostuu yksinkertaisesti tekemällä asioita, joita on sitoutunut tekemään. Varsinkin kokeuttomissa kehitystiimeissä luottamus tiimin jäsenten välillä voi olla heikkoa, ja tällöin ketterän johtajan tulisi kiinnittää siihen huomiota sekä tehdä toimenpiteitä sen parantamiseksi. Jäsenten välistä luottamusta voidaan parantaa lisäämällä tiimin välisen kommunikoinnin määrää ja parantamalla sen laatua. Tätä voidaan käytännössä toteuttaa muun muassa virallisilla ja epävirallisilla tiimitapaamisilla. Toinen luottamuksen elementti, johon ketterän johtajan tulisi kiinnittää huomiota, on jäsenten sitoutuminen projektiin: mitä sitoutuneempi jäsen on, sitä todennäköisemmin hän tekee sen, mihin on sitoutunut. (Appelo 2010, luku 7)

Appelon (2010, luku 9) mukaan ketterä johtaja vastaa kehitystiimin yhteisen päämäärän määrittelystä. Usein itseorganisoituva kehitystiimi kuitenkin kykenee muodostamaan organisaatiota palvelevan yhteisen päämäärän itsenäisesti, ja tässä tapauksessa ketterän johtajan ei kuulu puuttua tiimin tekemiseen. Yhteinen päämäärä toimii yhdistävänä tekijänä

jäsenien välillä ja motivoi ihmisiä asettamalla heidän työnsä johonkin suurempaan kontekstiin tai visioon. Sen puute voi johtaa sekä johtajan että tiimien jäsenten henkilökohtaisen edun tavoitteluun organisaation edun sijasta. Koska ketterään johtajuuteen kuuluu vahvasti päätöksentekovallan jakaminen tiimin jäsenille, tulee ketterän johtajan kommunikoida yhteinen tavoite kehitystiimille, jotta sen jäsenet voivat ottaa kokonaiskuvan huomioon omassa päätöksenteossaan. Ketterän johtajan tulee myös valvoa ja varmistaa, että tiimin autonominen toiminta palvelee yhteistä päämäärää eikä kenenkään henkilökohtaisia etuja. (Appelo 2010, luku 9)

3.2 Yhteiset teemat

Sisällönanalyysin avulla aineistoista löydettiin useita yhteneväisyyksiä, joiden avulla pystyttiin luomaan niitä kuvaavia yhteisiä teemoja. Löydettyjä teemoja oli yhteensä seitsemän, mutta tutkimukseen niistä valittiin viisi. Ensimmäinen karsittu teema oli ”jaettu johtamis- ja päätöksentekovastuu”, joka jäi pois siksi, että se käytännössä sisältyy teemaan ”kollaboraatio ja valtuutus”. Toinen pois jätetty teema oli ”sopiva ulkoinen autonomia”, joka taas jäi pois siksi, että se ei esiintynyt tarpeeksi monessa aineistossa. Taulukossa 2 on havainnollistettu löydettyjä teemoja sekä niiden esiintyvyyttä aineistoissa.

Taulukko 2. ihmisten ketterän johtamisen yhteiset teemat ja niiden esiintyvyys aineistoissa

Viite	Kollaboraatio ja valtuutus	Johtajan rooli fasilitaattorina ja valmentajana	Kehitystiimin itseorganisointi	Oppimisen tukeminen ja palaute	Ihmiskeikeinen toiminta
Appelo (2010)	x	x	x	x	x
Crowder ja Friess (2014)	x	x	x	x	x
Dubinsky ja Hazzan (2008)	x	x	x	x	x
Dubinsky ja Hazzan (2010)	x	x	x	x	x
Gren ja Lindman (2020)	x	x	x		x
Holtzhausen ja Klerk (2018)	x	x	x	x	x
Karhatsu et al. (2010)	x	x	x	x	x
León de la Barra et al. (2015)	x	x		x	
Levine (2020)	x	x	x	x	x
Medinilla (2012)	x	x	x	x	x
Moe et al. (2009)	x	x	x	x	x
Nguyen (2016)	x	x	x	x	x
Przybilla et al. (2019)	x	x	x	x	x
Shen ja Xu (2015)	x	x	x	x	x
Spiegler et al. (2019)	x	x	x	x	x
Spiegler et al. (2020)	x	x	x	x	x
Srivastava ja Jain (2017)	x	x	x	x	x
Stober ja Hansmann (2009)	x	x	x	x	x
Whitaker (2010)	x	x	x	x	x
Yang et al. (2009)	x	x		x	x

3.2.1 Kollaboraatio ja valtuutus

Kollaboratiivinen johtaminen on johtamisen malli, jonka ideana on jakaa päätöksentekovastuuta ja luoda yhteistyötä organisaation eri tasojen, kuten johdon ja kehittäjien välillä (Beck et al. 2011). Archer ja Cameron (2013, luku 1) esittävät kollaboratiivisen johtamisen olevan pääosin seuraavia asioita:

1. *Ihmissuhteiden rakentamista.* Hyvät henkilökohtaiset suhteet perustuen yhteisiin arvoihin ja intresseihin toimivat kollaboratiivisen johtamisen perustana.
2. *Konfliktien hallitsemista.* Organisaation eri tahojen väliset arvot ja intressit eivät aina ole samat ja tämä aiheuttaa väistämättä konflikteja. Kollaboratiivisen johtajan on kyettävä tunnistamaan konfliktit jo varhaisessa vaiheessa ja ratkaistava ne tavalla, joka johtaa parempaan osapuolten väliseen yhteisymmärrykseen.
3. *Kontrollin jakamista.* Kollaboratiivisen johtajan tulee ymmärtää, ettei hän voi yksin tehdä kaikkea.

