

Kari Hara

**KETTERÄ OHJELMISTOKEHITYS ASIAKKAAN
NÄKÖKULMASTA**



JYVÄSKYLÄN YLIOPISTO
TIETOJENKÄSITTELYTIETEIDEN LAITOS
2013

TIIVISTELMÄ

Hara, Kari

Ketterä ohjelmistokehitys asiakkaan näkökulmasta

Jyväskylä: Jyväskylän yliopisto, 2013, 104 s.

Tietojenkäsittelytiede, pro gradu -tutkielma

Ohjaaja: Leppänen, Mauri

Ketterä ohjelmistokehitys muuttaa merkittävästi asiakkaan roolia verrattuna perinteiseen ohjelmistokehitykseen. Tämä muutos merkitsee asiakkaalle paitsi uusia mahdollisuuksia, myös uusia haasteita, joista menestyksekkäs selviytymisen puolestaan vaatii uudenlaisia taitoja. Nämä haasteet sekä keinot niistä selviytymiseksi ovat kuitenkin jääneet merkittävyyteensä nähden vähäiselle huomiolle tutkimuksessa. Tässä tutkielmassa käsitelläänkin nimenomaan näitä ketterän ohjelmistokehityksen haasteita ja onnistumisen edellytyksiä asiakkaan näkökulmasta.

Asiakkaan näkökulman ketterään ohjelmistokehitykseen määritellään sisältävän kolme merkittävää osaa: sopimuksen tekeminen, kehitettävän ohjelmiston arvon määrittäminen ja osallistuminen kehittämisen aikana. Sopimuksen tekemistä käsittelevään osaan sisältyy yleisesti käytettyjen sopimusmallien esittely ja arviointi asiakkaan näkökulmasta. Arvon määrittämistä käsittelevään osaan puolestaan sisältyy yleisesti käytettyjen arvon määrittämisen menetelmien esittely ja arviointi asiakkaan näkökulmasta. Kehittämisen aikaista osallistumista käsittelevään osaan sisältyy yleisesti käytettyjen asiakkaan osallistumisen aktiviteettien esittely ja arviointi asiakkaan näkökulmasta. Lopuksi asiakkaan näkökulman kolme osaa liitetään yhteen tarkastelemalla niiden välisiä vuorovaikutussuhteita ja muodostetaan siten ketterän ohjelmistokehityksen kokonaiskuva asiakkaan näkökulmasta.

Asiakkaan näkökulman kolmen osan käsittelyn tuloksena löydetään selviä eroavuuksia sopimusmallien, arvon määrittämisen menetelmien ja osallistumisen aktiviteettien välillä niin niiden käyttökelpoisuudessa, sopivuudessa ketterään kehittämiseen kuin myös edullisuudessa asiakkaalle. Näkökulman eri osien vuorovaikutussuhteiden analysointi puolestaan antaa kokonaiskuvan asiakkaan menestyksekkään toiminnan edellytyksistä, jota vahvistaa myös esitetty prosessimalli asiakkaan toiminnasta ketterässä ohjelmistokehityksessä.

Tutkielma on käsitteellis-teoreettinen ja perustuu aihealueelta julkaistuihin teoreettisiin ja empiirisiin tutkimuksiin.

Asiasanat: ohjelmistokehitys, ketterät menetelmät, asiakkaat, ohjelmistoliiketoiminta

ABSTRACT

Hara, Kari

Agile Software Development from the Customer's Point of View

Jyväskylä: University of Jyväskylä, 2013, 104 p.

Computer Science, Master's Thesis

Supervisor: Leppänen, Mauri

Agile software development significantly changes the role of the customer compared to traditional software development. To the customer this change means not only new opportunities but also new challenges and surviving these challenges successfully requires a new set of skills from the customer. These challenges and different ways to manage them are, however, largely neglected in the current research compared to their significance. This thesis deals specifically with these agile software development challenges and keys to success from the customer's point of view.

Customer's point of view to agile software development is defined to include three parts: contract negotiation, determination of the value of the software under development, and participation during the development. The first part concerning the contract negotiation includes an introduction of commonly used contract models and their evaluation from the customer's point of view. The second part concerning value determination in turn includes an introduction of commonly used methods for determining the value of the software under development and their evaluation from the customer's point of view. The third part concerning the participation during the development includes an introduction of commonly used customer participation activities and their evaluation from the customer's point of view. Finally these three parts of the customer's point of view are combined by considering their mutual interactions and thereby forming an overall picture of the agile software development from the customer's point of view.

Based on the results of the aforementioned evaluations, several clear distinctions are found in the contract models, methods for determining the value and participation activities in terms of their usefulness, suitability for agile development and favorability for the customer. The analysis of the interactions between the parts of the customer's point of view offers in turn an overall picture on the conditions of a successful operation by the customer. This overall picture is also strengthened by a proposed process model of the customer's operation in agile software development.

The thesis is performed as a theoretical study, and it is based on theoretical and empirical studies published on the subject area.

Keywords: software development, agile methods, customers, software business

KUVIOT

KUVIO 1 Ketterän kehittämisen menetelmien evoluutio.....	13
KUVIO 2 Scrumin prosessi.....	16
KUVIO 3 XP:n käytänteiden suhteet	18
KUVIO 4 Lean-tuotanto.....	20
KUVIO 5 Asiakkaan näkökulma ketterään kehittämiseen.....	24
KUVIO 6 Kiinteähintainen sopimusmalli ja sen johdannaiset	28
KUVIO 7 Aika-ja-materiaalit –sopimusmalli ja sen johdannaiset.....	29
KUVIO 8 Asiakkaan näkökulma sopimusmalleihin	38
KUVIO 9 Riskin jakautuminen	38
KUVIO 10 Sopimusmallien arviointi Venn-diagrammina	42
KUVIO 11 Ansaitun arvon määrittäminen.....	51
KUVIO 12 Nykyisen nettoarvon kaava.....	53
KUVIO 13 Työnosituskaavio	55
KUVIO 14 Sisältöosan liiketoiminta-arvon kaava	55
KUVIO 15 Ansaitun liiketoiminta-arvon kaava.....	55
KUVIO 16 Asiakkaan näkökulma arvon määrittämisen menetelmiin.....	58
KUVIO 17 Järjestelmäkehityksen osallistumisen teoreettinen viitekehys	67
KUVIO 18 Asiakkaan työmäärän jakautuminen	78
KUVIO 19 Scrum-asiakkaiden luokittelujärjestelmä.....	79
KUVIO 20 Asiakkaan näkökulma osallistumisen aktiviteetteihin.....	80
KUVIO 21 Asiakkaan näkökulman osien vuorovaikutussuhteet	91
KUVIO 22 Asiakkaan toiminta ketterässä kehittämisessä	93

TAULUKOT

TAULUKKO 1 Scrumin sisältö	17
TAULUKKO 2 XP:n käytänteet	19
TAULUKKO 3 Lean-ohjelmistokehityksen periaatteet	21
TAULUKKO 4 Yhteenveto sopimusmallien arvioista	36
TAULUKKO 5 Sopimusmallien luokittelu	41
TAULUKKO 6 Arvon määritelmiä ketterän kehittämisen yhteydessä	48
TAULUKKO 7 Arvon määrittämisen menetelmiä	49
TAULUKKO 8 Ansaitun arvon määrittämisen laskelmat.....	51
TAULUKKO 9 Perinteisen ja ketterän ansaitun arvon määrittämisen erot.....	52
TAULUKKO 10 Suhteellisen painotuksen menetelmä	57
TAULUKKO 11 Arvon määrittämisen menetelmien luokittelu.....	60
TAULUKKO 12 Asiakkaan osallistumisen aktiviteetteja	69
TAULUKKO 13 Asiakkaan osallistumisen aktiviteetit yhdisteltyinä	70
TAULUKKO 14 Osallistumisen aktiviteettien luokittelu	82
TAULUKKO 15 Asiakkaan näkökulman osien väliset vuorovaikutussuhteet ..	92

SISÄLLYS

TIIVISTELMÄ	2
ABSTRACT	3
KUVIOT	4
TAULUKOT	5
SISÄLLYS.....	6
1 JOHDANTO.....	8
2 KETTERÄ OHJELMISTOKEHITYS	12
2.1 Taustaa	12
2.2 Määritelmä.....	14
2.3 Ketterän kehittämisen menetelmiä	15
2.3.1 Scrum	15
2.3.2 Extreme Programming (XP).....	18
2.3.3 Lean-ohjelmistokehitys.....	19
2.4 Asiakas ketterässä kehittämisessä.....	21
2.5 Ketterä kehittäminen asiakkaan näkökulmasta	23
2.6 Yhteenveto ketterästä ohjelmistokehityksestä.....	25
3 SOPIMUKSEN TEKEMINEN.....	26
3.1 Taustaa	26
3.2 Sopimusmalleja	27
3.2.1 Kiinteähintainen sopimusmalli ja sen johdannaiset.....	27
3.2.2 Aika-ja-materiaalit -sopimusmalli ja sen johdannaiset.....	29
3.2.3 Hybridisopimusmallit	31
3.3 Sopimusmallien arvioita kirjallisuudesta.....	32
3.4 Sopimusmallit asiakkaan näkökulmasta.....	37
3.4.1 Sopimusmallien arvioinnin kriteerit	37
3.4.2 Sopimusmallien arviointi.....	40
3.5 Yhteenveto sopimusmalleista	43
4 ARVON MÄÄRITTÄMINEN	45
4.1 Arvo ketterässä kehittämisessä.....	45
4.2 Arvon määritelmä.....	46
4.3 Arvon määrittämisen menetelmiä.....	49
4.3.1 Ansaitun arvon määrittäminen ketterään kehittämiseen sovellettuna	50

4.3.2	Taloudellisen arvon määrittäminen	52
4.3.3	Liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittäminen.....	53
4.3.4	Suhteellinen painotus ja negatiivinen arvo	56
4.4	Arvon määrittämisen menetelmät asiakkaan näkökulmasta.....	57
4.4.1	Arvon määrittämisen menetelmien arvioinnin kriteerit	57
4.4.2	Arvon määrittämisen menetelmien arviointi.....	59
4.4.3	Arvon määrittämisen kokonaiskuva.....	62
4.5	Yhteenveto arvon määrittämisestä.....	63
5	OSALLISTUMINEN KEHITTÄMISEN AIKANA	65
5.1	Yleisesti asiakkaan osallistumisesta.....	65
5.2	Osallistumisen aktiviteetteja	67
5.3	Teoreettisia näkemyksiä osallistumisen aktiviteeteista	70
5.3.1	Projektinhallinta	71
5.3.2	Sisällöstä päättäminen	71
5.3.3	Testaus ja hyväksyntä.....	72
5.3.4	Yhteydenpito ja konsultointi	73
5.4	Käytännön kokemuksia osallistumisen aktiviteeteista	74
5.4.1	Projektinhallinta	74
5.4.2	Sisällöstä päättäminen	74
5.4.3	Testaus ja hyväksyntä.....	76
5.4.4	Yhteydenpito ja konsultointi	77
5.4.5	Resurssien tarve.....	77
5.5	Osallistumisen aktiviteettien arviointi asiakkaan näkökulmasta.....	78
5.5.1	Osallistumisen aktiviteettien arvioinnin kriteerit	78
5.5.2	Osallistumisen aktiviteettien arviointi	81
5.6	Yhteenveto osallistumisesta kehittämisen aikana	85
6	KETTERÄN OHJELMISTOKEHITYKSEN KOKONAISKUVA ASIAKKAAN NÄKÖKULMASTA.....	86
6.1	Tulosten pohdinta.....	86
6.1.1	Sopimuksen tekeminen	86
6.1.2	Arvon määrittäminen	88
6.1.3	Osallistuminen kehittämisen aikana	89
6.2	Tuloksista muodostuva kokonaiskuva.....	90
7	YHTEENVETO	95
	LÄHTEET	98

1 JOHDANTO

Ohjelmistomarkkinat voidaan jakaa karkeasti kahteen segmenttiin: massamarkkinoitaviin ohjelmistotuotteisiin ja asiakkaalle räätälöitäviin ohjelmistoihin (Hoch, Roeding, Purkert, Lindner & Miller, 1999, s.27). Näistä segmenteistä ensimmäiselle on ominaista ohjelmiston vaatimusten muodostuminen markkina-analyysin ja ohjelmistoa kehittävän organisaation sisäisen tuotekehityksen perusteella. Tässä tutkielmassa tarkastellaan kuitenkin näistä segmenteistä jälkimmäistä, jolle on ominaista ohjelmiston vaatimusten muodostuminen asiakkaan ja hänen tarpeidensa perusteella. Asiakkaalla tarkoitetaan tässä yhteydessä yleisesti ottaen sitä tahoja, jolla on hallussaan kehittämiseen käytettävät varat ja valtuudet päättää hankinnoista.

Ketterä kehittäminen on viime vuosina yleistynyt merkittävästi ohjelmistotuotannossa, ja yhä useammat organisaatiot ovat jo siirtyneet tai viimeistään siirtymässä siihen. Laajaan verkkokyselyyn vuonna 2011 vastanneista 80 % kertoi organisaationsa ottaneen käyttöön ketterän kehittämisen menetelmiä (VersionOne, 2011). Tällaisia menetelmiä ovat muiden muassa Scrum (Schwaber & Sutherland, 2011) ja Extreme Programming (XP) (Beck, 1999a). Asiakkaan rooli on räätälöitävien ohjelmistojen tapauksessa yleensäkin hyvin tärkeä, mutta ketterässä kehittämisessä se on entisestään korostunut. Ketterä kehittäminen laajentaa asiakkaan roolin perinteisen kehittämisen alku- ja loppuvaiheen osallistumisen sijaan koko kehittämisprojektin ajan kestäväksi ja oikeuttaa asiakkaan osallistumaan kehittämisen ohjaamiseen huomattavasti perinteisiä kehittämismenetelmiä enemmän. Asiakkaan osallistuminen kehittämiseen on mahdollisuuden lisäksi välttämättömyys ja edellytys ketterän kehittämisen onnistumiselle ja sen puuttuminen on suurimpia ketterän kehittämisen epäonnistumisen syitä. (Hoda, Noble & Marshall, 2010.)

Ketterää kehittämistä on tutkittu pääosin kehittäjien näkökulmasta. Asiakkaan roolin keskeisyydestä huolimatta on olemassa varsin vähän teoreettista tutkimusta siitä, miten asiakkaan näkökulma ketterään kehittämiseen tulisi jäsentää. Esimerkkinä tällaisesta tutkimuksesta on Lohanin, Langin ja Conboyn (2010) käsitteellinen konstruktio, joka koostuu neljästä osasta: tietämys asiakkaasta (customer knowledge), asiakkaan osallistuminen, asiakasvaatimukset ja

asiakaspalaute. Tämä kattaa kuitenkin vain osan asiakkaan näkökulmasta. Empiiristä tutkimusta aihealueelta on jonkin verran (esim. Ilieva, Ivanov & Stefanova, 2004; Mann & Maurer, 2005; Martin, Biddle & Noble, 2004a & 2004b; Koskela & Abrahamsson, 2004; Griffin, 2001), mutta laajemman näkökulman sijaan siinä keskitytään kulloinkin vain tiettyihin erityiskysymyksiin.

Tämän tutkielman motiivina on tarve selvittää ketterän ohjelmistokehityksen ominaispiirteitä ja vaatimuksia asiakkaan näkökulmasta. Tätä varten näkökulma jaetaan karkeasti kolmeen osaan: sopimuksen tekeminen, kehitettävän ohjelmiston arvon määrittäminen ja osallistuminen kehittämisen aikana. Jokainen näistä kolmesta osasta sisältää omat haasteensa asiakkaalle. Agile-manifesti (Agile Manifesto, 2001) korostaa asiakasyhteistyön tärkeyttä yli kirjallisten sopimusten, mutta ulkoisten asiakkaiden kanssa toimittaessa sopimusten tekeminen on välttämätöntä (Stevens, 2009a). Ketterään kehittämiseen soveltuvien sopimusmallien ymmärtäminen saattaa kuitenkin tuottaa asiakkaalle ongelmia. Arvon käsite ja sen mukainen priorisointi on keskeinen piirre ketterässä kehittämisessä ja siirtyminen ketterään kehittämiseen merkitsee samalla siirtymistä kustannuskeskeisyydestä arvokeskeisyyteen (Racheva, Daneva & Sikkell, 2009). Asiakkaalla saattaa kuitenkin olla vaikeuksia luotettavasti arvioida ja priorisoida tuotoksia maksimaalisen arvon saavuttamiseksi. Lisäksi ketterä kehittäminen edellyttää asiakkaan aktiivista osallistumista tuotosten määrittelyyn, priorisointiin ja arviointiin (Hoda ym., 2010). Asiakkaan saattaa kuitenkin olla vaikeata omaksua rooliaan aktiivisena osallistujana. Näistä haasteista johdettu tutkimusongelma voidaan esittää seuraavalla tavalla:

Mitä ketterän ohjelmistokehityksen onnistuminen edellyttää asiakkaan näkökulmasta?

Tutkimusongelma voidaan jakaa kolmeksi tutkimuskysymykseksi:

- Millaisia sopimusmalleja ketterässä kehittämisessä käytetään ja mitkä niistä ovat asiakkaan kannalta suositeltavimpia?
- Millaisia menetelmiä on olemassa kehitettävän ohjelmiston arvon määrittämiseen ja mitkä niistä ovat tähän tehtävään asiakkaan kannalta parhaiten soveltuvia?
- Millaisia osallistumismuotoja asiakkaalla on valittavanaan ketterässä kehittämisessä ja mitkä niistä ovat asiakkaan kannalta suositeltavimpia?

Ensimmäisen tutkimuskysymyksen tavoitteena on muodostaa käsitys ketterässä kehittämisessä yleisesti käytettävistä sopimusmalleista sekä niiden riskeistä ja hyödyistä asiakkaan näkökulmasta. Lisäksi tavoitteena on saada käsitys asiakkaan näkökulmasta parhaiten ketterään kehittämiseen soveltuvista sopimusmalleista. Toisen tutkimuskysymyksen tavoitteena on muodostaa kuva menetelmistä ketterän kehittämisen asiakkaalle tuottaman arvon määrittämiseksi, eri menetelmien hyvistä ja huonoista puolista sekä niiden soveltuvuudesta asiakkaan näkökulmasta tarkasteltuna. Kolmannen tutkimuskysymyksen

tavoitteena on kartoittaa asiakkaan mahdollisuuksia osallistua ketterään kehittämiseen ja arvioida eri osallistumismuotojen hyödyllisyyttä asiakkaan näkökulmasta. Koko tutkielman tuloksena odotetaan selkeitä jäsennyksiä asiakkaan roolin vaatimuksista ketterässä kehittämisessä ja keinoista menestyksekkääseen toimimiseen tuossa roolissa.

Tutkielman tuloksena saatavan ymmärryksen voidaan odottaa ensiksikin selventävän niin asiakkaiden kuin kehittäjien käytännön työskentelyä määrittelemällä ketterässä kehittämisessä asiakkaan roolin kuuluvat vastuut. Toiseksi tutkielman tuloksena saatavan ymmärryksen ketterän kehittämisen haasteista asiakkaan näkökulmasta sekä mahdollisuuksista näiden haasteiden voittamiseksi voidaan odottaa auttavan asiakkaita menestymään vaativassa roolissaan ja mahdollistavan asiakkaiden kouluttamisen roolinsa vaatimuksiin. Tällaista asiakkaiden ketterän kehittämisen vaatimuksiin ja käytettävissä oleviin työkaluihin kouluttamisen tärkeyttä korostaa esimerkiksi Van Deursen (2001). Kolmanneksi tutkielman tuloksena saatavan ymmärryksen voidaan odottaa auttavan kehittäjiä ymmärtämään asiakkaan roolin vaatimuksia ja tukemaan asiakkaita tässä roolissa menestymiseksi molempien osapuolten yhteistä hyötyä edistävällä tavalla.

Tutkielma on käsitteellis-teoreettinen ja perustuu aihealueelta julkaistuihin teoreettisiin ja empiirisiin tutkimuksiin. Työssä pyritään välttämään keskittymistä tiettyihin ketterän kehittämisen menetelmiin, mutta milloin se ei ole mahdollista, rajataan käsittely Scrum (Schwaber & Sutherland, 2011) ja Extreme Programming (XP) -menetelmiin (Beck, 1999a) sekä Lean-ajatteluun perustuvaan lähestymistapaan (Poppendieck & Poppendieck, 2003).

Tutkielma on jäsennetty seitsemään lukuun. Luvussa 2 käsitellään ketterää ohjelmistokehitystä. Luvussa esitellään ketterä kehittäminen yleisellä tasolla ja yleisimmin käytetyt ketterät menetelmät ja lähestymistavat (XP, Scrum ja Lean). Lisäksi kuvataan asiakkaan roolia ketterässä kehittämisessä ja määritellään tutkielman loppuosan viitekehyksenä toimiva malli asiakkaan näkökulmasta ketterään kehittämiseen. Luvun tarkoituksena on toimia johdantona ketterään kehittämiseen. Lisäksi luvussa esitetty asiakkaan näkökulman määrittely toimii teoreettisena viitekehyksenä ja rajauksena koko muun tutkielman aihepiirille.

Luvussa 3 käsitellään sopimuksen tekemistä. Luvussa kuvataan ensin sopimuksia yleisesti, esitellään erilaisia ohjelmistokehityksen sopimusmalleja ja käsitellään teoreettisia arvioita ja empiirisiä havaintoja esitellyistä sopimusmalleista. Tämän jälkeen määritellään sopimusten arvioinnin teoreettisena viitekehyksenä toimiva asiakkaan näkökulma sekä lopuksi arvioidaan sopimusmalleja tästä näkökulmasta.

Luvussa 4 käsitellään arvon määrittämistä. Arvon käsitteen tarkastelun ja määrittämisen jälkeen esitellään erilaisia arvon määrittämisen menetelmiä ja määritellään menetelmien arvioinnin viitekehyksenä toimiva asiakkaan näkökulma sekä lopuksi vertaillaan eri menetelmiä tästä näkökulmasta.

Luvussa 5 käsitellään asiakkaan osallistumista kehittämisen aikana. Ensimmäisen tarkastellaan yleisesti asiakkaan osallistumista ja esitellään asiakkaan osallistumisen aktiviteetteja. Tämän jälkeen tarkastellaan kirjallisuudesta löytyviä teo-

reettisiä arvioita ja empiirisiä kokemuksia esitellyistä aktiviteeteista. Kolmanneksi luvussa määritellään aktiviteettien käsittelyn teoreettisena viitekehysenä toimiva asiakkaan näkökulma ja arvioidaan aktiviteetteja tästä näkökulmasta..

Luvussa 6 esitetään kolmen edellisen luvun tulokset, niihin liittyvä pohdinta sekä kootaan ne yhteen muodostaen kokonaiskuvan asiakkaan näkökulmasta ketterään kehittämiseen. Tutkielma päättyy yhteenvetoon.

2 KETTERÄ OHJELMISTOKEHITYS

Tässä luvussa käsitellään ketterää ohjelmistokehitystä yleisellä tasolla. Ensin esitellään sen taustoja ja pohditaan ketteryyden käsitteen sisältöä sekä määritellään ketterä kehittäminen. Sen jälkeen kuvataan lyhyesti yleisimmin käytetyt ketterän ohjelmistokehityksen menetelmät (Scrum ja XP) ja lähestymistapa (Lean). Kolmanneksi luvussa kuvataan asiakkaan roolia ketterässä kehittämisessä ja määritellään tutkielman loppuosan viitekehyksenä toimiva asiakkaan näkökulma ketterään kehittämiseen. Luku päättyy yhteenvetoon.

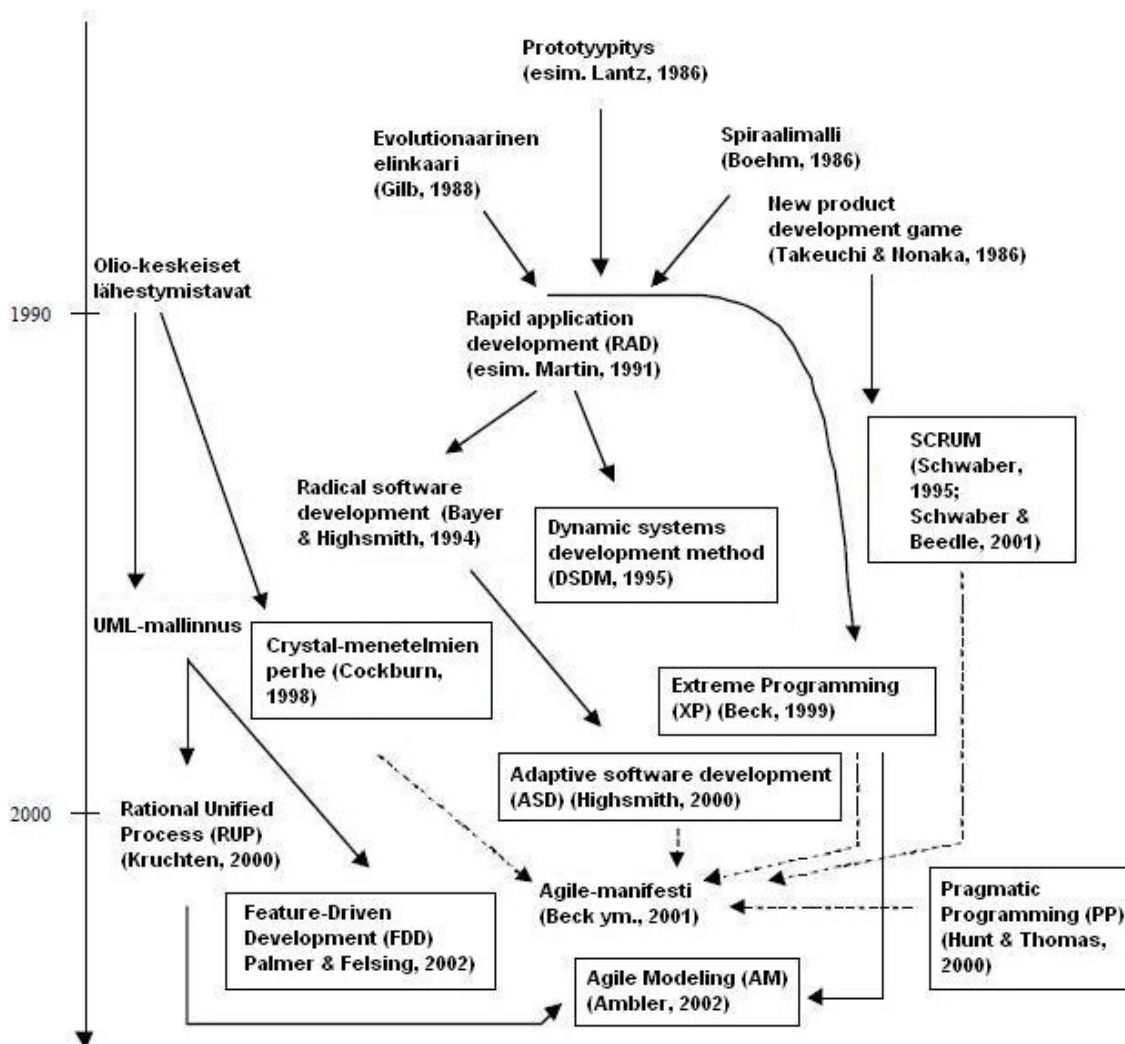
2.1 Taustaa

Ketterän kehittämisen käsite pohjautuu vuonna 2001 julkaistuun Agile-manifestiin (Agile Manifesto, 2001), jonka mukaan ketterässä kehittämisessä arvostetaan:

- yksilöitä ja vuorovaikutusta enemmän kuin prosesseja ja työkaluja
- toimivaa ohjelmistoa enemmän kuin kattavaa dokumentaatiota
- asiakasyhteistyötä enemmän kuin sopimusneuvotteluita
- muutokseen vastaamista enemmän kuin suunnitelman seuraamista

Agile-manifesti ei kuitenkaan ollut alku ketterän kehittämisen ajatuksille. Abbasin, Gravellin ja Willsin (2008) mukaan useat ketterän kehittämisen ajatuksista olivat tunnettuja jo kauan ennen ketterän kehittämisen käsitteen syntymistä. Monet uskoivat juuri senkaltaisten ajatusten olevan paras tapa uuden tyyppisten sovellusten kehittämiseen, mutta aiemmin noita ajatuksia ei kuitenkaan vielä laajalti otettu vakavasti (Abbas ym., 2008). Cohenin, Lindvallin ja Costan (2004) mukaan tiedostettiin tarve löytää vaihtoehto raskaille dokumentaatiokeskeisille kehitysmenetelmille, joiden toimintatapoja useat pitivät turhauttavina ja jopa mahdottomina toteuttaa. Abrahamsson, Warsta, Siponen ja Ronkainen (2003) ovat havainnollistaneet ketterän kehittämisen menetelmien evo-

luutiota sekä Agile-manifestin taustalla olleita ajatuksia ja menetelmiä kuvion 1 mukaisesti (kuviossa mainitut nimet ja vuosiluvut viittaavat lähteiden sijasta menetelmien kehittäjiin ja ajankohtaan). Kuvioista voidaan havaita ketterien menetelmien historiallisten juurien pohjautuvan yhtäältä prototyypityksen (Lantz, 1986), evolutionaarisen elinkaaren (Gilb, 1988) ja spiraalimallin (Boehm, 1988) ajatuksiin. Toisaalta lähtökohtia ovat tarjonneet myös oliokeskeiset lähestymistavat ja UML sekä autoteollisuudesta alkunsa saanut tuotantomalli (Takeuchi & Nonaka, 1986). Näin on syntynyt laaja kirjo erilaisia ketteriä menetelmiä kuten Scrum (Schwaber, 1995), Crystal-menetelmien perhe (Cockburn, 2002), XP (Extreme Programming) (Beck, 1999a), Pragmatic Programming (Hunt & Thomas, 1999) ja ASD (Adaptive software development) (Highsmith, 2000).



KUVIO 1 Ketterän kehittämisen menetelmien evoluutio (Abrahamsson ym., 2003, s.246)

Highsmithin ja Cockburnin (2001) ja Cohenin ym. (2004) mukaan ketterän kehittämisen käytänteet eivät sinällään ole mitään uutta vaan uutta ovat sen sijaan ketterän kehittämisen arvot ja huomion kohteet (Cohen ym., 2004). Highsmithin

ja Cockburnin (2001) mukaan ketterässä kehittämisessä uutta on ihmisten arvostaminen kehittämisen pääasiallisina menestystekijöinä sekä huomion kiinnittäminen vaikuttavuuteen (effectiveness) ja mukautuvuuteen (maneuverability). Nämä yhdessä muodostavat yhdistelmän arvoja ja periaatteita, jotka määrittelevät ketterän kehittämisen. (Highsmith & Cockburn, 2001.)

2.2 Määritelmä

Abbasin ym. (2008) mukaan käsitykset ketterän kehittämisen käsitteestä vaihtelevat, ja sitä on vaikea määritellä tarkasti. Osa tutkijoista pyrkii määrittelemään ketterän kehittämisen sen taustalla vaikuttavan filosofian kautta, kun taas toiset näkevät sen eri menetelmille yhteisten käytänteiden kautta (Abbas ym., 2008).

Highsmithin ja Cockburnin (2001) mukaan ketterässä kehittämisessä on pohjimmiltaan kyse luovuudesta ja muutokseen vastaamisesta. Kehittämisprojekteissa tapahtuu kehittämisen kuluessa ulkoisista syistä johtuvia muutoksia esimerkiksi vaatimuksissa tai teknologiassa. Nämä muutokset ovat käytännön välttämätön paha. Sen vuoksi kysymys ei voikaan olla siitä, kuinka estää muutokset vaan kuinka paremmin vastata näihin välttämättömiin muutoksiin koko kehittämisen ajan. Perinteisen kehittämisen menetelmät perustuvat olettamukselle, että on mahdollista tuottaa kattava ja muuttumaton lista vaatimuksia alkuvaiheessa. Mutta käytännössä tällainen muutoksen vastustaminen merkitsee välipitämättömyyttä liiketoimintaympäristön muutoksista. Ketterä kehittäminen on vastaus tähän tarpeeseen, koska sen tavoite on alentaa muutosten kustannuksia koko kehittämisprojektin ajan. (Highsmith & Cockburn, 2001.)

Miller (2001) tarjoaa hyvin yksityiskohtaisen määritelmän ketterälle kehittämiselle kuvaillessaan sitä yhdeksällä piirteellä, jotka ovat: modulaarisuus menetelmän tasolla, iteratiivisuus lyhyissä sykleissä, aikasidonaisuus iteraatioiden osalta, resurssien järkevä käyttö välttämällä turhia aktiviteetteja, mukautuvuus, inkrementaalisuus, konvergenssi, ihmiskeskeisyys ja yhteistyö.

Abrahamsson, Salo, Ronkainen ja Warsta (2002) puolestaan painottavat ketterän kehittämisen tärkeimpinä piirteinä yksinkertaisuutta ja nopeutta. Ketterässä kehittämisessä keskitytään toteuttamaan aluksi nopeasti tärkeimmät ominaisuudet, jonka jälkeen saadaan palautetta ja reagoidaan siihen. Lisäksi Abrahamsson ym. (2002) määrittelevät ketterälle kehittämiselle ominaisiksi piirteiksi inkrementaalisuuden, yhteistyön ja mukautuvuuden.

Erickson, Lyytinen ja Siau (2005) määrittelevät ketteryyden merkitsevän pohjimmiltaan sitä, että riisutaan pois mahdollisimman paljon perinteisen kehittämisen menetelmiin yleisesti liitettyä raskautta, jotta mahdollistetaan nopea vastaaminen muuttuviin olosuhteisiin, muutoksiin käyttäjävaatimuksissa, muuttuviin aikatauluihin ja muihin vastaaviin.

Abbas ym. (2008) päätyvät aiempaan tutkimukseen kattavasti perehdyttyään määrittelemään ketterän kehittämisen niiden piirteiden mukaan, joita menetelmän tulee ilmentää ollakseen ketterä. Nämä piirteet ovat: mukautuvuus, iteratiivisuus ja inkrementaalisuus sekä ihmiskeskeisyys. Ohjelmistokehitys on

ennustamatonta ja ennustavuutta kontrolloimaan tarvitaan mukautuva prosessi. Tämä tapahtuu parhaiten iteratiivisuuden ja inkrementaalisuuden avulla. Lisäksi tarvitaan luovia ja lahjakkaita ihmisiä. (Abbas ym. 2008.)

Conboy (2009) esittelee kattavan analyysin pohjalta oman lopullisen määritelmänsä ketterydestä, jonka mukaan ketteryys on kehittämismenetelmän ”jatkuvaa valmiutta nopeasti tai luonnostaan luoda muutosta, proaktiivisesti tai reaktiivisesti hyväksyä muutos ja oppia muutoksesta samalla kun edistää asiakkaan arvoa (taloudellista, laadullista ja yksinkertaisuutta), ympäristön kanssa yhteisten osiensa ja suhteidensa kautta” (Conboy, 2009, s.340).