Kollaboratiivinen johtaminen ei teoriassa sisällä päätöksentekovallan jakamista kehittäjien kesken. Ketterään ohjelmistokehitykseen kuuluu kuitenkin vahvasti se, että kehitystiimin jäsenet valtuutetaan tekemään päätöksiä. Täten ketterissä kehitystiimeissä kontrollin jakaminen viedään käytännössä pidemmälle, kuin Archerin ja Cameronin (2013, luku 1) kollaboratiiviseen johtamiseen teoreettisesti kuuluu.

Toinen aineistoissa vahvasti esiin tullut johtamisen malli oli jaettu johtajuus. Jaettu johtajuus on johtamisen malli, jonka ideana on jakaa vastuu johtamisesta ja päätöksenteosta tiimin jäsenten kesken. Jaetussa johtajuudessa kontrollin jakaminen viedään pidemmälle, kuin kollaboratiivisessa johtajuudessa. Jaettu johtajuus toimii hyvin silloin, kun tiimi on kokenut ja sen jäsenillä on sekä itseluottamusta että ammattitaitoa tehdä päätöksiä itsenäisesti. (Moe et al. 2019, s. 3)

Spiegler et al. (2019, s. 46–48) esittää, että tiimin saadessa kokemusta eli sen kypsyessä johtajuus siirtyy tiimin johtajalta (kyseisessä tutkimuksessa scrummaster) tiimin jäsenille vaiheittain. Tiimin kypsyessä ketterä johtaja voi enemmän ja enemmän antaa tiimin jäsenille valtaa tehdä päätöksiä ja siirtyä itse vain tukemaan tiimin työskentelyä. Tämä tapahtuu johtajan näkökulmasta johtamalla tiimiä ikään kuin puutteellisesti, eli luomalla johtajuuden tyhjiö, jonka kehitystiimin ajan myötä täyttää. Mitä kokeneemmaksi kehitystiimi kypsyy, sitä enemmän ketterä johtaja voi johtajuuden tyhjiötä luomalla valtuuttaa tiimin jäseniä. Ketterän kehitystiimin autonomian aste riippuu siis hyvin paljon siitä, kuinka kokenut se on. (Spiegler et al. 2019, s. 37)

Aineistojen perusteella voidaan päätellä, että ketterissä ohjelmistoprojekteissa ihmisten johtaminen on sekä kollaboratiivista että jaettua johtamista. Mitä kokemattomampi kehitystiimi on, sitä enemmän sen johtaminen muistuttaa kollaboratiivista johtamista, kun taas tiimin kypsyessä johtamisen malli alkaa muistuttamaan enemmän jaettuja johtajuutta. Teeman nimellä ”Kollaboraatio ja valtuutus” on pyritty kuvaamaan ketterän johtamisen kollaboratiivisen sekä jaetun johtajuuden osa-alueita ja yhdistämään ne yhdeksi teemaksi.

3.2.2 Johtajan rooli fasilitaattorina ja valmentajana

Moe et al. (2009, s. 2) esittää, että ketterässä ohjelmistokehityksessä perinteinen projektipäällikkö on korvattu ketterällä johtajalla, joka toimii kehitystiimin fasilitaattorina ja valmentajana. Fasilitointi on ketterän johtajan näkökulmasta kehitystiimin jäsenten välisen yhteistyön suunnittelua ja toteuttamista joka pitää Levinen (2020, s. 10) sisällään kaksi tehtävää:

1. Johtaa tiimiä toimittamaan arvoa tuottavia ratkaisuja
2. Mahdollistaa tiimin toiminta ja onnistuminen osana laajempaa organisaatiota.

Ketterän johtajan tulee olla tukemassa kehitystiimin toimintaa ja ohjata sitä oikeaan suuntaan yhteisen päämäärän kannalta (Levine 2020, s. 10). Tämä voidaan toteuttaa muun muassa päivittäisillä tai viikoittaisilla palaverilla, joissa kehitystiimin jäsenet kertovat, mitä he ovat tehneet, mitä he aikoivat tehdä ja tuovat esiin ongelmat, mikäli niitä on. Tällöin ketterä johtaja voi tarvittaessa tehdä toimenpiteitä tukeakseen tiimin toimintaa, ohjata yksittäisiä kehittäjiä tekemään oikeita asioita tai auttaa heitä pääsemään yli ongelmistaan.