Kaikki edellä käsitellyt ketterän kehittämisen määritelmät lähestyvät asiaa filosofiselta kannalta käsittelemällä ketterän kehittämisen takana vaikuttavia ajatuksia ja periaatteita. Käsitellyissä määritelmissä yleisimmin esiintyvät piirteet näyttävät olevan muutoksen hyväksyminen, pyrkimys tuottaa asiakkaalle arvoa nopeasti ja jatkuvasti, menetelmän keveys, mukautuvuus ja yksinkertaisuus, toteutus iteratiivisesti ja inkrementaalisesti sekä ihmiskeskeisyyden ja yhteistyön korostaminen. Näiden piirteiden voidaankin katsoa olevan tässä tutkielmassa käytettävä määritelmä ketterälle kehittämiselle: *ketterä kehittäminen* on keveiden, mukautuvien ja yksinkertaisten kehitysmenetelmien käyttöä iteratiivisesti ja inkrementaalisesti suoritettavassa ohjelmistokehityksessä, jossa korostetaan ihmiskeskeisyyttä ja yhteistyötä, tavoitteena tuottaa asiakkaalle arvoa nopeasti ja jatkuvasti sekä hyväksyen muutokset.

2.3 Ketterän kehittämisen menetelmiä

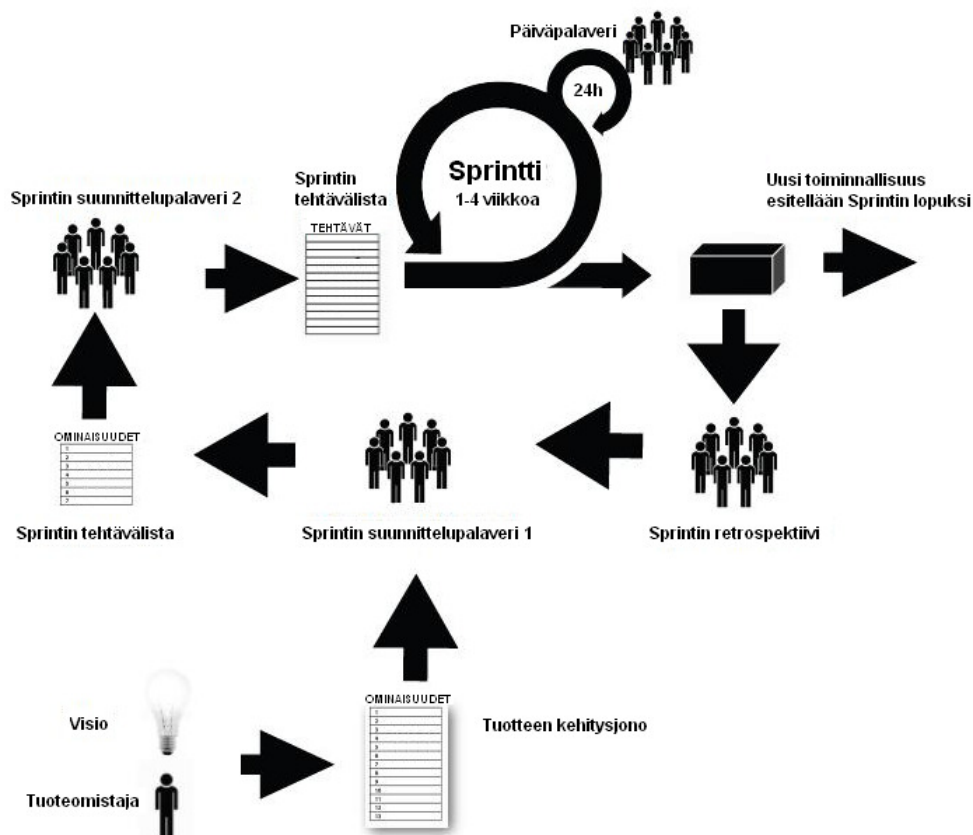
Tässä alaluvussa esitellään ensin kaksi eniten tutkittua ja yleisimmin käytettyä ketterän kehittämisen menetelmää eli Scrum (Schwaber & Sutherland, 2011) ja Extreme Programming (XP) (Beck, 1999a). Niiden jälkeen kuvataan Lean-ajatteluun perustuvaa lähestymistapaa ohjelmistokehitykseen (Poppendieck & Poppendieck, 2003). Yleisimmin käytettyjen menetelmien ja lähestymistapojen valinta perustuu VersionOne-yrityksen (2011) teettämän laajan verkkokyselyn tuloksiin. Mainitun kyselyn tulosten mukaan 52 % vastanneista kertoi käyttävänsä ketterän kehittämisen menetelmistä Scrumia, 2 % XP:tä ja 14 % Scrum/XP-hybridimenetelmää (VersionOne, 2011). Lean-ajatteluun perustuvaa lähestymistapaa kertoi käyttävänsä 2 % vastanneista (VersionOne, 2011).

2.3.1 Scrum

Scrumin esitteli alun perin Ken Schwaber vuoden 1995 artikkelissaan (Schwaber, 1995). Siinä hän kuvailee Scrumia menetelmäksi, joka hyväksyy kehittämisen ennustamattomuuden ja määrämuotoistaa (formalize) usein kompleksisissa ja jopa kaoottisissa kehittämissuhteissa käytössä olevan ”tee mitä täytyy”-asenteen. Scrum-termi on peräisin rugby-pelistä, jossa se merkitsee hyökkäävistä pelaajista koostuvaa aloitusmuodostelmaa (Schwaber, 1995).

Schwaberin ja Sutherlandin (2011, s.3) mukaan "Scrum on viitekehys, jonka puitteissa ihmiset voivat ratkaista monimutkaisia ongelmia kehittäessään tuotteita tuottavasti ja luovasti pyrkien mahdollisimman korkeaan lisäarvoon". Scrum on kevyt ja yksinkertaisesti ymmärrettävä, mutta erittäin vaikea hallita täysin. Scrum perustuu empiriiräiselle prosessinhallinnan teorialle ja käyttää iteraatiivista, inkrementaalista lähestymistapaa ennustettavuuden optimoimiseksi ja riskin hallitsemiseksi. (Schwaber & Sutherland, 2011.) Taulukossa 1 on esitetty Scrumin sisältö Schwaberin ja Sutherlandin (2011) mukaisesti jaoteltuna rooleihin, tapahtumiin (events) ja tuotoksiin (artefacts).

Scrumin mukaista toimintaa voidaan Sutherlandin (2010) mukaisesti havainnollistaa kuvion 2 tavoin. Aluksi tuoteomistaja konkretisoi visionsa tuotteesta vaatimusten muotoon ja koostaa priorisoiduista vaatimuksista tuotteen kehitysjonon. Sprintin suunnittelupalaverin ensimmäisessä osassa tuotteen kehitysjonosta valitaan vaatimuksia sprintin tehtävälisterille, ja sprintin suunnittelukokouksen toisessa osassa ne jaetaan yksityiskohtaisemmiksi tehtäviksi. Sprintin pituus on 1-4 viikkoa, jonka aikana pidetään päivittäisiä päiväpalaveri- ja sprintin lopuksi toimitetaan ja esitellään uudet toiminnallisuudet sprinttikatselmuksessa. Lopuksi pidetään sprintin retrospektiivi. Tämän jälkeen aloitetaan jälleen seuraavaan sprinttiin valmistautuminen sprintin suunnittelupalaverilla. (Sutherland, 2010.)



KUVIO 2 Scrumin prosessi (Sutherland, 2010, s.11)

TAULUKKO 1 Scrumin sisältö (koostettu lähteestä Schwaber & Sutherland, 2011)

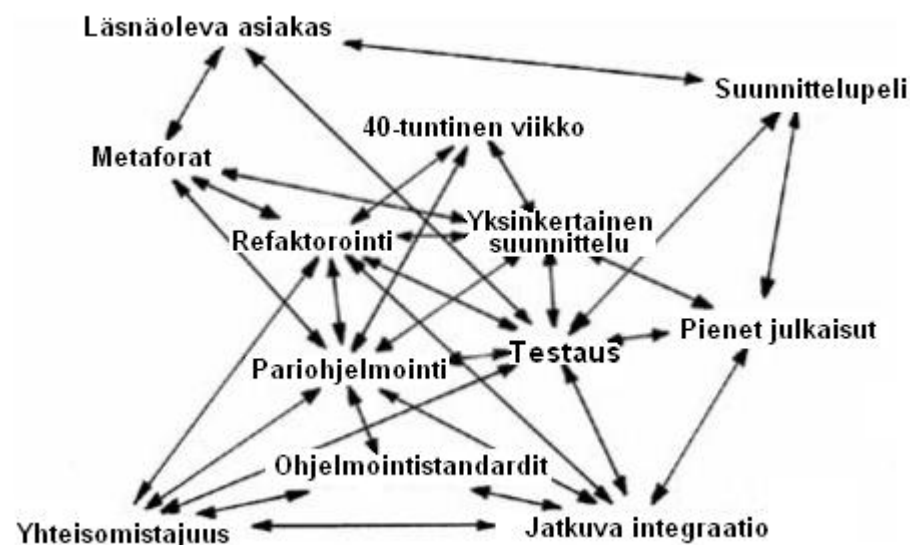
Tyyppi	Käytänteet	Selitys
Rooli	Kehitystiimi	<i>Kehitystiimi</i> koostuu ohjelmiston kehittäjistä ja suorittaa toteutustyötä itseohjautuvasti. Kehitystiimillä ei ole perinteistä projektipäällikköä, vaan se on itse vastuussa työstään. Kehitystiimin jäsenillä tulisi taustastaan ja työnkuvastaan huolimatta olla kollektiivinen vastuu kaikista suoritettavista toimenpiteistä.
Rooli	Scrum-mestari	<i>Scrum-mestari</i> on Scrum-tiimin jäsen, jonka vastuulla on ryhmän työn esteiden poistaminen ja menetelmän oikeasta soveltamisesta huolehtiminen. Scrum-mestari on Scrum-tiimin palveleva johtaja (servant-leader).
Rooli	Tuoteomistaja	<i>Tuoteomistaja</i> on henkilö, jonka vastuulla on kehitettävän ohjelmistotuotteen arvon optimointi. Hän vastaa yksin vaatimuksista sekä niiden priorisoinnista.
Rooli	Scrum-tiimi	<i>Scrum-tiimi</i> koostuu kehitystiimistä, Scrum-mestarista ja tuoteomistajasta.
Tapahdus	Sprintti	<i>Sprintti</i> on etukäteen määritellyn pituinen (yleensä 1-4 viikkoa) ajanjakso, jonka aikana toteutetaan uusi ohjelmiston tuoteversio. Jokainen sprintti sisältää suurimman osan ohjelmistokehitysprosessin vaiheista alkaen olemassa olevan järjestelmän analysoinnista ja päättyen uuden toteutuksen kautta julkaisuun ja arviointiin. Sprintin aikana Scrum-tiimin koostumus ja tavoitteet pysyvät muuttumattomina.
Tapahdus	Päiväpalaveri	<i>Päiväpalaveri</i> on kehitystiimin päivittäinen 15-minuutin mittainen kokous, jossa kerrataan edellisen kokouksen jälkeen tehty työ, suunnitellaan seuraavaan kokoukseen mennessä tehtävä työ ja arvioidaan sprintin tavoitteen edistyminen.
Tapahdus	Sprintin suunnittelupalaveri	<i>Sprintin suunnittelupalaveri</i> koostuu kahdesta osiosta. Ensimmäisessä osiossa Scrum-tiimi sopii alkavan sprintin tavoitteet ja valitsee tuotteen kehitysjonosta vaatimuksia toteutettavaksi sprintissä. Tämän jälkeen kokouksen toisessa osiossa kehitystiimi suunnittelee sprintin käytännön toteutuksen ja jakaa toteutettavaksi valitut vaatimukset pienemmiksi tehtäviksi.
Tapahdus	Sprinttikatselmus	<i>Sprinttikatselmus</i> on sprintin lopetuskokous, jossa Scrum-tiimi esittelee sprintin aikana tehdyn työn ja uuden ohjelmiston tuoteversion tuoteomistajalle ja muille mahdollisesti kiinnostuneille sidosryhmille.
Tapahdus	Sprintin retrospektiivi	<i>Sprintin retrospektiivi</i> on tapaaminen, johon Scrum-tiimi kokoontuu käymään läpi sprintin aikana tapahtunutta ja kehittämään työskentelytapojaan.
Tuotos	Tuotteen kehitysjono	<i>Tuotteen kehitysjono</i> on priorisoitu lista kehitettävän ohjelmistotuotteen vaatimuksista sekä muista tehtävistä. Tuotteen kehitysjonon ylläpito on tuoteomistajan vastuulla, ja sitä päivitetään koko kehittämisprojektin ajan poistaen, lisäten tai muokaten vaatimuksia. Tuotteen kehitysjonosta poimitaan tärkeimmiksi priorisoituja vaatimuksia sprintin tehtävällyställe toteutettavaksi sprintissä.
Tuotos	Sprintin tehtävällystä	<i>Sprintin tehtävällystä</i> koostuu sprintin aikana toteutettavaksi sovitusta tehtävistä. Se sisältää tuotteen kehitysjonosta poimittuja vaatimuksia jaettuina pienemmiksi tehtäviksi.
Tuotos	Tuoteversio	<i>Tuoteversio</i> sisältää kaiken tuotteen kehitysjonosta siihen saakka toteutetun toiminnallisuuden. Jokaisen sprintin loppuun valmistuu uusi tuoteversio, ja jokaisen valmistuneen tuoteversion tulee olla toimiva kokonaisuus.

2.3.2 Extreme Programming (XP)

XP on Kent Beckin ensimmäistä kertaa vuonna 1998 esittelemä ja artikkelilla Beck (1999b) laajempaan tietoisuuteen tuoma menetelmä. XP koostui alkujaan neljästä arvosta (kommunikointi, yksinkertaisuus, palaute, rohkeus), viidestä periaatteesta (nopea palaute, yksinkertaisuuden olettaama, inkrementaalinen muutos, muutoksen hyväksyminen, työn laatu) ja 12 käytänteestä (Beck, 1999a). Beck ja Andres (2004) lisäsivät menetelmään vielä viidennen arvon (kunnioitus), esittelivät lisäyksiä käytänteisiin sekä jakoivat käytänteet ensisijaisiin ja täydentäviin. Tässä yhteydessä kuitenkin käsittelemme XP:n Beckin (1999a) esittelemässä muodossa.

Dybån ja Dingsøyryn (2008) mukaan XP keskittyy pääasiassa ohjelmistokehityksen parhaisiin käytänteisiin. Voidaankin sanoa XP:n ytimen muodostuvan juuri käytänteistä, arvojen ja periaatteiden toimiessa käytännössä lähinnä teoreettisena taustatukena. Taulukossa 2 on esitetty Beckin (1999a, 1999b) mukaisesti XP:n käytänteet eli suunnittelupeli, pienet julkaisut, metaforat, yksinkertainen suunnittelu, testit, refaktorointi, pariohjelmointi, yhteisomistajuus, jatkuva integraatio, 40-tuntinen viikko, läsnäoleva asiakas (on-site customer) ja ohjelmointistandardit.

Beckin (1999a, 1999b) mukaan mitkään XP:n ajatuksista eivät yksittäisinä ole suinkaan uusia vaan "yhtä vanhoja kuin ohjelmointi itse". Sen sijaan XP:n innovaatio on näiden ajatusten kerääminen yhteen, niiden oikean ja perinpohjaisen hyväksikäyttämisen varmistaminen ja ajatusten järjestäminen tukemaan toisiaan. Kuviossa 3 on esitetty, mitkä käytänteistä Beckin (1999a) mukaan tulevat toisiaan. (Beck, 1999a & 1999b). Myös Abrahamsson ym. (2002) toteavat, etteivät XP:n yksittäiset käytänteet ole sellaisinaan uusia. Heidän mukaansa ne on kuitenkin XP:ssä kerätty ja järjestetty toimimaan toistensa kanssa uudella tavalla. Samoin Cohenin ym. (2004) mukaan XP:n vahvuutena ei ole mitään yksittäinen ajatus vaan niiden yhdistelmästä syntyvät emergentit ominaisuudet.



KUVIO 3 XP:n käytänteiden suhteet (Beck, 1999a, s.58)

TAULUKKO 2 XP:n käytänteet (koostettu lähteistä Beck, 1999 & 1999b)

Käytänteet	Selitys
Suunnittelu-peli	Asiakkaat priorisoivat käyttäjätarinat ja päättävät julkaisujen sisällön ja aikataulun kehittäjien työmääräarvioihin perustuen.
Pienet julkaisut	Ohjelmiston ensimmäinen versio julkaistaan hyvin pian ja uusia julkaisuja tehdään sen jälkeen useasti, jopa päivittäin.
Metaforat	Kehittäjien ja asiakkaan välistä kommunikaatiota helpotetaan kehittämällä metaforia ohjelmistosta ja kommunikoimalla niiden avulla.
Yksinkertainen suunnittelu	Pyritään mahdollisimman yksinkertaiseen, halutun tarkoituksen täyttävään ohjelmiston rakenteeseen. Ylimääräinen tai päällekkäinen osa toteutuksesta poistetaan välittömästi turhana.
Testit	Kehittäminen suoritetaan testausvetoisesti. Kehittäjät kirjoittavat jokaiselle ohjelmiston osalle automaattisesti ajettavat yksikkötestit, joiden kaikkien kuuluu tulla jatkuvasti hyväksytyksi suoritetuiksi. Asiakkaat kirjoittavat toiminnallisen testauksen testitapaukset käyttäjätarinoiden yhteyteen.
Refaktorointi	Ohjelmiston rakennetta parannetaan muuttamatta sen toiminnallisuutta.
Pariohjelmointi	Kaksi kehittäjää työskentelee samanaikaisesti yhdellä tietokoneella.
Yhteisomistajuus	Jokainen kehittäjä on yhteisesti vastuussa jokaisesta ohjelmiston osasta ja on velvollinen korjaamaan ja parantamaan sitä mikäli ilmenee tarvetta.
Jatkuva integraatio	Uusi ohjelmiston osa integroidaan ja testataan välittömästi toteutuksen valmistumisen jälkeen. Ohjelmisto integroidaan ja käännetään jopa useita kertoja päivässä ja uuden ohjelmiston osan tulee läpäistä automaattisesti ajettavat testit tullakseen hyväksytyksi.
40-tuntinen viikko	Kehittäjien tulee pyrkiä työskentelemään normaalin työajan puitteissa ilman ylityitä. Kaksi peräkkäistä viikkoa ylityitä tulkitaan ongelmatilanteeksi.
Läsnäoleva asiakas	Asiakkaan tai sellaisena toimivan tulee olla paikalla ja kehitystiimin käytävissä kokoaikaisesti.
Ohjelmointi-standardit	Yhteisomistajuuden ja pariohjelmoinnin soveltaminen edellyttävät standardoituja työskentelymenetelmiä ohjelmoinnissa. Jokaisen työryhmän jäsenen tulee noudattaa yhtenäistä ohjelmointitapaa, jolloin tekijöistä johtuvat eriävyydet tuotetussa lähdekoodissa vähenevät ja jopa katoavat.

2.3.3 Lean-ohjelmistokehitys

Lean-ajattelun juuret ovat Lean-tuotannossa, joka Likerin (2010) mukaan on alkujaan lähtöisin autovalmistaja Toyotan mullistavasta tuotantomenetelmästä. Likerin (2010) mukaan Toyotan tuotantomenetelmän aikaansaama menestys on mahdollista saavuttaa myös muualla ja tätä tuotantomallia voidaan soveltaa mihin tahansa organisaatioon millä alalla tahansa. Lean-tuotanto saavutti yleisen tietoisuuden Womackin, Jonesin ja Roosin (1990) julkaiseman kirjan myötä. Womack ym. (1990, 1996) määrittelevät Lean-tuotannon prosessiksi, jonka periaatteet ovat (esitetty myös kuviossa 4):

- Arvo: Asiakkaalle tuotettavan arvon tunnistaminen
- Arvovirta: Arvovirran kartoittaminen eli mitkä asiat tuotantoprosessissa lisäävät arvoa ja mitkä eivät

- Virtaus: Tuotantoprosessin virtaus eli tuotantoprosessin läpimenoajan optimointi
- Imu: Tuotantoprosessin imuohjaus (pull) eli tuotanto perustuu asiakkaan tarpeeseen
- Täydellisyys: Erinomaisuuden tavoittelu



KUVIO 4 Lean-tuotanto (Lean Enterprise Institute, 2009)

Lean-ajattelua on sovellettu myös ohjelmistokehitykseen. Ensimmäisten joukossa tätä ovat käsitelleet Poppendieck ja Poppendieck (2003). Taulukossa 3 on esitetty Poppendieckin ja Poppendieckin (2003) mukaisen Lean-ohjelmistokehityksen seitsemän periaatetta: hukan (waste) eliminointi, oppimisen vahvistaminen, päätöksenteko mahdollisimman myöhään, toimittaminen mahdollisimman nopeasti, työryhmän voimaannuttaminen, laadun kehittäminen ja kokonaisuuden näkeminen. Seitsemän periaatteen lisäksi Poppendieckin ja Poppendieckin (2003) mukainen Lean-ohjelmistokehitys sisältää 22 työkalua näiden periaatteiden toteuttamiseksi. Työkaluihin kuuluvat mm. hukan eliminointi -periaatteeseen liittyvät hukan havaitseminen ja arvovirran kartoittaminen sekä kokonaisuuden näkemiseen liittyvät mittaaminen ja sopimukset.

Lean-periaatteiden pohjalta on kehitetty myös Kanban-periaatteet (Anderson, 2010; Kniberg & Skarin, 2009), mutta niiden käsittely rajataan tämän tutkielman ulkopuolelle.

TAULUKKO 3 Lean-ohjelmistokehityksen periaatteet (Poppendieck & Poppendieck 2003, s. 13-15)

Periaate	Selitys
Hukan eliminointi	Hukka on mitä tahansa asiakkaalle arvoa tuottamatonta.
Oppimisen vahvistaminen	Ohjelmistokehitys on uusien ratkaisujen etsimistä kokeilemisen ja oppimisen kautta. Näin ollen ohjelmistokehitys on oppimisprosessi ja paras tapa kehittää ohjelmistoja on vahvistaa oppimista.
Päätöksenteko mahdollisimman myöhään	Muuttuvissa olosuhteissa päätöksenteko mahdollisimman myöhään antaa mahdollisuuden tehdä arvausten sijaan tiedolle perustuvia päätöksiä. Monimutkaisissa järjestelmissä myöhäisen päätöksenteon mahdollistaminen edellyttää muutosvalmiuden sisäänrakentamista järjestelmään.
Toimittaminen mahdollisimman nopeasti	Nopea toimittaminen mahdollistaa myöhäisen päätöksenteon sekä nopean palautteen saamisen, palautteeseen reagoinnin ja siitä oppimisen. Se mahdollistaa myös asiakkaiden juuri tietyllä hetkellä, ei jolloinkin menneisyydessä, tarvitsemien ominaisuuksien toteuttamisen.
Työryhmän voimaantuttaminen	Varsinaista kehitystyötä tekevät ovat riittävän osaamisen ja ohjauksen avulla parhaita asiantuntijoita kehitystyön suhteen. Myöhäinen päätöksenteko ja nopea toimittaminen eivät ole mahdollisia keskusjohtoista päätöksentekoa noudatettaessa.
Laadun kehittäminen	Laatu sisältää asiakkaan kokeman ulkoisen laadun (perceived quality) sekä tuotteen sisäisen laadun (conceptual quality). Tutkimusten perusteella laatua parantavia seikkoja ovat hyvä johtajuus, sopiva asiantuntemus, tehokas kommunikointi ja terve kuri, mutta sen sijaan prosessit ja mittaaminen eivät paranna laatua.
Kokonaisuuden näkeminen	Ohjelmistokehityksessä ongelmana on usein vain ohjelmiston osan optimointi kokonaisuuden jäädessä huomiotta. Yksilöitä ja organisaatioita tulisi arvioida oman osansa kehittämisen sijasta kokonaisuuden toimivuuden kannalta.

2.4 Asiakas ketterässä kehittämisessä

Ketterässä kehittämisessä termi asiakas voi tarkoittaa kehittämisprojektiin liittyvää asiakasorganisaation edustajaa. Asiakas voi olla, mutta ei välttämättä ole kehitettävän tuotteen lopullinen käyttäjä. Asiakas voi olla myös kehittävässä organisaatiossa työskentelevä toimiala-asiantuntija. (Mohammadi, Nikkhahan & Sohrabi, 2009.)

Tässä tutkielmassa termillä *asiakas* tarkoitetaan sitä tahoa, jolla on hallussaan kehittämiseen käytettävät varat ja valtuudet päättää hankinnoista. Tämä on sama merkitys, jota Mohammadi ym. (2009) artikkelissaan käyttävät. Käytännössä tämä voi tarkoittaa kehittäjäorganisaation sisäisissä kehittämisprojekteissa esimerkiksi budjettivastuullisen tuotepäällikön toimimista asiakkaana. Ulkoisen asiakkaan tapauksessa tämä voi puolestaan tarkoittaa esimerkiksi riittävät velvollisuudet ja valtuudet omaavaa projektipäällikköä tai asiakkaan hyväksymistä edustajista koostuvaa ohjausryhmää.

Läheinen yhteistyö asiakkaan kanssa on ketterän kehittämisen keskeisimpiä periaatteita. Agile-manifesti (Agile Manifesto, 2001) korostaa asiakasyhteistyötä yhtenä neljästä ketterän kehittämisen keskeisestä arvosta. Lisäksi Agile-manifestin korostama yksilöiden ja vuorovaikutuksen arvostaminen pitää sisälleen asiakkaan arvostamisen sekä toimivan yhteistyön asiakkaan ja toimittajan välillä.

Ketterän kehittämisen käsitettä määriteltäessä eräänä pääkohtana pidetään aina nimenomaan ihmiskeskeisyyttä ja yhteistyötä. Tähän kuuluu tärkeänä osana läheinen asiakassuhde (Highsmith & Cockburn, 2001). Abbasin ym. (2008) mukaan ketterä kehittäminen korostaa nimenomaan henkilökohtaista suhdetta asiakkaan kanssa kommunikoinnissa ja osallistumisessa verrattuna kirjoitettujen dokumenttien kautta tapahtuvaan.

Ketterän kehittämisen menetelmistä esimerkiksi Scrumissa asiakkaan tehtäviin kuuluu käyttäjätarinoiden ja toiminnallisten testien tuottaminen, prioriteeteista päättäminen ja tuotosten hyväksyntä (Abrahamsson ym., 2002). XP:ssä läsnäolevan asiakkaan tehtäviin kuuluvat asiakkaan ja loppukäyttäjän tarpeiden tunteminen, kommunikointi kehittäjien kanssa esimerkiksi vaatimuksista, toiminnallisten testien tuottaminen ja niiden suorituksen valvominen, iteraatioiden ja julkaisujen suunnitteluun osallistuminen sekä yhteydenpito johtoon (Van Deursen, 2001). Myös muissa menetelmissä vastuut ovat hyvin samanlaisia. Tärkeätä on huomata, että asiakkaan osallistuminen on paitsi oikeus, myös velvollisuus. Ketterää kehittämistä ei ole mahdollista suorittaa, ainakaan menestyksellisesti, ilman asiakkaan roolin täyttämistä.

Boehmin (2002) ja Van Deursenin (2001) mukaan ketterä kehittäminen esittää merkittäviä osaamisvaatimuksia asiakkaalle. Boehmin (2002) mukaan asiakkaan tulisi omata riittävästi tietoa koko sovellusalueesta. Lisäksi Boehm (2002) listaa ketterän kehittämisen asiakkaalle asettamat vaatimukset seuraavasti: omistautuneita, osaavia, yhteistyökykyisiä, oikeita asioita ajavia ja riittävästi valtuutettuja. Van Deursenin (2001) mukaan asiakkaan sovellusalueosaamisella ei sen sijaan ole suurta merkitystä, vaan tärkeämpää olisi kouluttaa asiakkaan roolissa toimivia ammattimaisia asiakkaita sovellusalueesta riippumattomiin asiakkuuden taitoihin. Tämän jälkeen nämä ammattimaiset asiakkaat toimisivat välittäjinä varsinaiset ratkaisut tekevän asiakkaan ja kehittäjien välillä. Highsmithin ja Cockburnin (2001) mukaan mikäli asiakkaalla, oli kyseessä sitten sisäinen tai ulkoinen asiakas, ei ole selkeää käsitystä suunnasta, koko kehittämiseltä puuttuu suunta. Huonot asiakkaat johtavat huonoihin järjestelmiin. Boehmin (2002) mukaan käytännön kehittämisprojekteissa on havaittu, että mikäli asiakas ei omaa rooliin tarvittavia ominaisuuksia, kehitetyt tuotteet eivät yleensä tule olemaan käytännössä tyydyttäviä, vaikka ne tyydyttäisivätkin asiakkaan vaatimukset.

2.5 Ketterä kehittäminen asiakkaan näkökulmasta

Suuri osa ketterää kehittämistä koskevasta kirjallisuudesta tarkastelee ketteriä menetelmiä kehittäjiä kiinnostavista näkökulmista. Tämän tutkielman tarkoitus on kuitenkin tutkia asiaa nimenomaan asiakkaan näkökulmasta. Siksi on aluksi tarpeen määritellä, mitä tarkoitetaan asiakkaan näkökulmalla.

Lohan ym. (2010) pyrkivät löytämään käsitteellisen konstruktion asiakkaaseen liittyvien asioiden jäsentämiseksi ketterässä kehittämisessä. He totesivat, että liiketoimintaa käsittelevässä kirjallisuudessa tunnetaan TQM:n (Total Quality Management) (Ahire, Goldhar & Waller, 1996; Issac, Rajendran & Anantharaman, 2004) yhteydessä asiakasnäkökulma (customer focus), johon lasketaan kuuluviksi muiden muassa asiakaspalaute ja asiakkaan osallistuminen. Tietojärjestelmien ja niiden kehittämistä koskevassa kirjallisuudessa viitataan asiakkaaseen usein kommunikoinnin (Korkala, Pikkarainen & Conboy, 2009), asiakastyytyväisyyden (Mann & Maurer 2005) ja asiakkaan osallistumisen (Kautz, 2009) yhteydessä. Osittain samoja aiheita on esillä myös ketterää kehittämistä koskevassa kirjallisuudessa. Lohan ym. (2010) päätyivät esittämään asiakasnäkökulmaa, joka koostuu neljästä osasta: tietämys asiakkaasta (customer knowledge), asiakkaan osallistuminen, asiakasvaatimukset ja asiakaspalaute.

Tässä tutkielmassa halutaan saada laajempi näkökulma asiakkaan rooliin. Tästä syystä edellisen lisänä käytetään Dybån ja Dingsøyryn (2008) tekemää systemaattista kirjallisuuskatsausta ketterää kehittämistä käsittelevistä empiirisistä tutkimuksista. Tällaisia tutkimuksia ovat: Ilieva ym. (2004), Mann ja Maurer (2005), Martin ym. (2004a, 2004b) sekä Koskela ja Abrahamsson (2004). Edellä mainittujen tutkimusten lähdeviitteistä löydettiin lisäksi vielä Griffinin (2001) asiakkaiden käytännön havaintoja käsittelevä tutkimus.

Ilievan ym. (2004) tutkimus käsittelee tutkijoiden kehittämän XP-pohjaisen menetelmän soveltamista käytäntöön. Tutkimukseen kirjatut asiakkaan havainnot koskevat tyytyväisyyttä mahdollisuuksiin ohjata kehitystä ja antaa jatkuvasti palautetta. Koskelan ja Abrahamssonin (2004) tutkimus käsittelee yhtä tapausta XP:n soveltamisesta käytäntöön. Tutkimuksessa tehdään havaintoja läsnäolevan asiakkaan roolin vaativuudesta ja työläydestä (Koskela & Abrahamsson, 2004). Mann ja Maurer (2005) käsittelevät tutkimuksessaan yhtä tapausta Scrum-menetelmän soveltamisesta käytäntöön. Tulokset tuovat ilmi asiakkaiden yleistä tyytyväisyyttä ketterään kehittämiseen, ja erityisenä tyytyväisyyden aiheena asiakkaat mainitsevat mahdollisuuden osallistua jokapäiväiseen kehittämiseen huomattavasti aiempaa enemmän.

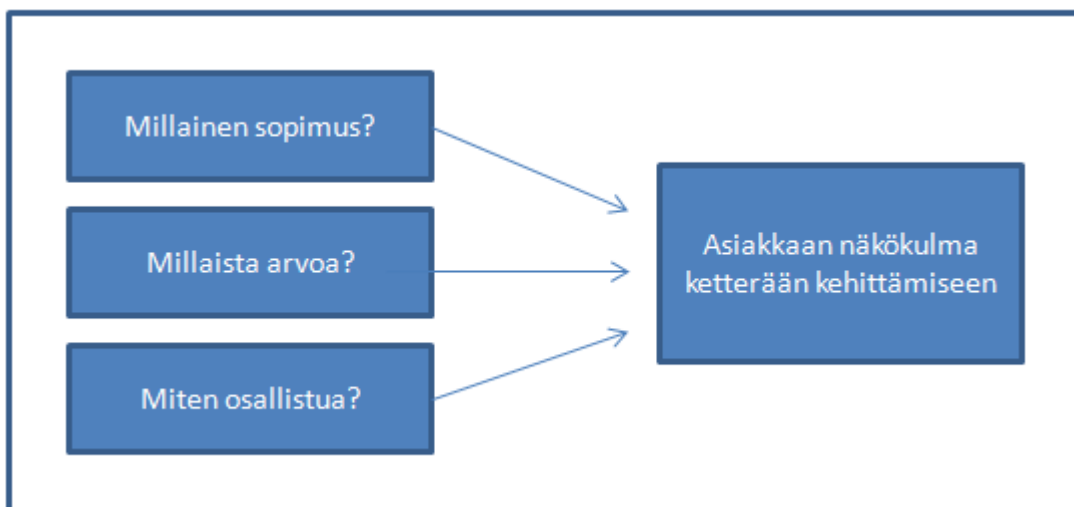
Martinin ym. (2004a) tutkimus käsittelee kolmea tapausta XP:n soveltamisesta käytäntöön. Tulosten mukaan asiakkaiden merkittävimmät havainnot käsittelevät mm. asiakkaan roolin työläyttä, vaatimusten asteittaisen tarkentumisen hyötyjä, vaatimusten priorisoinnin vaikeutta sekä ongelmia tasapainoilussa liiketoimintavaatimusten ja teknisten vaatimusten välillä. Martinin ym. (2004b) XP:tä ja alihankintaa kahden tapauksen osalta käsittelevässä tutkimuksessa on

esitetty huomioita pääasiassa sopimusmalleista, erityisesti aika-ja-materiaalit – tyyppisen sopimuksen paremmuudesta suhteessa kiinteähintaiseen sopimukseen ja kiinteähintaisen sopimuksen vaatiman tarkan etukäteismäärittelyn ongelmista ketterän kehittämisen yhteydessä.