Ketterän johtajan rooli valmentajana muistuttaa hyvin paljon urheilujoukkueen valmentajan roolia. Valmentajana ketterä johtaja pyrkii parantamaan tiimin suorituskykyä, hoitaa kommunikointiin liittyviä ongelmia sekä rohkaisee tiimin jäseniä tekemään asioita itsenäisesti. Voidaan sanoa, että hyvä valmentaja jättää jälkeensä hyvin toimivan itseorganisoituvan kehitystiimin. (Dubinsky & Hazzan 2008, s. 264)

3.2.3 Kehitystiimin itseorganisointi

Siinä missä ketterä kehitystiimi ei ideaalitalanteessa tarvitse nimettyä johtajaa tekemään päätöksiä, se ei myöskään tarvitse johtajaa jakamaan työtä. Tällaista tiimiä, joka kykenee itse löytämään työtä ja jakamaan sitä kutsutaan itseorganisoituvaksi. Työn löytämisen ja sen jakamisen lisäksi itseorganisoituva kehitystiimi kykenee reagoimaan nopeasti muutoksiin ja ongelmiin sekä on usein tehokkaampi ja motivoituneempi kuin perinteisesti johdettu tiimi. (Karhatsu et al. 2010, s. 1)

Karhatsun et al. (2010, s. 3) mukaan tällaisen tiimin toiminnan kannalta on oleellista, että tiimeillä on korkea omistajuuden ja vastuun tunne projektia kohtaan. Itseorganisoituvan tiimin onnistumisen kannalta korostuu erityisesti tiimityön merkitys sekä tiivis jäsenten keskeinen kommunikointi. Myös jäsenten keskeinen kunnioitus ja luottamus sekä omaan että toisten kompetenssiin ovat oleellisia tekijöitä tällaisissa tiimeissä, joissa ulkoista auktoriteettia on hyvin vähän tai ei ollenkaan. (Karhatsu et al. 2010, s. 3)

Ennen kuin tiimi kykenee olemaan täysin itseorganisoituva, se usein tarvitsee alkuvaiheessa koulutusta ja mentorointia joko tiimin sisäisesti tai ulkoiselta taholta. Ihmisten johtamisen näkökulmasta ketterän johtajan rooli fasilitaattorina ja valmentajana korostuu varsinkin tiimin ollessa kokematon. Tällöin tiimin toiminta muistuttaa alkuvaiheessa perinteisesti johdettua tiimiä, joka vähitellen jäsenien tietotaidon karttuessa muuttuu enemmän ja enemmän itseorganisoituvammaksi. (Spiegler et al. 2019, s. 37, 43)

Itseorganisoituva tiimi saattaa kuulostaa kaoottiselta, mutta sitä se ei todellisuudessa ole. Ketterän ohjelmistokehityksen julistuksen (Beck et al. 2011) mukaan ”parhaat arkkitehtuurit, vaatimukset ja suunnitelmat syntyvät itseorganisoituvissa tiimeissä”, edellyttäen että itseorganisoituvuus on toteutettu hyvin.

3.2.4 Oppimisen tukeminen ja palaute

Crowder ja Friess (2015, s. 14) mukaan ketterä kehitystiimi koostuu yksilöistä, joilla on toistaan eroavia vahvuuksia, heikkouksia ja heidän työkokemuksensa spektri on usein toisistaan eroava. Tällainen tiimi tarvitsee kehitysympäristön, jossa ei pelkästään ole edellytykset menestykseen vaan sen lisäksi myös mahdollisuudet kehittää omaa tietotaitoaan sekä oman työpanoksensa että muiden jäsenten kautta. Tästä syystä ketterän johtajan tulisi siis luoda edellytyksen oppimiselle ja palautteelle, sekä fasilitoida sitä parhaansa mukaan. Ketterän johtajan tulisi myös varmistaa, että jokaisella tiimin jäsenellä on mahdollisuus olla mentoroinnin antavana ja vastaanottavana osapuolena. Tällainen toiminta tulisi olla sisäänrakennettuna tiimin toimintaan ja aikatauluun ja se on usein järjestetty käytännössä esimerkiksi retrospektiiveinä sprinttien välillä. (Crowder ja Friess 2015 s. 14, 23)

Toimialasta riippumatta ammattitaidon kehittyminen tapahtuu oppimisen kautta ja oppimista taas tukee palaute. Ketterän kehitystiimin itsenäisyys ja tätä kautta kehitystiimin ketteryyden aste riippuu taas hyvin paljon kehitystiimin jäsenien ammattitaidosta. Voidaan siis tehdä johtopäätös, että kehitystiimin ketteryyden asteen kasvattaminen vaatii oppimisen tukemista ja rakentavaa palautetta. Tästä syystä ketterän johtajan tulee parhaansa mukaan vaalia oppimista ja rakentavaa palautetta, sillä varsinkaan kokemattomien kehitystiimien ei tule olettaa pystyvän toimimaan ainakaan täysin itseorganisoituvasti. (Karhatsu et al. s. 2)