Griffinin (2001) tutkimus käsittelee yhden yrityksen pitkän aikavälin kokemuksia XP-menetelmän käytöstä. Tärkeimmät asiakkaan näkökulmaa koskevat tutkimuksen tulokset ovat koulutuksen ja käytetyn ketterän kehittämisen menetelmän ymmärtämisen tarve kaikille kehitykseen osallistuville, liiketoiminnallisten päätösten tekemisen ja asiakkaan päätösten noudattamisen tärkeys sekä asiakkaan ja kehittäjien välisen suhteen tärkeys (Griffin, 2001).

Ketterästä kehittämisestä luodun yleiskatsauksen (alaluvut 2.1-2.4), asiakasta koskevan teoreettisen tutkimuksen (Lohan ym., 2010) sekä empiirisen tutkimuksen perusteella voidaan nyt muodostaa asiakkaan näkökulman määrittelmä, joka tulee toimimaan teoreettisena viitekehyksenä ja rajauksena koko muun tutkielman aihepiirille. Asiakkaan näkökulman tulee ensinnäkin tulla esille jo sopimuksetekovaiheessa (esim. Martin ym., 2004). Toiseksi asiakas tarkastelee kehittämistä liiketaloudellisesta näkökulmasta (esim. Poppendieck & Poppendieck, 2003), jolle on tyypillistä vaatimusten priorisointi arvon perusteella (esim. Agile Manifesto, 2001). Kolmanneksi ketterä kehittäminen tarjoaa asiakkaalle mahdollisuuden vaikuttaa kehittämisprojektin menestykseen osallistumalla siihen itse kehittämisen aikana (esim. Lohan ym., 2010; Kautz, 2009; Koskela & Abrahamsson, 2004). Näin ollen tulosvastuun ja päätäntävällän omaavan *asiakkaan näkökulmasta* ketterässä kehittämisessä voidaan katsoa olevan pääasiallisesti kolme kiinnostavaa seikkaa (kuvio 5):

- Millainen sopimus hänen kannattaa tehdä?
- Millaista arvoa kehittäminen tuottaa?
- Miten hänen kannattaa osallistua kehittämiseen voidakseen taata sen onnistumisen?



KUVIO 5 Asiakkaan näkökulma ketterään kehittämiseen

Nämä kolme asiakkaan näkökulman osaa muodostavat kukin oman osittaisen näkökulmansa ketterään kehittämiseen. Tutkielman seuraavissa osissa (luvut 3, 4 ja 5) käsitellään ketterää kehittämistä yksityiskohtaisesti kustakin näistä kolmesta näkökulmasta. Luvussa 6 kootaan tulokset yhteen muodostaen kokonais kuvan ketterästä kehittämisestä asiakkaan näkökulmasta.

2.6 Yhteenveto ketterästä ohjelmistokehityksestä

Ketterän ohjelmistokehityksen käsite pohjautuu vuonna 2001 julkaistuu Agile-manifestiin (Agile Manifesto, 2001). Se merkitsee keveiden, mukautuvien ja yksinkertaisten kehitysmenetelmien käyttöä iteratiivisesti ja inkrementaalisesti suoritettavassa ohjelmistokehityksessä, jossa korostetaan ihmiskeskeisyyttä ja yhteistyötä, tavoitteena tuottaa asiakkaalle arvoa nopeasti ja jatkuvasti sekä hyväksyen muutokset. Ketterän kehittämisen menetelmistä yleisimmin käytettyjä ovat Scrum ja Extreme Programming (XP). Näiden lisäksi on yleisesti käytetty myös Lean-ajatteluun perustuvaa lähestymistapaa ohjelmistokehitykseen.

Läheinen yhteistyö asiakkaan kanssa on ketterän kehittämisen keskeisimpiä periaatteita. Asiakas tarkoittaa sitä tahoa, jolla on hallussaan kehittämiseen käytettävät varat ja valtuudet päättää hankinnoista. Asiakkaan tehtäviin kuuluu esimerkiksi käyttäjätarinoiden tuottaminen, prioriteeteista päättäminen, tuotosten hyväksyntä, loppukäyttäjän tarpeiden tunteminen, kommunikointi kehittäjiä kanssa esimerkiksi vaatimuksista, toiminnallisten testien tuottaminen ja niiden suorituksen valvominen, iteraatioiden ja julkaisujen suunnitteluun osallistuminen sekä yhteydenpito johtoon. Ketterää kehittämistä ei ole mahdollista suorittaa, ainakaan menestyksellisesti, ilman asiakkaan roolin täyttämistä. Huonot asiakkaat johtavat huonoihin järjestelmiin.

Luvussa määriteltiin asiakkaan näkökulma ketterään kehittämiseen perustuen ketterästä kehittämisestä luotuun yleiskatsaukseen, asiakasta koskevan teoreettiseen tutkimukseen (Lohan ym., 2010) sekä aiheesta tehtyyn empiriseen tutkimukseen). Asiakkaan näkökulma koostuu kolmesta osasta: sopimuksen tekemisestä, kehitettävän ohjelmiston arvon määrittämisestä ja osallistumisesta kehittämisen aikana.

3 SOPIMUKSEN TEKEMINEN

Tässä luvussa tarkastellaan ensimmäistä osaa asiakkaan näkökulmasta ketterään ohjelmistokehitykseen, eli sopimuksen tekemistä. Ensiksi tarkastellaan sopimusta yleisesti ja erityisesti ohjelmistokehityksessä sekä ketterän kehittämisen erityisvaatimuksia sopimuksille. Tämän jälkeen esitellään erilaisia ohjelmistokehityksen sopimusmalleja sekä käsitellään kirjallisuudesta löytyviä teoreettisia arvioita ja empiirisiä kokemuksia esitellyistä sopimusmalleista. Seuraavaksi määritellään sopimusten käsittelyn teoreettisena viitekehyksenä toimiva asiakkaan näkökulma ja arvioidaan eri sopimusmalleja tästä näkökulmasta. Lopuksi esitetään yhteenveto.

3.1 Taustaa

Agile-manifestin (Agile Manifesto, 2001) mukaan ketterässä kehittämisessä arvostetaan asiakasyhteistyötä enemmän kuin sopimusneuvotteluita. Lisäksi tärkeä osa ketterää kehittämistä on yksityiskohtien päättäminen kehittämisen kuluessa, tarkan alkuvaiheen suunnittelun sijasta. Ketterä kehittäminen tulisi siis rakentaa tarkkojen sopimusten sijasta yhteistyön ja molemminpuolisen luottamuksen sekä kehittämisen kuluessa tarkentuvan suunnittelun pohjalle.

Warstan (2001) mukaan ihanteellisessa maailmassa tilanne olisikin juuri tällainen, eivätkä sopimukset silloin olisi tarpeen. Todellisessa maailmassa sopimukset ovat kuitenkin korvaamattomia riskien hallitsemisessa (Warsta, 2001). Stevensin (2009a) mukaan sopimukset ovat välttämättömiä yritysten välisessä yhteistyössä myös ketterän kehittämisen tapauksessa.

Sopimus on osapuolten välinen oikeustoimi, jolla ne järjestelevät keskinäiset oikeussuhteensa (Hyvärinen, Hulkko & Ohvo, 2002). Osapuolilla tarkoitetaan asiakasta ja toimittajaa. Sopimus hallinnoi osapuolten oikeuksia ja velvollisuuksia, ja sen tavoitteena on määritellä riskin jakautuminen osapuolten kesken (Thorup & Jensen, 2009). Sopimus voidaan nähdä pelisääntöinä, jotka oikein toteutettuina parantavat molempien osapuolten mahdollisuutta onnistua. Huo-

nosti toteutettuna ne sen sijaan vaikeuttavat yhteistyötä ja hidastavat toimintaa. Usein sopimukset nähdään kilpailuna, jossa tavoitteena on asettaa toinen osapuoli huonompaan asemaan varsinkin mahdollisen epäonnistumisen tapauksessa. Suuret organisaatiot käyttävät vakiomuotoisia sopimusehtoja, jotka sopimuskumppanin tulee hyväksyä sellaisenaan. Sopimukset ovat myös arvoa tuottamattomia tuotoksia, joten niihin käytetty aika tulisi minimoida. (Stevens, 2009a.)

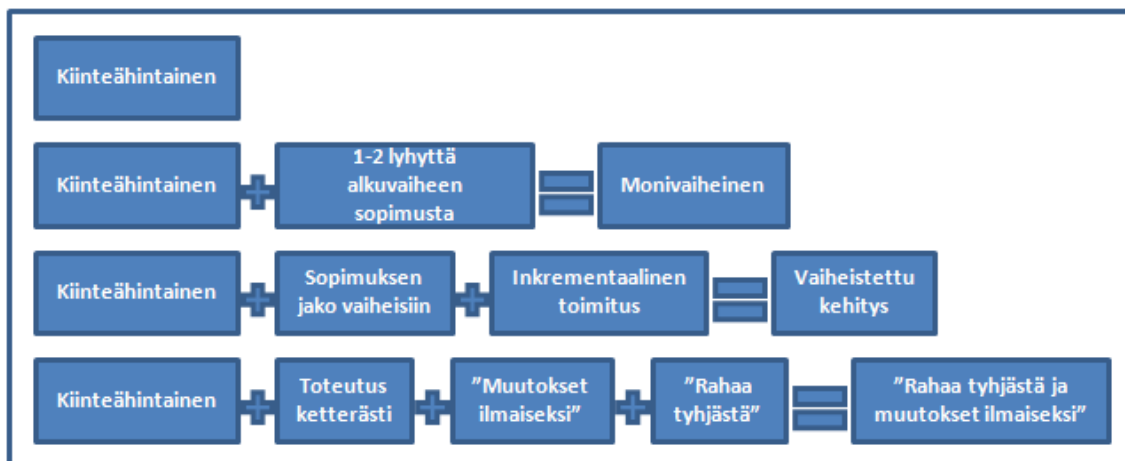
Ohjelmistokehityksen alalla perinteisen kehittämisen yhteydessä vakiintuneiden sopimusmallien soveltaminen ketterään kehittämiseen voi osoittautua ongelmalliseksi. Ketterän kehittämisen kohdalla pyritäänkin kehittämään sopimusmalleja, jotka ottavat huomioon kirjallisen sopimuksen tarpeen eri osapuolten välille, sisältämättä kuitenkaan perinteisten sopimusmallien huonoimpia puolia kuten kilpailuasetelmaa, joustamattomuutta ja monimutkaisten sopimusmallien laatimisen, neuvottelun ja hallinnoinnin vaatimaa arvoa tuottamatonta työtä.

3.2 Sopimusmalleja

Tässä alaluvussa esitellään yleisesti tunnettuja sopimusmalleja. Sopimusmallit on jaettu kolmeen ryhmään niiden perusrakenteen mukaisesti. Ensimmäinen ryhmä käsittää kiinteähintaisen sopimusmallin ja siitä johdetut sopimusmallit. Toinen ryhmä käsittää aika-ja-materiaalit -sopimusmallin ja sen johdannaiset. Kolmannen ryhmän muodostavat hybridi-sopimusmallit, joissa on yhdistelty kiinteähintaisen sopimusmallin sekä aika-ja-materiaalit -sopimusmallin ominaisuuksia.

3.2.1 Kiinteähintainen sopimusmalli ja sen johdannaiset

Kiinteähintaisen sopimusmallin ja sen johdannaisten rakenteita on havainnollistettu kuviossa 6. *Kiinteähintainen* (fixed price) sopimusmalli perustuu tarkasti ja kiinteästi etukäteismääriteltyyn hintaan, aikatauluun ja sisältöön (Van Cauwanberghe, 2003a). Siinä edellytetään, että toimittaja tekee sovitun työn sovitus- aikataulussa ja asiakas maksaa sovitun hinnan. Muutoksia sisältöön, hintaan tai aikatauluun ei hyväksytä suoraan, vaan jokainen muutos vaatii tarkasti määritellyn ja työlään muutostenhallintaprosessin läpiviennin. (Stevens, 2009b.) Kiinteähintainen on yleisimmin käytetty sopimusmalli, ja esimerkiksi julkisten organisaatioiden hankintoja määrittelevät lait vaativat usein nimenomaan kiinteähintaisen sopimusmallin käyttämistä ja halvimman tarjouksen valitsemista (Poppendieck & Poppendieck, 2003, s.158).



KUVIO 6 Kiinteähintainen sopimusmalli ja sen johdannaiset

Monivaiheinen (multi-stage) sopimusmalli perustuu kiinteähintaisen sopimusmallin pohjalle, mutta sen tavoitteena on käsitellä paremmin toimittajan vastuulle tulevaa riskiä. Monivaiheisia sopimusmalleja on kahdenlaisia: suurempaan kiinteähintaiseen sopimukseen tähtääviä ja pysyvästi monivaiheisia sopimuksia. Suurempaan kiinteähintaiseen sopimukseen tähtäävä monivaiheinen sopimus voi pitää sisällään yksi tai kaksi lyhyttä sopimusta tiedon hankkimiseksi, riskin vähentämiseksi ja siten kiinteähintaisen sopimuksen mahdollistamiseksi. (Poppendieck & Poppendieck, 2003, s.161.)

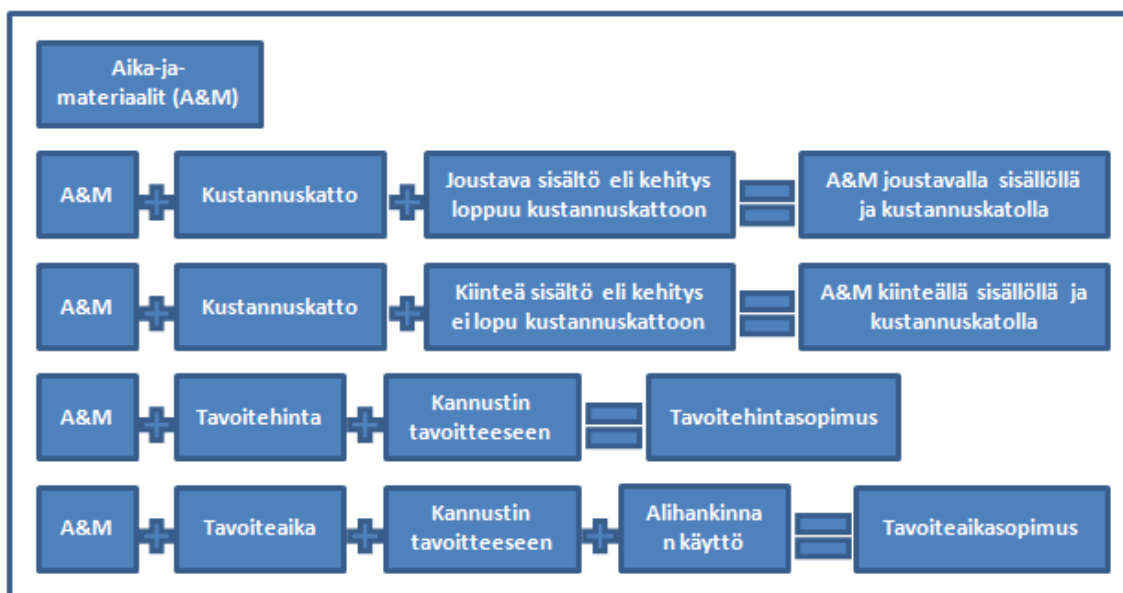
Vaiheistettu kehitys (phased development) -sopimusmalli vastaa jatkuvasti monivaiheista sopimusmallia (Stevens, 2009b; Poppendieck & Poppendieck, 2003, s.161). Tässä kehittämisprojekti jaetaan ajallisesti rajoitettuihin osavaiheisiin, joista jokainen tuottaa uuden julkaisun ohjelmistosta. Stevens (2009b) korostaa asiakkaan rahoittavan kerrallaan vain yhden osavaiheen, jonka päätyttyä onnistuneesti sovitaan mahdollisen seuraavan vaiheen rahoituksesta, ja asiakkaan riski rajoittuu siis vain yhden osavaiheen rahoitukseen kerrallaan. Asiakkaan ja toimittajan suhde on yhteistoimintaan perustuva, ja molemmilla osapuolilla on kannustin saada jokainen osavaihe suoritettua onnistuneesti, jotta myös seuraavat osavaiheet toteutuvat. Kunkin osavaiheen sisältö voi muuttua aikataulussa pysymiseksi, ja kynnys sisällön siirtämisestä seuraavaan osavaiheeseen ei yleensä ole kovin suuri. (Stevens, 2009b.)