3.2.5 Ihmiskeskeinen toiminta

Ketterän ohjelmistokehityksen julistuksen (Beck et al. 2011) mukaan yksi ketterän ohjelmistokehityksen perusarvoista on laittaa ihmiset ja yhteistyö ennen prosesseja ja menetelmiä. Tällaista toimintamallia, jossa korostetaan yhteistyötä, kasvotusten kommunikointia, palautetta, luottamusta sekä vastuun ja auktoriteetin jakamista kutsutaan ihmiskeskeiseksi. Ketterän johtajan tulee vaalia edellä mainittuja asioita sekä motivoinnin keinoin saada kehitystiimin jäsenet tuntemaan itsensä arvokkaiksi. Ihmiskeskeiseen toimintaan kuuluu myös kehitystiimin jäsenten ryhmäytyminen ja yhteenkuuluvuuden tunne, jota ketterä johtaja voi vahvistaa esimerkiksi järjestämällä ryhmäaktiviteetteja. (Srivastava ja Jain 2017, s. 307)

Ketterässä ohjelmistokehityksessä ihmiskeskeinen toiminta ei pelkästään rajoitu kehitystiimin keskeiseksi, vaan kanssakäyminen asiakkaankin kanssa pyritään tekemään ihmiskeskeisesti. Kasvotusten pidetyt tapaamiset asiakkaan kanssa pitävät sisällään paljon ei-verbaalista kanssakäymistä, jonka pois jääminen heikentää kehitystiimin ja asiakkaan välistä yhteisymmärrystä. (Srivastava ja Jain 2017, s. 15) Koska ketterän ohjelmistokehityksen periaatteisiin ei kuulu kattava dokumentaatio tai sopimusneuvottelut (Beck et al. 2011), yhteisymmärryksen puute asiakkaan kanssa saattaa johtaa siihen, että ohjelmistokehityksen tuotos ei vastaa asiakkaan tarpeita tai toiveita (Srivastava ja Jain 2017, s. 15).

3.3 Johtopäätökset

Tutkimusaineiston perusteella voidaan sanoa, että ketterän johtamisen tarkoituksena on ajan myötä luovuttaa valta johtajalta kehitystiimille. Tämä tosin edellyttää sitä, että kehitystiimi oppii tekemään päätöksiä itsenäisesti, ja tähän kokematon kehitystiimi ei välttämättä kykene. Ajan myötä kehitystiimi oppii toimimaan itsenäisesti, ja johtaja alkaa luovuttamaan kontrollia kehitystiimille. Kehitystiimin oppiminen siis korreloi sen kykyyn itseorganisoitua, ja kyky itseorganisoitua kertoo taas kehitystiimin ketteryyden asteen. Tästä voidaan päätellä, että johtajan kontrolli on kääntäen verrannollinen kehitystiimin kokemukseen, kykyyn itseorganisoitua sekä sen ketteryyden asteeseen. Ajan myötä kehitystiimin oppiessa itseorganisoituvammaksi ja ketterämmäksi, johtajan kontrollin tulisi vähentyä samanaikaisesti.

Kehitystiimin johtaja voi toiminnallaan edesauttaa sen itseorganisoituvuutta ja vaikuttaa positiivisesti sen ketteryyden asteeseen. Luvussa 3.2 on esitetyistä teemoista kollaboraatio ja valtuutus, johtajan rooli fasilitaattorina ja valmentajana, oppimisen tukeminen ja palaute sekä ihmiskeskeinen toiminta ovat työkaluja kehitystiimin itseorganisoituvuuden parantamiseen ja tätä kautta ketteryyden asteen kasvattamiseen. Myös luottamus tiimin jäsenien kykyyn tehdä päätöksiä on oleellinen osa kehitystiimin ketteryyden edistämistä.

4 Pohdinta

Ihmisten johtaminen on ohjelmistoalalla siirtynyt perinteisestä hierarkkisesta mallista autonomisemmaksi, mikä on tuonut tullessaan itsenäisesti päätöksiä tekevät itseorganisoidut kehitystiimit. Tällaisen kehitystiimin tehokas toiminta ei kuitenkaan ole itsestään selvää, vaan ketterän johtajan on toiminnallaan fasilitoitava ja valmennettava tiimi itsenäiseksi sekä luoda sellainen kehitysympäristö, jossa se pystyy esteettömästi toimimaan. Luottamus, arvostus, kommunikointi, oppinen ja palaute ovat suuri osa ketterää ihmisten johtamista.

Tutkimuksen toiseen tutkimuskysymykseen saatiin vastattua paremmin kuin ensimmäiseen. Aineistojen pohjalta pystyttiin keräämään ketterään johtamiseen liittyviä teemoja, joiden pohjalta voisi kehittää viitekehyksen ihmisten johtamiselle ketterissä ohjelmistoprojekteissa. Tutkimustulokset eivät suoranaisesti anna ohjenuoraa ketterälle ihmisten johtamiselle. Tutkimuksessa pyrittiin etsimään aineistojen välisiä yhtenäisyyksiä, joten aiempiin tutkimuksiin ja taustateoriaan tämä tutkimus suhtautuu myötäilevästi. Tutkimusmenetelmänä systemaattinen kirjallisuuskatsaus on rajoittunut tutkimaan ainoastaan teoreettisesti ja empiirisyys näkyy ainoastaan katsaukseen otettujen aineistojen mahdollisina empiirisinä osuuksina.

Katsauksessa käytettyjen lähteiden määrä oli suuri, joten sen pohjalta voidaan sanoa tulosten olevan yleistettävissä. Ketterässä ohjelmistoprojektissa toimimista käydään mielestäni liian vähäisesti läpi tietotekniikan opinoissa Tampereen yliopistossa, joten tutkimustuloksia voitaisiin hyödyntää ihmisten johtamisen viitekehyksenä toimiessa ketterissä kehitysympäristöissä. Tutkimustulokset voivat auttaa tietotekniikan opiskelijaa ymmärtämään ihmisten johtamista ketterissä ohjelmistoprojekteissa, mikä taas auttaa sekä toimimaan ketterissä kehitystiimissä että johtamaan niitä.

Tutkimuksen perusteella mielenkiintoisia jatkotutkimusaiheita avautui esimerkiksi ketterästä ohjelmistoympäristöstä ohjelmistokehittäjän tai asiakkaan näkökulmasta. Nämä perspektiivit voisivat täydentää ketterissä ohjelmistoprojekteissa toimimisen viitekehystä, jota voisi hyödyntää esimerkiksi tulevaisuuden tietotekniikan ammattilaisten kouluttamisessa. Tällainen kokonaisvaltainen tieto olisi auttanut allekirjoittanutta ymmärtämään töissä toimintaympäristöään.

5 Yhteenveto

Tässä tutkimuksessa tarkasteltiin ihmisten johtamista ketterissä ohjelmistoprojekteissa systemaattisin kirjallisuuskatsauksen keinoin. Tutkimuskysymykset olivat:

1. Kuinka ihmisiä johdetaan ketterissä ohjelmistoprojekteissa?
2. Mitä yhteisiä teemoja aiemmasta kirjallisuudesta löytyy ihmisten johtamisesta ketterissä ohjelmistoprojekteissa?

Ihmisten johtamista ketterässä ohjelmistoprojektissa voidaan tulosten valossa luonnehtia valvotuksia autonomiaksi. Ideaalitapauksessa kehitystiimi on itseorganisoituva, eli tekee itsenäisesti päätöksiä ja johtaa itseään kohti yhteistä päämäärää. Voidaan siis sanoa, että hyvin onnistuneen ketterän johtajan ei tarvitse johtaa. Tällainen toimintamalli vaatii kuitenkin luottamusta ketterän johtajan ja kehitystiiminvälillä, luottamusta kehitystiimien jäsenten välillä sekä yksittäisiltä jäseniltä itseluottamusta tehdä päätöksiä itsenäisesti ja ottaa vastuuta. Kehitystiimin kykyyn olla itseorganisoituva vaikuttaa tiimin kypsyys: mitä kokeneempi kehitystiimi sitä korkeampi ketteryyden asti sillä on. Kokematon tiimi vaatii paljon enemmän huomiota ja ulkoista johtajuutta ja ketterän johtajan tulisi tiimin kypyydessä valtuuttaa yhä enemmän kehitystiimiä toiminaan itsenäisesti.

Aineistoista löydetty yhteiset teemat olivat:

- Kollaboraatio ja valtuutus – ketterissä ohjelmistoprojekteissa johtamisen malli muistuttaa useimmiten kollaboratiivista johtamista, johon on lisätty jaetusta johtajuudesta valtuuttamisen elementti
- Johtajan rooli fasilitaattorina ja valmentajana – ketterä johtaja pyrkii ohjaamaan ja opettamaan kehitystiimin jäseniä tekemään oikeita asioita itsenäisesti
- Kehitystiimin itseorganisointi – kehitystiimi kykenee toiminaan itsenäisesti ilman johtajaa
- Oppimisen tukeminen ja palaute – oppiminen auttaa kehitystiimiä kypsymään ja kehitystiimin kypsyys on suoraan verrannollinen tiimin ketteryyden asteeseen. Palaute taas tukee oppimista
- Ihmiskeskeinen toiminta – kanssakäymiset ja ihmiset ennen prosesseja ja työkaluja

Tutkimuksen tuloksista saatu ketterän johtamisen viitekehys on yleiskäyttöinen ja auttaa lukijaa sekä toimimaan ketterässä ohjelmistoprojektissa että johtamaan sitä.