"Rahaa tyhjästä ja muutokset ilmaiseksi" (money for nothing, change for free) -sopimusmalli perustuu myös kiinteähintaisen sopimusmallin pohjalle. Lisäksi sen perustana on ehdoton vaatimus suorittaa kehittäminen ketterästi ja myös asiakkaan täytyy osallistua täysipainoisesti ketterään kehittämiseen. Asiakas priorisoi ominaisuudet arvon mukaan, ja ne toteutetaan priorisoidussa järjestyksessä. Tässä sopimusmallissa kiinteähintaisen sopimusmallin päälle on lisätty *muutokset ilmaiseksi* -ehto, eli kiinteästi määritellystä sisällöstä huolimatta yksittäisiä ominaisuuksia voidaan vaihtaa toiseen työmäärältään vastaavaan kokoiseen ominaisuuteen. (Sutherland, 2008.) Stevensin (2009b) mukaan ylimääräiset, kokonaissisältöä kasvattavat ominaisuudet sen sijaan nostavat hintaa. Lisäksi tähän sopimusmalliin kuuluu *rahaa tyhjästä* -ehto, jonka mukaan asiakas päättää

kehittämisprojektin, kun prioriteetin mukaan seuraavaksi toteutettavan ominaisuuden tuoma lisäarvo olisi sen toteutuksen kustannuksia pienempi. Tällöin toimittaja saa kehittämisprojektin päättämisestä korvauksena 20 % jäljellä olevan sopimuksen arvosta eli "rahaa tyhjästä". (Sutherland, 2008.) Stevensin (2009b) mukaan riski on tässä sopimusmallissa jaettu osapuolten kesken ja molemmilla osapuolilla on kannustin pyrkiä nopeaan valmistumiseen, koska se merkitsee asiakkaalle alhaisempia kustannuksia ja toimittajalle korkeampaa voittomarginaalia.

3.2.2 Aika-ja-materiaalit -sopimusmalli ja sen johdannaiset

Aika-ja-materiaalit -sopimusmallin ja sen johdannaisten rakenteita on havainnollistettu kuviossa 7. *Aika-ja-materiaalit* (time and materials) -sopimusmalli perustuu toimituksen hinnoitteluun toimittajan toteutuneiden kustannusten perusteella. Toimittaja työskentelee esimerkiksi kuukauden ajan ja laskuttaa asiakasta toteutuneista kustannuksista. Toteutuksen sisältöä, hintaa tai aikataulua ei tarvitse sopia alussa tarkasti vaan niitä voidaan muuttaa toteutuksen kuluessa. Toimitus katsotaan valmiiksi, kun asiakas ei ole enää halukas maksamaan jatkokehityksestä. (Stevens, 2009b.)



KUVIO 7 Aika-ja-materiaalit -sopimusmalli ja sen johdannaiset

Poppendieckin ja Poppendieckin (2003, s.159) mukaan aika-ja-materiaalit -sopimusmalli on suunniteltu toimimaan epävarmuuden ja monimutkaisuuden kanssa, mutta se ei kuitenkaan vähennä kokonaisriskiä vaan ainoastaan siirtää sen toimittajalta asiakkaalle. Stevensin (2009b) mukaan aika-ja-materiaalit -sopimusmalliin yhdistetään usein kustannuskatto (cost ceiling) rajoittamaan asiakkaan taloudellista riskiä. Kustannuskaton mukaan asiakkaan maksettavaksi voi tulla enintään määritellyn suuruinen enimmäishinta. Aika-ja-materiaalit -

sopimusmalli kustannuskatolla voi olla sisällöltään joko muuttuva tai kiinteä. (Stevens, 2009b.)

Aika-ja-materiaalit joustavalla sisällöllä ja kustannuskatolla (time and materials with variable scope and cost ceiling) -sopimusmallissa määritellään kustannuskatto ja kehitystyö lopetetaan kustannusten saavuttaessa määritellyn kustannuskaton. Näin ollen osa asiakkaan toivomista vaatimuksista saattaa jäädä toteuttamatta. (Stevens, 2009b.)

Aika-ja-materiaalit kiinteällä sisällöllä ja kustannuskatolla (time and materials with fixed scope and cost ceiling) -sopimusmallissa määritellään myös kustannuskatto, mutta sisältö on silti kiinteä ja toimittaja on velvollinen kehittämään ohjelmiston valmiiksi omalla kustannuksellaan. Tämä sopimusmalli eroaa kiinteähintaisesta vain siten, että kustannussäästön suunniteltua nopeammasta valmistumisesta saa toimittajan sijaan asiakas. Aika-ja-materiaalit kiinteällä sisällöllä ja kustannuskatolla -sopimusmalli onkin asiakkaan näkökannalta hyvin matalariskinen. (Stevens, 2009b.)

Tavoitehintasopimus (target-cost contract) perustuu aika-ja-materiaalit -sopimusmallin pohjalle, mutta siihen on lisäksi määritelty etukäteen arvioitava tavoitehinta. Tämä sopimusmalli antaa kuitenkin myös mahdollisuuden poiketa sovitusta tavoitehinnasta tavalla, joka jakaa siitä aiheutuvan voiton tai tappion. Tavoitehinnan ylittyessä molemmat osapuolet maksavat enemmän, kun taas tavoitehinnan alittuessa molemmat osapuolet hyötyvät säästöstä. Tavoitehintasopimus solmitaan aluksi korkeamman tason suunnitelmana ja tietoisena yksityiskohtien tarkentumisesta vasta työn edistyessä. Sopimusmallin tärkein osa on tavoitehinnasta sopiminen, ja molempien osapuolten tulee sitoutua siihen. (Poppendieck & Poppendieck, 2003, s.164.)

Tavoitehintasopimuksessa on tärkeätä luoda sekä asiakkaalle että toimittajalle riittävät kannustimet tavoitehintaan pyrkimiselle, eli asiakkaan tapauksessa pitää vaatimukset kohtuullisina ja toimittajalle puolestaan alittaa kustannukset suhteessa tavoitehintaan. Asiakkaalle riittävä kannustin on yleensä tavoitehinnasta poikkeavien kulujen tai säästöjen jakaminen. Toimittajalle riittävä kannustin sen sijaan voidaan luoda kahdella tavalla. Näistä tavoista ensimmäinen on *kustannukset ja kiinteä palkkio* -malli (cost plus fixed fee). Siinä tavoitehintaa ei sisällä toimittajan voittoa vaan toimittaja saa erillisen palkkion työn suorittamisen jälkeen. Tavoitehinnan ylittyessä lisäkulut jaetaan ja toimittaja saa vähemmän voittoa. Tavoitehinnan alittuessa saatu säästö jaetaan ja toimittaja saa enemmän voittoa. Tavoitehinnan alittamisesta voidaan myös maksaa erillinen lisäpalkkio. Toinen mahdollisuus on *voitto, jos kustannuksia ei ylitetä* -malli (profit not to exceed). Siinä tavoitehintaa sisältää toimittajan voiton. Tavoitehinnan ylittävältä osalta toimittaja tekee työtä omakustannehintaan, kun taas tavoitehinnan alittuessa toimittaja saa ennalta sovitun lisäpalkkion. (Poppendieck & Poppendieck, 2003, s.164.)

Tavoiteaikasopimus (target-schedule contract) on tavoitehintasopimuksen kaltainen, mutta siinä sovitaan kustannusten sijasta tarkka aikataulu. Tätä sopimusmallia voidaan käyttää silloin, kun aikataulu on kustannuksia tärkeämpi. (Poppendieck & Poppendieck, 2003, s.166.) Tavoiteaikasopimus mahdollistaa työvoiman lisäämisen tai alihankinnan käyttämisen tarvittaessa tavoiteaikatauluun

pyrittäessä (Stevens, 2009b). Mikäli työntekijöiden määrä ei muutu eikä käytetä alihankintaa, tavoiteaikasopimus ja tavoitehintasopimus ovat itse asiassa täysin sama asia (Poppendieck & Poppendieck, 2003, s.166).

3.2.3 Hybridisopimusmallit

Kiinteähintaisten tai aika-ja-materiaalit -tyyppisten sopimusmallien sijasta voidaan käyttää näiden mallien piirteitä yhdisteleviä malleja, joita kutsutaan hybridisopimusmalleiksi. Tällaisia ovat yhteistyösopimus (Thorup & Jensen, 2009) ja adVantage-sopimusmalli (Book, Gruhn & Streamer, 2012).

Yhteistyösopimuksessa (collaborative contract) sekä riski että hyöty pyritään jakamaan tasaisesti asiakkaan ja toimittajan kesken. Tätä tavoitellaan jakamalla kehittämisprojektin hinta kahteen osaan, toimittajan kuluihin aika-ja-materiaalit -sopimusmallin mukaan sekä tämän päälle tulevaan kiinteään valmistumishintaan. Sekä kehittämisprojekti että valmistumishinta jaetaan toisiaan vastaaviin osiin ja kunkin osan valmistuminen aiheuttaa sille määritellyn valmistumishinnan lankeamisen maksettavaksi. Valmistumisen kriteerinä pidetään sitä, kun kehitetty ohjelmiston osa alkaa tuottaa asiakkaalle arvoa. Molemmilla osapuolilla on yleensä pyrkimys tähän tilanteeseen mahdollisimman nopeasti. Toimittajan tehokkuus ja luovuus palkitaan ja asiakas pyrkii mahdollisimman yksinkertaiseen arvoa tuottavaan ohjelmistoon. Yhteistyösopimukseen määritellään kehittämissuhteiden tavoite väljästi rajattuna, tuntihinta (10 - 50 % normaalia aika-ja-materiaalit -sopimusta alhaisempana), kehittämisprojektin jako osiin ja jokaiseen osaan liittyvä valmistumishinta, ketteriä periaatteita noudattava kehitysprosessi sekä alustava aikataulu koko kehittämisprojektille ja kullekin sen osalle. Yhteistyösopimuksesta sen sijaan nimenomaan puuttuvat ehdoton valmistumisen aikaraja, yksityiskohtaiset vaatimukset ja viivästyskorvaukset. Tuntihinnan ja valmistumishinnan välistä tasapainoa muuttamalla sopimusmallin ominaisuuksia voidaan säätää sen mukaan, halutaanko sopimuksesta enemmän kiinteähintaisen vai aika-ja-materiaalit -sopimusmallin kaltainen, ja sen myötä kehittämisprojektin luonnetta voidaan muuttaa nopeampaan valmistumiseen tai laajempaan sisältöön ja korkeampaan laatuun tähtääväksi. Yhteistyösopimus mahdollistaa erityisesti ketterän kehittämissuhteen ja yhteistyöhön perustuvan suhteen. (Thorup & Jensen, 2009.)

adVantage-sopimusmallin (Book ym., 2012) tavoitteena on saavuttaa alusta asti kehittämisprojektin sisällön korkean tason hahmotelma, kuitenkin altistamatta toimittajaa kiinteään sisältöön sitoutumisen riskille. Lisäksi sopimusmalli pyrkii varmistamaan, että asiakas maksaa vain todellisuudessa saamista ominaisuuksista sekä ehkäistä riskiä joutua toimittajan peräytymisen uhriksi. Sopimusmallin peruseriaatteet ovat riskin jakautuminen ja laskutusmallin tuomat kannustimet tehokkuuteen. Sopimusmallin mukaisessa kehittämisprojektissa toteutus tehdään vaiheittain sprinteissä. Ensimmäisessä vaiheessa kerätään asiakkaan vaatimukset käyttäjätarinoiden muodossa ja toimittaja tekee käyttäjätarinoille työmäärä- sekä kustannusarviot. Toisessa vaiheessa asiakas priorisoi käyttäjätarinat aiemmin tehtyjen arvioiden ja budjettinsa perusteella ja

osapuolet sopivat ensimmäisen sprintin sisällöstä. Kolmas vaihe on ensimmäisen sprintin sisällön toteutus ja neljäs vaihe sprintin toteutuksen käyttäjätarinakohtainen tarkistus ja laskutus. Kunkin sprintin laskutuksessa on kolme vaihtoehtoa. Mikäli valmistuneiden käyttäjätarinoiden toteutuksen yhteenlasketut kustannukset sprintin aikana ovat suunniteltua pienemmät, vain nuo toteutuneet kustannukset laskutetaan. Mikäli sen sijaan kustannukset ovat suunniteltua suuremmat, ylimääräiset kustannukset laskutetaan toimittajan omakustannehintaan. Sprintin sisältöön kuuluvien valmistumattomien käyttäjätarinoiden kohdalla asiakkaalla on puolestaan mahdollisuus poistaa kyseinen käyttäjätarina kokonaan tai siirtää se seuraavaan sprinttiin. Poistettavien käyttäjätarinoiden osalta toimittaja saa siihen asti toteutuneet kustannukset, kun taas siirrettävien käyttäjätarinoiden osalta kustannukset laskutetaan vasta myöhemmin niiden valmistuessa. Jokaisen sprintin jälkeen asiakkaalla on mahdollisuus joko aloittaa uuden sprintin suunnittelu tai lopettaa kehittäminen. (Book ym., 2012.)

3.3 Sopimusmallien arvioita kirjallisuudesta

Tässä aluvuossa esitellään kirjallisuudessa esitettyjä käsityksiä ja kokemuksia eri sopimusmallien sopivuudesta ketterään kehittämiseen. Ensin kerrotaan kiinteähintaisesta sopimusmallista ja sen johdannaisista, sen jälkeen aika-ja-materiaalit -sopimusmallista ja sen johdannaisista ja lopuksi hybridisopimusmalleista.

Poppendieckin ja Poppendieckin (2003, s.159) mukaan kiinteähintainen sopimusmalli pyrkii suojaamaan asiakasta toimittajan kustannuksella ja siirtää riskin toimituksen onnistumisesta toimittajalle. Toimittajalla onkin kiinteähintaisessa sopimusmallissa merkittävä riski toimituksen työmäärän arvioimisesta oikein ja sopimusmallin käyttö edellyttää toimittajan riskin sisällyttämistä hinnoitteluun, kuten jokainen asiansa osaava toimittaja tekee. Osaamattomat toimittajat sen sijaan ovat taipuvaisia antamaan liian optimistisia työmäärä- ja kustannusarvioita, jotka usein valitettavasti tulevat edullisuutensa takia valituksi. Epärealistisesta arviosta johtuva tarjouksen edullisuus kuitenkin kosta tuu toteutusvaiheessa. Niinpä voidaan sanoa ongelmille alttiimpien tarjousten tulevan yleensä valituiksi. (Poppendieck & Poppendieck, 2003, s.159.)

Van Cauwanberghen (2003a) mukaan kiinteähintainen sopimusmalli ei vastoin yleistä käsitystä suinkaan suojaa asiakasta riskeiltä. Sen sijaan niin toimittaja kuin asiakaskin hyvin todennäköisesti kokevat taloudellisia tappioita, jotka johtuvat yleensä vasta kehittämisprojektin loppuvaiheessa realisoituvista riskeistä ja joiden vuoksi alussa määritellyt vaatimukset jäävät useasti toteuttamatta sovitun aikataulun ja budjetin rajoissa. Van Cauwanberghe (2003a) esittää tutkimuksessaan useita ehdotuksia kiinteähintaisen sopimusmallin alaisuudessa tapahtuvan ketterän kehittämisen riskien pienentämiseksi niin toimittajan kuin asiakkaankin kannalta, mutta suosittelee yleisesti ottaen kiinteähintaisen sopimusmallin käyttämistä vain osapuolille tutuissa ympäristöissä ja ennalta turvallisiksi arvioitavissa kehittämisprojekteissa. Monimutkaisemmissa ja vai-

keammassa kehittämiprojekteissa kiinteähintaista sopimusmallia ei sen sijaan tulisi käyttää. (Van Cauwanberghe, 2003a.)

Van Cauwanberghe (2003b) luettelee kuitenkin myös asiakkaalle ainakin lyhyellä aikavälillä koituvia hyötyjä kiinteähintaisesta sopimusmallista. Kiinteä aikataulu mahdollistaa projektien tarkan aikataulutuksen. Kiinteä hinta mahdollistaa tarkan budjetoinnin ja tuloarvion tekemisen. Lisäksi asiakkaalla on mahdollisuus priorisoida vaatimuksensa, mikäli kehittämisen aikataulu tai budjetti uhkaa ylittyä. (Van Cauwanberghe, 2003b.)

Hodan, Noblen ja Marshallin (2009) mukaan suurin osa asiakkaista vaatii kiinteähintaisen sopimusmallin käyttämistä myös ketterän kehittämisen tapauksessa, koska he kokevat sen tuovan ennustettavuutta sekä kehittämiprojektin sisällön, aikataulun ja kustannusten hallintaa. Tämä aiheuttaa kuitenkin selvän ongelman asiakkaan tavoitteleman ennustettavuuden ja ketterän kehittämisen muutoksen hyväksymisen välillä. Tutkimuksen mukaan kehittämiprojektien toimittajat pyrkivätkin vaikuttamaan asiakkaisiinsa saadakseen heidät luopumaan kiinteähintaisen sopimusmallin suosimisesta. Äärimmäisenä esimerkkinä mainitaan jopa ketterien menetelmien käyttäminen vain sisäisesti ja tämän seikan piilottaminen asiakkaalta. (Hoda ym., 2009.) On vaikea nähdä tällaisessa toimintatavassa mitään rakentavaa kummankaan kannalta, joten asiakkaan olisi varmasti usein hyödyllisempää luopua kiinteähintaisen sopimusmallin käyttämisen vaatimuksesta.

Eckfeldtin, Maddenin ja Horowitzin (2005) empiiristen kokemusten mukaan kiinteähintainen sopimusmalli ei useimmiten toimi ketterässä kehittämisessä, koska siitä puuttuu mahdollisuus muutoksen hyväksymiseen ja joustavuuteen. Kiinteähintainen sopimusmalli on huono toimittajan kannalta sen sisältämän riskin vuoksi, ja huono myös asiakkaan kannalta tuon riskin vaatiman ylihinnon vuoksi. Asiakkaat myös häviävät sitoutuessaan liian aikaisin tiettyyn sisältöön ja joutuessaan myöhemmin tekemään tarvittavat muutokset muutostenhallintaprosessin kautta ja maksamaan siitä. Lisäksi kiinteähintainen sopimusmalli vahvistaa perinteistä toimittaja-asiakas-asetelmaa ja haittaa yhteistyön muodostumista. (Eckfeldt ym., 2005.)

Thorupin ja Jensenin (2009) empiirisen kokemuksen mukaan kiinteähintaisen sopimusmallin käyttäminen on johtanut matalaan luottamukseen osapuolten välillä ja syyttelevään käytökseen sekä muihin epäoptimaalisiin käyttäytymisen muotoihin, jotka ovat lopulta johtaneet huonoihin tuloksiin ja viivästyksiin.

Franklinin (2008) empiiriset tulokset eroavat merkittävästi muista tutkimuksista ja kokemuksista. Niiden mukaan ketterää kehittämistä voidaan menestyksekkäästi suorittaa kiinteähintaisen sopimusmallin alaisuudessa. Tähän kannattaa kuitenkin ryhtyä vain, mikäli asiakas ei hyväksy ketterään kehittämiseen optimaalisempaa aika-ja-materiaalit -sopimusmallia. Tärkeätä on sopimusneuvotteluvaiheessa ymmärtää, mitä kehittämiprojektista ei voida etukäteen tietää, sekä hyväksyä ja myöntää tämä. (Franklin, 2008.)

Poppendieckin ja Poppendieckin (2003, s.161) mukaan monivaiheisessa sopimusmallissa ennen kiinteähintaisen sopimuksen solmimista suoritettava

tiedonhankinta vähentää tiedonpuutteesta johtuvia ongelmia määrittelyssä. Toisaalta se voi nostaa kynnyistä hyväksyä muutoksia tehtyyn määrittelyyn, koska asiakas on jo maksanut toimittajalle sen tekemisestä hyvin. (Poppendieck & Poppendieck, 2003, s.161.)

Poppendieckin ja Poppendieckin (2003, s.161) mukaan vaiheistettu kehitys-sopimusmalli soveltuu hyvin iteratiiviseen ketterään kehittämiseen. Siinä on kuitenkin myös omat riskinsä, suurimpana molempien osapuolten useasti toistuvat mahdollisuudet hylätä koko yhteistyö. Vaiheistetun kehityksen käyttäminen luo bilateraalisen monopolin, jossa molemmat osapuolet ovat riippuvaisia toisistaan. (Poppendieck & Poppendieck, 2003, s.161.) Asiakkaan kannalta tämä riski on erityisen merkittävä, koska toimittajan irtaantuminen ennen aikaisesti voi aiheuttaa asiakkaalle kalliin ohjelmiston jäämisen puolivalmiiksi ja käyttökelvottomaksi. Poppendieckin ja Poppendieckin (2003, s.162) mukaan sopimusmallin kustannukset nousevat lisäksi merkittäviksi, mikäli sopimus täytyy neuvotella uusiksi jokaisessa vaiheessa ja yleensä niitä hallinnoidaankin laajemmalla, alkuvaiheessa neuvoteltavalla pääsopimuksella. (Poppendieck & Poppendieck, 2003, s.162.)

Stevensin (2009b) mukaan aika-ja-materiaalit -sopimusmalli on toimittajan edun mukainen, koska asiakas vastaa rahallisesti toimituksen ongelmista ja muutoksista. Poppendieckin ja Poppendieckin (2003, s.159) mukaan riskinä on toisaalta toimittajan kannalta työn jatkuvuuden epävarmuus. Ongelmaksi molemmille osapuolille voi muodostua sopimusmallista puuttuva kannustin toimittajalle toimia tehokkaasti ja alhaisilla kustannuksilla (Stevens, 2009b). Lisäksi sopimusmallissa on ongelmana merkittävät sopimuksen laskutuksen valvomiseen liittyvät kustannukset (Poppendieck & Poppendieck, 2003, s.160).

Thorupin ja Jensenin (2009) mukaan aika-ja-materiaalit -sopimusmallin ongelmia ovat asiakkaalta puuttuva mahdollisuus valvoa toimittajan työskentelyn tehokkuutta, toimittajalta puuttuva kannustin toimia tehokkaasti sekä epä-tietoisuus kehittämissuunnitelman kokonaiskustannuksista ja siten asiakkaan budjetoinnin mahdottomuus. (Thorup & Jensen, 2009.)

Franklinin (2008) empiiristen kokemusten mukaan aika-ja-materiaalit -sopimusmalli antaa asiakkaalle joustavuutta kehittämisen sisällön ja prioriteettien hallintaan iteraatiokohtaisesti suunnittelun tarkentuessa ja onkin ideaalinen sopimusmalli ohjelmistokehitykseen. Tutkimuksen perusteella aika-ja-materiaalit -sopimusmallin yhteydessä saavutettiin merkittäviä tuloksia sisällön suhteen, mutta asiakkaan odotukset kustannusten ja aikataulun osalta ylittyivät. Tämä ei kuitenkaan ole mahdollinen malli kaikille asiakkaille, koska esimerkiksi joidenkin julkishallinnon organisaatioiden säännöt vaativat tarkasti etukäteen määriteltäviä kehittämissuunnitelmia. (Franklin, 2008.)

Eckfeldtin ym. (2005) empiiristen kokemusten mukaan aika-ja-materiaalit -sopimusmalli sisältää pienimmän riskin toimittajalle, mutta sen myyminen on hankalaa. Monet asiakkaat pelkäävät sopimusmallin aiheuttamaa sisällön avoimuutta ja kustannusten ennakoimattomuutta. Asiakkaat pitävät aika-ja-materiaalit -sopimusmallia turvallisena vain pieniin kehittämissuunnitelmiin tai ennalta tutun ja hyväksi havaitun toimittajan kanssa työskentelyyn. Lisäksi ai-

ka- ja -materiaalit -sopimusmalli vahvistaa perinteistä toimittaja-asiakasasetelmaa ja haittaa yhteistyön muodostumista. (Eckfeldt ym., 2005.)

Eckfeldtin ym. (2005) empiiristen kokemusten mukaan tavoitehintasopimuksen suurin hyöty on sen kannustama asiakassuhde. Sopimusmalli tarjoaa toimittajalle mahdollisuuden jakaa riski asiakkaan kanssa ja asiakkaalle mahdollisuuden työskennellä yhteistyössä toimittajan kanssa koko kehittämissuorituksen hyväksi ja nämä seikat edesauttavat hyvää ja rakentavaa asiakassuhdetta. Asiakkaan kantaessa osan riskistä, asiakkaalla on myös kannustin pyrkiä tehostamaan omia työtapojaan ja yhteistyötä toimittajan kanssa sekä karsia turhaa monimutkaisuutta. Riskin jakautuminen poistaa yksiselitteisen voittaja-häviöasetelman ja sen myötä koko kehittämissuoritus hyötyy. Toisaalta taas muutostenhallinta on lähes yhtä hankalaa kuin kiinteähintaisessa sopimusmallissa. Eckfeldtin ym. (2005) mukaan tämä johtuu tarpeesta päätellä jokaisen muutoksen kohdalla, onko kyseessä sovittuun sisältöön kuuluva vai ylimääräinen ominaisuus, ja jälkimmäisessä tapauksessa lisäksi sen kustannusvaikutusten laskeamisen vaikeudesta. Kokemusten mukaan sopimusmalli on toiminut erinomaisen hyvin ja sen sanotaan myös kannustavan tehokkaaseen ja joustavaan ohjelmistokehitykseen. (Eckfeldt ym., 2005.)

Franklinin (2008) empiiristen kokemusten mukaan hybridisopimusmallissa on tärkeitä priorisoida vaatimukset ja keskittyä olennaiseen. Aika- ja materiaalit -sopimusmallia kiinteämmäksi määriteltujen sisällön, aikataulun ja kustannusten joustamattomuus rajoittaa kehittäjien vapauksia suunnittelun optimoinnin suhteen. Asiakkaalta puolestaan vaaditaan vaatimusten priorisointia ja olennaiseen keskittymistä aikataulun ja kustannusten sisällä pysymiseksi. (Franklin, 2008.)

Thorupin ja Jensenin (2009) empiiriset kokemukset yhteistyösopimuksesta ovat erittäin hyviä. Asiakkaiden kerrotaan hyötyvän suuresti mahdollisuudesta muuttaa kehittämissuorituksen suunniteltua sisältöä lisäämällä uusia tärkeitä havaittuja ominaisuuksia vähemmän arvoa tuottaviksi paljastuneiden ominaisuuksien tilalle. Kokemusten mukaan asiakkaan kehitetystä ohjelmistosta sama arvo vastaa hyvin pitkälti toimittajan voittoa. Tutkimuksessa viitataan myös Fehrin, Kleinin ja Schmidtin (2004) tutkimukseen, jonka mukaan riskin ja hyödyn jakautuminen tasapuolisesti sopimuksissa saa todennäköisimmin aikaan rakentavan ja yhteistyöhön perustuvan asiakassuhteen. (Thorup & Jensen, 2009.)

Yhteenvedo edellä käsitellyistä teoreettisista tutkimuksista sekä empiirisisistä kokemuksista on esitetty taulukossa 4, jossa kunkin tutkimuksen kohdalla on mainittu sen käsittelemä sopimusmalli, tutkimuksen tyyppi ja tutkimuksen merkittävimmät havainnot asiakkaan kannalta.

TAULUKKO 4 Yhteenveto sopimusmallien arvioista

Tutkimus	Sopimusmalli	Tutkimuksen tyyppi	Merkittävimmät havainnot asiakkaan kannalta
Poppendieck & Poppendieck, 2003	Kiinteähintainen	Teoreettinen	-ongelmille altteimmat tarjoukset tulevat yleensä valituiksi
Van Cauwanberghe, 2003a	Kiinteähintainen	Teoreettinen	-ei vastoin yleistä käsitystä suojaa asiakasta -tulisi käyttää vain tutuissa ympäristöissä ja turvallisissa kehittämissuunnitelmissa
Van Cauwanberghe, 2003b	Kiinteähintainen	Teoreettinen	+mahdollistaa tarkan aikataulutuksen ja budjetoinnin
Hoda ym., 2009	Kiinteähintainen	Teoreettinen	-ennustettavuuden tavoittelemisen on ristiriidassa muutoksen hyväksymisen kanssa -toimittajat pyrkivät saamaan asiakkaat luopumaan sopimusmallin suosimisesta
Eckfeldt ym., 2005	Kiinteähintainen	Empiirinen	-toimittajan riskin vaatima ylihinnoittelu -sitoutuminen aikaisin tiettyyn sisältöön -vahvistaa vastakkainasettelua ja haittaa yhteistyötä
Thorup & Jensen, 2009	Kiinteähintainen	Empiirinen	-matala luottamus osapuolten välillä -johtaa huonoihin tuloksiin ja viivästyksiin
Franklin, 2008	Kiinteähintainen	Empiirinen	+soveltuu ketterään kehittämiseen
Poppendieck & Poppendieck, 2003	Monivaiheinen	Teoreettinen	+vähentää tiedonpuutteesta johtuvia ongelmia määrittelyssä -voi nostaa kynnyksen hyväksyä muutoksia
Poppendieck & Poppendieck, 2003	Vaiheistettu kehitys	Teoreettinen	+soveltuu hyvin ketterään kehittämiseen - toistuvat mahdollisuudet hylätä yhteistyö -kustannukset nousevat merkittäviksi, mikäli neuvotellaan uusiksi jokaisessa vaiheessa
Stevens, 2009b	Aika-jamateriaalit	Teoreettinen	-asiakas vastaa ongelmista ja muutoksista -toimittajalta puuttuu kannustin tehokkuuteen
Poppendieck & Poppendieck, 2003	Aika-jamateriaalit	Teoreettinen	- laskutuksen valvomiseen liittyvät kustannukset
Thorup & Jensen, 2009	Aika-jamateriaalit	Empiirinen	-toimittajan tehokkuuden valvominen vaikeata -toimittajalta puuttuu kannustin toimia tehokkaasti -epätietoisuus kokonaiskustannuksista
Eckfeldt ym., 2005	Aika-jamateriaalit	Empiirinen	-sisällön ja kustannusten avoimuus -turvallinen vain pieniin kehittämissuunnitelmiin tai ennalta tutun toimittajan kanssa työskentelyyn -vahvistaa vastakkainasettelua ja haittaa yhteistyötä

(jatkuu)

TAULUKKO 4 Yhteenveto sopimusmallien arvioista (jatkuu)

Tutkimus	Sopimusmalli	Tutkimuksen tyyppi	Merkittävimmät havainnot asiakkaan kannalta
Eckfeldt ym., 2005	Tavoitehintasopimus	Empiirinen	+sopimusmallin kannustama asiakassuhde +toiminut erinomaisen hyvin, tehokas ja joustava -muutostenhallinta hankalaa
Franklin, 2008	Hybridimallit	Empiirinen	-joustamattomuus rajoittaa kehittäjien vapauksia
Thorup & Jensen, 2009	Yhteistyösopimus	Empiirinen	+kokemukset erittäin hyviä +”paras yhteistyön ilmapiiri 20 vuoteen” +asiakkailta mahdollisuus muuttaa sisältöä

3.4 Sopimusmallit asiakkaan näkökulmasta

Tässä aluvussa arvioidaan aiemmin esiteltyjä sopimusmalleja asiakkaan näkökulmasta. Aluksi esitellään sopimusmallien arvioinnin kriteerit ja niiden muodostama asiakkaan näkökulma. Sen jälkeen käytetään esiteltyjä kriteerejä arvioitaessa sopimusten soveltuvuutta ketterään kehittämiseen ja edullisuutta asiakkaalle.

3.4.1 Sopimusmallien arvioinnin kriteerit

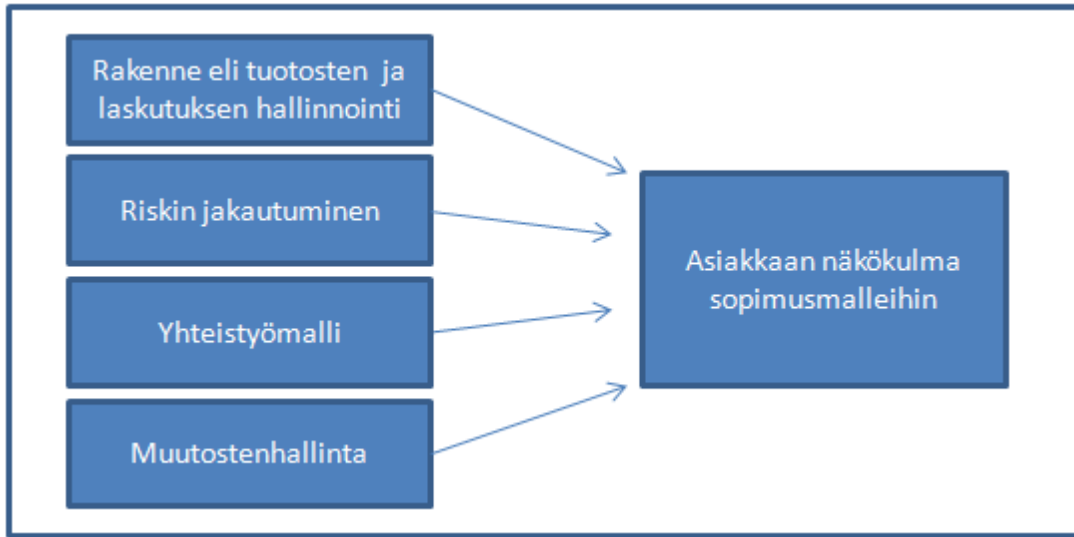
Stevens (2009a) esittelee keinoja sopimusmallien arviointiin ketterässä kehittämisessä ja neuvoo kiinnittämään huomion neljään seikkaan:

- rakenne eli tuotosten ja laskutuksen hallinnointi
- riskin jakautuminen asiakkaan ja toimittajan välillä
- muutostenhallinta
- yhteistyömalli (kilpailullinen, yhteistyöhön perustuva, välinpitämätön vai riippuvainen)

Edellä mainitut neljä seikkaa toimivat kriteereinä sopimusmallien arviointiin asiakkaan näkökulmasta. Näiden kriteereiden muodostama asiakkaan näkökulma on esitetty kuviossa 8, ja se toimii sopimusten vertailun teoreettisena viitekehystenä.

Sopimuksen rakenne käsittää tuotosten ja laskutuksen hallinnoinnin mekanismit. Kehittämiprojektin sopimuksen perustavin seikka on juuri kysymys siitä miten ja millä perusteilla toimitus ja maksu suoritetaan. Toimitus voidaan suorittaa lopuksi kerralla tai esimerkiksi inkrementti kerrallaan. Maksun suuruus voidaan määritellä etukäteen sovitun perusteella, toimittajan toteutuneiden kulujen perusteella tai näitä kahta perusmallia eri tavoin muunnellen. Maksun suoritus puolestaan voi tapahtua kerralla heti kehittämisprojektin alussa tai vasta lopussa toimituksen yhteydessä tai osina esimerkiksi jokaisen inkrementin toimituksen yhteydessä. Maksun suoritukseen voi myös liittyä kus-

tannuskatto tai erilaisia lisäpalkkioita tai sopimussakkoja, mutta nämä kuuluvat pikemminkin riskin jakautumisen alle.