Lähteet

- Appelo, J. (2010). *Management 3.0: Leading agile developers, developing agile leaders* (1st ed.) Addison-Wesley Professional.
- Archer, D., Cameron A. (2013). *Collaborative leadership building relationships, handling conflict, sharing control* (2nd ed. ed.). New York: Routledge.
- Beck, K., Beedle, M., Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R., Mel-lor, S., Schwaber, K., Sutherland, J., Thomas, D. (2001) *The Agile Manifesto*. Agile Alliance. <http://agilemanifesto.org/>
- Crowder, J. A., & Friess, S. (2015). The psychology of agile team leadership. In J. A. Crowder, & S. Friess (Eds.), *Agile project management: Managing for success* (pp. 9–25). Cham: Springer International Publishing. doi:10.1007/978-3-319-09018-4_2 Retrieved from https://doi.org/10.1007/978-3-319-09018-4_2
- Dubinsky, Y., & Hazzan, O. (2010). Ad-hoc leadership in agile software development environments ACM. doi:10.1145/1833310.1833316
- Fink, A. (2013). *Conducting research literature reviews: From the internet to paper* Sage Publications.
- Gren, L., & Lindman, M. (2020). What an agile leader does: The group dynamics perspective. In: Stray V., Hoda R., Paasivaara M., Kruchten P. (eds) *Agile Processes in Software Engineering and Extreme*, 178–194.
- Karhatsu, H., Ikonen, M., Kettunen, P., Fagerholm, F., & Abrahamsson, P. (2010). Building blocks for self-organizing software development teams a framework model and empirical pilot study, 2010 2nd International Conference on Software Technology and Engineering, 1 V1-304. doi:10.1109/ICSTE.2010.5608848
- Hazzan, O., & Dubinsky, Y. (2008). Leadership. In O. Hazzan, & Y. Dubinsky (Eds.), *Agile software engineering* (pp. 1–22). London: Springer London. doi:10.1007/978-1-84800-198-5_13 Retrieved from https://doi.org/10.1007/978-1-84800-198-5_13
- León de la Barra, Claudio, Galdames, S., Crawford, B., Soto, R., & Crawford, K. (2015). Leadership in agile software development methods. In: Stephanidis C. (eds) *HCI International 2015*, 154–158.
- Levine, M. K. (2020). *People over process: Leadership for agility* (1st ed.). Milton: Routledge. doi:10.4324/9780429324758

- Medinilla, Á. (2012). The agile manager's role. In Á Medinilla (Ed.), *Agile management: Leadership in an agile environment* (pp. 53–66). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-28909-5_3 Retrieved from https://doi.org/10.1007/978-3-642-28909-5_3
- Moe, N. B., Dingsyr, T., & Kvangardsnes, O. (2009). Understanding shared leadership in agile development: A case study IEEE. doi:10.1109/HICSS.2009.480
- Nguyen, D. (2013). Success factors for building and managing high performance agile software development teams. *International Journal of Sciences: Basic and Applied Research*, 72-93. Retrieved from <https://core.ac.uk/reader/229655470>
- Nico, H., & de Klerk Jeremias, J. (2018). Servant leadership and the scrum team's effectiveness. *Leadership & Organization Development Journal*, 39(7), 873–882. doi:10.1108/LODJ-05-2018-0193
- Pallavi, S., & Shilpi, J. (2017). A leadership framework for distributed self-organized scrum teams. *Team Performance Management: An International Journal*, 23(5), 293-314. doi:10.1108/TPM-06-2016-0033
- Przybilla, L., Wiesche, M., & Krcmar, H. (2019). Emergent leadership in agile teams--an initial exploration. Retrieved from <https://doi.org/10.1145/3322385.3322423>
- Sánchez, J. (2018). Challenges of traditional and agile software processes. Computer Science Department, University of Chile
- Spiegler, S. V., Graziotin, D., Heinecke, C., & Wagner, S. (2020). A quantitative exploration of the 9-factor theory: Distribution of leadership roles between scrum master and agile team. In: Stray V., Hoda R., Paasivaara M., Kruchten P. (eds) *Agile Processes in Software Engineering and Extreme*, 162–177.
- Spiegler, S. V., Heinecke, C., & Wagner, S. (2019). Leadership gap in agile teams: How teams and scrum masters mature. In: Kruchten P., Fraser S., Coallier F. (eds) *Agile Processes in Software Engineering and Extreme*, 37–52.
- Stober, T., & Hansmann, U. (2009). *Agile software development: Best practices for large software development projects* (1st ed.). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-540-70832-2
- Weber, R. P. (1990). *Basic content analysis* (2nd ed.). Newbury Park, Calif: Sage Publications.
- Whitaker, K. (2010). *Principles of software development leadership applying project management principles to agile software development*. Boston: Course Technology.

- Xu, P., & Shen, Y. (2015). Leading agile teams: An exploratory study of leadership styles in agile software development. Retrieved from <https://aisel.aisnet.org/amcis2015/ITProjMan/GeneralPresentations/4/>
- Yang, H., Huff, S., & Strode, D. (2009). Leadership in software development: Comparing perceptions of agile and traditional project managers. Retrieved from <https://www.semanticscholar.org/paper/Leadership-in-Software-Development%3A-Comparing-of-Yang-Huff/d31a54056b85e8d968494dfa1fadc4bf7d738100?p2df>

Liitteet

Liite 1. Systemaattiseen kirjallisuuskatsaukseen valikoituneet tutkimusartikkelit

Viite	Osallistujat	Tavoitteet	Tutkimusmenetelmät	Konteksti	Tulokset
de la Barra et al. (2015)	Ei osallistujia	Analysoida, kuinka ihmisten johtaminen näkyy ohjelmistokehityksessä ja etenkin Extreme Programmingia hyödyntävissä tiimeissä	Kirjallisuuskatsaus	Ei kuvailtua kontekstia	Ketteriä menetelmiä, kuten Extreme Programmingia hyödyntäviin kehitystiimeihin sopii parhaiten jaettu lähestymistapa johtajuuteen
Dubinsky ja Hazzan (2010)	Kehitystiimien jäsenet ja organisaatioiden avainhenkilöt	Selvitetään, kuinka johtajuus ilmenee muutosprosessissa siirtyessä ketteriin menetelmiin. Erityisesti keskitytään muutosjohtajaan (change leader tai CL)	Havainnointi (tärkeät palaverit, sprint planningit, sprint reviewit, retrospektiivi), kyselyt tiimien jäsenille ja johdon edustajille, organisaation avainhenkilöiden haastattelu sekä kirjeenvaihto muutosjohtajien kanssa.	Osasto ohjelmistoalan yrityksessä, joka on juuri ottanut Scrumin käyttöön	Muutosjohtajan rooli on ehdottoman tärkeä siirtyessä ketteriin menetelmiin ja hänen onnistuminen työssään riippuu hyvin paljon ylempien tason johdon antamasta tuesta. Artikkelin kuvaus muutosjohtajan johtamistyyliä ad hoc -johtamiseksi: muutosjohtaja johtaa vain silloin, kun sille on tarvetta. Usein tällaiset tilanteet syntyvät, kun ilmenee jonkin spesifi ongelma, johon ketterät menetelmät ovat sopiva lähestymistapa.

Viite	Osallistujat	Tavoitteet	Tutkimusmenetelmät	Konteksti	Tulokset
Gren ja Lindman (2020)	15 osallistujaa eri organisaatioiden eri tasoilta, jotka hyödynsivät ketterää johtajuutta	Tutkia, mitä haasteita ketterän kehitystiimin johtamisessa on ryhmädynamiikan näkökulmasta	Kvalitatiiviset haastattelut ja haastatteludatan temaattinen analyysi	Kaikki osallistajat olivat tutkimuksen tekijöiden henkilökohtaisia kontakteja	Tiimin kypsyys, rakenne sekä kulttuuri ja ajattelutapa olivat ryhmädynamiikan haasteisiin liittyviä kategorioita, joita osallistajat kohtaava työelämässään
Holtzhausen ja Klerk (2018)	73 scrum-tiimin jäsentä ja 22 scrummasteria	Tutkia, missä määrin scrummasterit käyttävät palvelevaa johtamista	Kvantitatiivinen verkkokysely	Osallistajat olivat 17:sta eri eteläafrikkalaisesta organisaatiosta	Scrummasterit käyttivät laajasti palvelevaa johtajuutta, mutta ne scrummasterit, jotka formaalisti nimitettiin tiimin johtajiksi miellettiin tiimien jäsenien mukaan paremmiksi palvelevassa johtajuudessa. Scrummasterin palvelevan johtajuuden ja tiimitehokkuuden välillä oli kohtalainen korrelaatio

Viite	Osallistujat	Tavoitteet	Tutkimusmenetelmät	Konteksti	Tulokset
Karhatsu et al. (2010)	Kehitystiimien jäsenet ja asiakkaiden edustajat	Raportoida malli ketterän ohjelmistokehitystiimin rakentamiseen	Kirjallisuuskatsaus, projektien tarkkailu ja haastattelut	Projektit ja tutkimus toteutettiin Helsingin yliopiston Software Factoryssa	Tutkimus antoi empiiristä todistusaineistoa siitä, että ketterän kehitystiimin tärkeimmät elementit ovat autonomia, tehokas kommunikointi sekä yhteistyö.
Moe et al. (2009)	Kaksi kehittäjää, scrummaster, projektipäällikkö ja tuotteen omistaja	Ymmärtää tiimin johtamista ketterässä ohjelmistokehityksessä ja sen haasteita	Havainnointi (daily-palaverit, sprint planning, sprint reviewit), haastattelut (kehittäjät ja scrummaster) sekä dokumenttien läpikäynti	Osasto ohjelmistotalan yrityksessä, joka on juuri ottanut scrumin käyttöön	Mekanismeja kaksikehäiseen oppimiseen ei asetettu ja koska tiimi yritti tehdä vaikutusta tuotteen omistajaan, oppimisesta tuli vaikeaa. Osastolla oli myös haasteita löytää sopiva autonomian suhde ulkoisesti, sisäisesti sekä yksilöllisesti. Kehittäjät eivät halunneet ottaa vastuuta projektissa, sillä he pelkäsivät, että sen tehtyä vastuu siirtyisi pysyvästi heille.

Viite	Osallistujat	Tavoitteet	Tutkimusmenetelmät	Konteksti	Tulokset
Nguyen (2016)	Ei osallistujia	Määrittää menestystekijät tehokkaan ketterän kehitystiimin johtamiseen	Kirjallisuuskatsaus	Ei kuvailtua kontekstia	Kirjallisuuskatsauksessa käsitellään muun muassa ketterän johtamisen, tuotejohtamisen, kehitystiimin, tekniikoiden, teknologioiden ja kehitystyökalujen merkitystä ketterän ohjelmistokehityksen tehokkuuteen ja johtamiseen.
Przybilla et al. (2019)	Kehitystiimin jäsenet	Selvittää, missä muodossa emergenttiä johtajuutta ilmenee ketterissä tiimeissä ja mitkä ovat sen vaikutukset	Kyselytutkimus	Ei kuvailtua kontekstia	Ehdotettu tutkimus, tuloksia ei ilmoitettu

Viite	Osallistujat	Tavoitteet	Tutkimusmenetelmät	Konteksti	Tulokset
Shen ja Xu (2015)	Kehitysprojektin johtaja ja jäsenet	Tutkia eri johtamistyylien rooleja ketterissä menetelmissä ja ymmärtää, miten eri tyyppiset johtamistavat parantavat ketterän kehitysprojektin suorituskykyä	Case-tutkimus kvalitatiivisilla menetelmillä (haastattelut, tarkkailu ja dokumentaatioiden katselmointi)	Ketterä ohjelmistoprojekti	Ei ilmoitettu
Spiegler et al. (2019)	22 scrummasteria, 8 tuotteen omistajaa ja 23 tiimin jäsentä members	Ymmärtää, kuinka scrummasterin rooli muuttuu kehitystiimin kokemuksen karttuessa	Kvalitatiiviset haastattelut ja tarkkailu	Tiimejä oli yhteensä 29 ja niiden koko oli 5-12 jäsentä ja ne olivat usein monikansallisia. Tiimien ikä vaihteli kolmesta kuukaudesta kolmeen vuoteen	Kokemattomassa tiimissä scrummasterin tehtävä on johtaa. Kun tiimin kokemus karttuu, johtamisen vastuu siirtyy yhdeltä ihmiseltä koko tiimin vastuulle

Viite	Osallistujat	Tavoitteet	Tutkimusmenetelmät	Konteksti	Tulokset
Spiegler et al. (2020)	67 työntekijää monialayritys Robert Bosch GmbH:sta	Selvittää, mitä johtajuuden rooleja scrummasterilla ja ketterällä kehitystiimillä itsellään on. Tutkimus pyrkii myös selvittämään, jakautuvatko johtajuuden roolit scrummasterin ja tiimin kesken ja jos jakautuu, niin jakautuuko se enemmän kokoneemmalle kuin kokemattomalle kehitystiimille	Kvantitatiivinen verkkokysely	Tutkimuksessa tutkittiin teoriaa, jossa scrummasterilla olisi 9 johtajuuden roolia (the 9-Factor Theory of Scrum Master roles)	Scrummasterin 9 johtajuuden roolia jakautuvat scrummasterin ja kehitystiimin kesken eri tavoin eri tiimeissä. Scrummasterin roolin oli useimmiten jaettu tiimin jäsenille tiimeissä, jotka olivat työskennelleet 3-5 kuukautta ketterästi. Johtajuuden roolit kuitenkin usein pysyvät scrummasterilla
Srivastava ja Jain (2017)	18 scrummasteria	Selvittää, minkälaiset johtamismekanismit itse- organisoituvassa scrum- tiimissä johtavat projektin onnistumiseen ja asiakkaan tyytyväisyyteen	Kvalitatiiviset menetelmät avoimilla kyselyillä ja puolistrukturoiduilla haastattelulla	Osallistajat olivat saman organisaation alla olevista 75:stä eri projektista. Heistä 13 oli miehiä ja 5 naisia	Tutkimus identifioi onnistumisen kannalta oleelliset johtamismekanismit, joita olivat muun muassa ihmiskehinen johtaminen, kommunikointi, avoimuus, jaettu johtajuus jne. Nämä mekanismit liittyvät erityisesti scrummasterin rooliin

Viite	Osallistajat	Tavoitteet	Tutkimusmenetelmät	Konteksti	Tulokset
Yang et al. (2009)	8 projektipäällikköä	Selvittää, millainen johtamis- käyttäytymisen on projektipäällikköiden mielestä sopivaa perinteisissä ja ketterissä ohjelmistoprojekteissa ja miksi	Kvantitatiiviset kyselyt ja kvalitatiiviset puolistrukturoidut haastattelut	Projektipäällikköistä 4 johti ketterää ohjelmistoprojektia ja 4 perinteistä. He olivat 7:stä eri organisaatiosta, heillä oli vähintään 6 vuotta kokemusta projektipäällikön työstä, 4 heistä oli naisia ja 4 miehiä, ikähaarukka oli 30-50 vuotta ja tiimien koko oli 5-50 henkilöä.	Ketterissä projekteissa työskennelleet projektipäälliköt arvostivat kyselyssä transformationaalisen johtamisen osa-alueita enemmän, kuin perinteisiä menetelmiä käyttävät projektipäälliköt. Transaktionaalisen johtamisen menetelmät saivat sekä ketteriä että perinteisiä menetelmiä käyttäviltä projektipäälliköitä tyydyttävät arvosanat. Laissez-faire -johtaminen sai molemmilta huonot arvostelut. Tutkimuksen otosryhmä on kuitenkin niin pieni, ettei siitä voi vetää tilastollisia johtopäätöksiä.