KUVIO 8 Asiakkaan näkökulma sopimusmalleihin

Asiakkaan näkökulmasta sopimuksen rakenteen tulisi mahdollistaa mahdollisimman pikaisesti ja mahdollisimman paljon arvoa tuottavan ohjelmiston saamisen käyttöön ja toisaalta sitoa mahdollisimman vähän yrityksen varoja etukäteen sekä olla kustannuksiltaan mahdollisimman ennustettava. Huono rakenne asiakkaan kannalta olisi esimerkiksi toimitus kerralla lopuksi ja koko kiinteän hinnan maksu etukäteen, tai vaihtoehtoisesti täysin avoin kustannusten laskutusoikeus toimittajalle. Hyvä rakenne puolestaan olisi esimerkiksi ohjelmiston toimitus inkrementteittäin kehittämisen edistyessä ja etukäteen sovitun suuruisen maksu kerralla lopuksi tai toimittajan kustannusten maksaminen inkrementteittäin toimitusten yhteydessä.

Riski jakautuu eri sopimusmalleissa hyvin vaihtelevasti ja monissa sopimusmalleissa hyvin epätasaisesti. Beaumont (2008) esittää esimerkkinä riskin jakautumisen perinteisten kiinteähintaisen sekä aika-ja-materiaalit -sopimuksen yhteydessä (kuvio 9). Kiinteähintaisessa sopimuksessa toimittaja kantaa kaiken riskin, kun taas aika-ja-materiaalit -sopimuksessa koko riski on asiakkaan vastuulla.



KUVIO 9 Riskin jakautuminen (Beaumont, 2008)

Asiakkaan näkökulmasta riskin jakautumisen suhteen tärkeintä on minimoida oma riski ilman kehittämisprojektin kokonaisriskin kasvamista. Koko riskin kaataminen toimittajalle saattaisi vaikuttaa hyvältä ratkaisulta, mutta tosiasias-
sa se saattaa toteutuessaan aiheuttaa kehittämisprojektin epäonnistumisen toi-
mittajan tullessa kykenemättömäksi jatkamaan kehitystyötä. Optimaalisin rat-
kaisu löytyy käytännössä useimmiten tasapuolisesti jaetusta riskistä, jolloin mo-
lemmilla osapuolilla on kannustin työskennellä kehittämisprojektin kokonaise-
dun hyväksi.

Muutos on keskeinen piirre ketterässä kehittämisessä, ja *muutostenhallinta* on usein hyvin keskeinen ongelma ketterän kehittämisen sopimuksissa. Perin-
teiset sopimusmallit pyrkivät kuitenkin estämään muutoksen. Ne käsittelevät
muutoksia usein hyvinkin raskaan muutostenhallintaprosessin keinoin. Ketterä
kehittäminen sen sijaan hyväksyy muutoksen ja ketterän kehittämisen sopimus-
ten tulisi siksi myös hyväksyä muutos oleellisena osana kehittämisprojektia ja
sisältää sisäänrakennettu mahdollisuus muutosten tekemiseen ilman raskasta
muutostenhallintaprosessia.

Asiakkaan näkökulmasta muutosten tekemisen sisältöön tulisi olla yksin-
kertaisesti toteutettavissa. Sisällön kiinteä lukitseminen varhaisessa vaiheessa ei
palvele ketterän kehittämisen periaatteita. Muutosten tekeminen mahdollistaa
asiakkaan tarpeita paremmin vastaavan ohjelmiston tuottamisen, kun vaati-
muksia voidaan tarkentaa tiedon lisääntyessä kehittämisen aikana ja asiakkaan
tarpeiden mahdollisesti muuttuessa.

Stevensin (2009a) mukaan sopimus voi kannustaa toista tai molempia so-
pimusosapuolia joko kilpailulliseen, yhteistyöhön perustuvaan, välinpitämät-
tömmään tai riippuvaiseen *yhteistyömalliin*.

Kilpailullisessa yhteistyömallissa (competitive) yhteistyö perustuu kilpailu-
asetelmaan, jossa molemmat osapuolet pyrkivät hyötymään toisesta ja toisen
osapuolen voitto merkitsee toisen tappiota (my win is your loss) (Stevens,
2009a). Tämä saattaa ilmetä esimerkiksi riskin siirtämisenä mahdollisimman
paljon tai jopa kokonaan toiselle osapuolelle tai omalle osalle jäävän riskin yli-
hinnoittelussa.

Yhteistyöhön perustuva yhteistyömallissa (win-win) yhteistyö perustuu
molempinpuolisen hyödyn ja koko kehittämisprojektin menestyksen tavoitte-
luun (Stevens, 2009a). Tällöin kummallakaan osapuolella ei ole tavoitteena hyö-
tyä toisen ja mahdollisesti koko kehittämisprojektin kustannuksella.

Välinpitämättömässä yhteistyömallissa (indifferent) yhteistyö perustuu myös
oman edun tavoittelulle, mutta eri syistä kuin kilpailullisessa yhteistyömallissa.
Välinpitämättömässä yhteistyömallissa toinen osapuoli suhtautuu kehittämi-
seen välinpitämättömästi ja siten hankaloittaa sitä (I don't care - you lose)
(Stevens, 2009a). Välinpitämättömyys johtuu kehittämisprojektin merkitykset-
tömyydestä kyseiselle osapuolelle. Tarkoituksena ei ole hyötyä toisesta osapu-
olesta, mutta ei myöskään käyttää omaa panosta vähämerkityksiseen työhön.

Riippuvainen yhteistyömalli (dependent) perustuu toisen osapuolen riippu-
vaisuuteen toisesta. Molemmat osapuolet tavoittelevat omaa etuaan, mutta sen
saavuttaakseen toinen osapuoli tarvitsee välttämättä myös toista. Toisesta riip-

puvainen osapuoli kantaa automaattisesti suuremman riskin kuin toinen (heads-I-win-tails-you-lose) (Stevens, 2009a).

Asiakkaan näkökulmasta sopimuksen yhteistyömallin tulisi olla yhteistyöhön perustuva. Voisi näyttää houkuttelevalta tähdätä myös joko kilpailulliseen tai riippuvaiseen yhteistyömalliin asiakkaan ollessa hallitsevana osapuolena, mutta tällöin on olemassa riski kehittämisprojektin kärsimisestä. Tuollaista riskiä tuskin on kannattavaa ottaa, kun tavoitteena on uuden arvoa tuottavan ohjelmiston kehittäminen, jonka epäonnistumisen aiheuttamaa arvon menetystä toimittajan kustannuksella saatava rahallinen hyöty tuskin korvaisi tai ainaakaan ylittäisi. Näin ollen useimmiten tulisi pyrkiä yhteistyöhön perustuvaan yhteistyömalliin eikä vähiten sen ketterään kehittämiseen tuomien merkittävien etujen vuoksi.

3.4.2 Sopimusmallien arviointi

Taulukossa 5 on luokiteltu aiemmin käsitellyt sopimusmallit Stevensin (2009a) esittelemien neljän kriteerin (rakenne, muutostenhallinta, riskin jakautuminen ja yhteistyömalli) perusteella, joita kutakin vastaa yksi taulukon sarake. Luokittelussa on käytetty lähteenä Stevensin (2009b) luokittelua muiden paitsi monivaiheisen, tavoitehintasopimuksen, tavoiteaikasopimuksen, yhteistyösopimuksen ja adVantage-sopimusmallin kohdalla. Kyseisten sopimusmallien osalta luokittelu on tehty niiden kuvausten perusteella.

Luokittelun perusteella voidaan vetää johtopäätöksiä kunkin sopimusmallin soveltuvuudesta ketterään kehittämiseen ja edullisuudesta asiakkaalle. Aiemmin käsitellyn mukaisesti tavoiteltavat ominaisuudet sopimusmallissa ovat asiakkaalle edullinen rakenne, mahdollisuus tehdä muutoksia, jaettu riski sekä yhteistyöhön perustuva yhteistyömalli. Näistä kolme viimeksi mainittua ominaisuutta sopivat mahdollisten arvojensa puolesta parhaiten vertailtaviksi, joten sopimusmallien arviointi suoritetaan aluksi niiden perusteella ja vasta myöhemmässä vaiheessa käsitellään sopimusmallien rakennetta lopullisten tulosten löytämiseksi.

Sopimusmallien arviointi suoritetaan joukko-opin menetelmillä. Käsitellyt sopimusmallit muodostavat perusjoukon, ja ne jaetaan osajoukkoihin muutostenhallinnan, riskin jakautumisen ja yhteistyömallin perusteella. Käsitelyyn otetaan kolme osajoukkoa, joista ensimmäisessä ovat muutosten tekemisen mahdollistavat sopimusmallit, toisessa riskin jakavat sopimusmallit ja kolmannessa yhteistyöhön perustuvaan yhteistyömalliin kannustavat sopimusmallit. Lisäksi osa käsitellyistä sopimusmalleista ei kuulu mihinkään edellä mainittuun joukkoon ja ne mainitaan myöhemmin erikseen. Osajoukot niihin kuuluvine sopimusmalleineen ovat seuraavanlaiset:

- Muutokset mahdollisia: vaiheistettu kehitys, ”rahaa tyhjästä ja muutokset ilmaiseksi”, aika-ja-materiaalit, tavoitehintasopimus, tavoiteaikasopimus, yhteistyösopimus ja adVantage

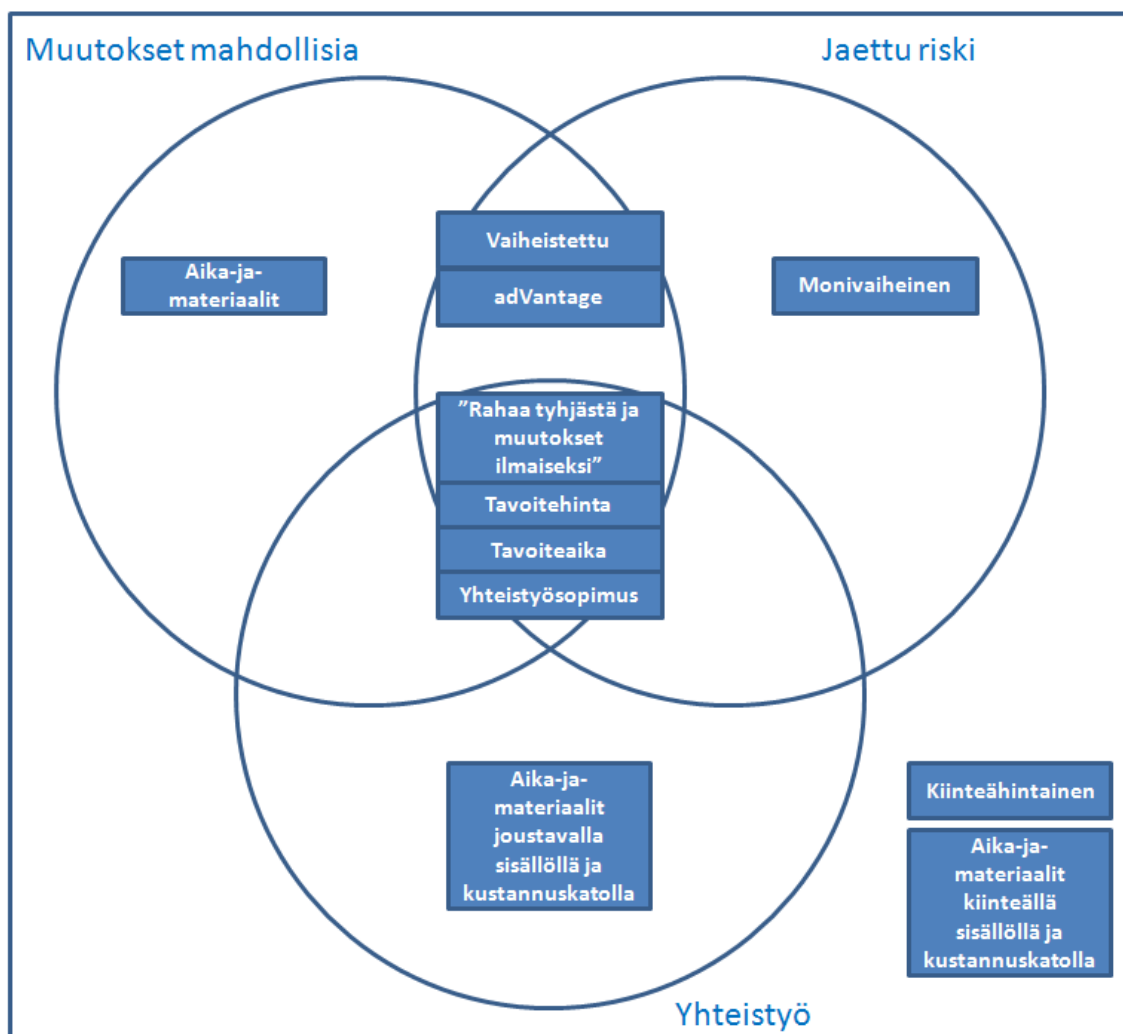
- Riski jaettu: monivaiheinen, vaiheistettu kehitys, "rahaa tyhjästä ja muutokset ilmaiseksi", tavoitehintasopimus, tavoiteaikasopimus, yhteistyösopimus ja adVantage
- Yhteistyö: "rahaa tyhjästä ja muutokset ilmaiseksi", aika-ja-materiaalit joustavalla sisällöllä ja kustannuskatolla, tavoitehintasopimus, tavoiteaikasopimus, yhteistyösopimus
- Mihinkään edellisistä kuulumattomat: kiinteähintainen, aika-ja-materiaalit kiinteällä sisällöllä ja kustannuskatolla

TAULUKKO 5 Sopimusmallien luokittelu

Sopimusmalli	Rakenne	Muutostenhallinta	Riskin jakautuminen	Yhteistyömalli
Kiinteähintainen	Etukäteen sovitun suuruinen hinta, toimitus ja maksu lopussa	Kiinteä sisältö	Toimittajalla	Kilpailullinen / välinpitämätön
Monivaiheinen	Hinta vaihekohtaisesti sovitun, toimitus ja maksu vaiheittain	Kuten kiinteähintainen	Jaettu	Kuten kiinteähintainen
Vaiheistettu kehitys	Kuten monivaiheinen	Vapaasti	Jaettu	Riippuvainen
"Rahaa tyhjästä ja muutokset ilmaiseksi"	Etukäteen sovitun suuruinen hinta, toimitus ja maksu osissa	Priorisointi vapaasti	Jaettu	Yhteistyö
Aika-ja-materiaalit (A&M)	Hinta kulujen mukaan, toimitus lopussa, maksu osissa	Vapaasti	Asiakkaalla	Välinpitämätön
A&M kiinteällä sisällöllä ja kustannuskatolla	Kuten aika-ja-materiaalit	Kiinteä sisältö	Toimittajalla	Riippuvainen
A&M joustavalla sisällöllä ja kustannuskatolla	Kuten aika-ja-materiaalit	Kiinteä sisältö	Asiakkaalla	Yhteistyö
Tavoitehintasopimus	Hinta kulujen mukaan, toimitus lopussa, maksu osissa	Mahdollista	Jaettu	Yhteistyö
Tavoiteaikasopimus	Kuten tavoitehintasopimus	Mahdollista	Jaettu	Yhteistyö
Yhteistyösopimus	Hinta yhdistelmä etukäteen sovittua ja kulujen mukaista, toimitus ja maksu osissa	Vapaasti	Jaettu	Yhteistyö
adVantage	Hinta kulujen mukaan, toimitus ja maksu osissa	Priorisointi vapaasti	Jaettu	Kuten vaiheistettu kehitys

Edellämainittujen kolmen osajoukon leikkaus eli kaikkiin kolmeen osajoukkoon kuuluvat sopimusmallit muodostavat ketterään kehittämiseen soveltuvien ja asiakkaan kannalta edullisten sopimusmallien joukon. Siihen kuuluvat "rahaa tyhjästä ja muutokset ilmaiseksi", tavoitehintasopimus, tavoiteaikasopimus sekä yhteistyösopimus. Edellä suoritettua sopimusmallien arviointia muutostenhal-

linnan, riskin jakautumisen ja yhteistyömallin perusteella on havainnollistettu Venn-diagrammilla kuviossa 10.



KUVIO 10 Sopimusmallien arviointi Venn-diagrammina

Keskimmäisellä leikkausalueella olevat neljä sopimusmallia jakautuvat muutostenhallinnan perusteella kolmeen ryhmään. Kaikkein helpointa muutostenhallintaa on yhteistyösopimuksen tapauksessa. "Rahaa tyhjästä ja muutokset ilmaiseksi"-sopimusmallissa on mahdollista vaatimusten uudelleen priorisointi ja vaatimusten vaihtaminen toisiin työmäärältään vastaavaan kokoon. Tavoitehintasopimuksessa ja tavoiteaikasopimuksessa muutokset ovat mahdollisia, mutta ne saattavat aiheuttaa lisäkustannuksia asiakkaalle. Näistä neljästä sopimusmallista kaikkia voidaan asiakkaan kannalta suositella käytettäväksi ketterän kehittämisen yhteydessä. Rakenteidensa erojen vuoksi ne kuitenkin soveltuvat etupäässä erilaisiin tilanteisiin.

"Rahaa tyhjästä ja muutokset ilmaiseksi" -sopimusmalli rakentuu kiinteähintaisen sopimusmallin pohjalle ja soveltuu parhaiten kiinteähintaisesta sopimusmallista pakon sanelemana kiinni pitävälle asiakkaalle. Kyseinen sopimusmalli on tällaisessa tilanteessa hyvä vaihtoehto ja tarjoaa mahdollisuuden

luopua ketterää kehittämistä vaikeuttavasta puhtaasti kiinteähintaisesta sopimusmallista.

Tavoitehintasopimus ja tavoiteaikasopimus rakentuvat aika-ja-materiaalit-sopimusmallin pohjalle ja niissä on monia ketterää kehittämistä tukevia piirteitä. Eckfeldtin ym. (2005) mukaan näissä sopimusmalleissa on ongelmana muutostenhallinnan vaikeus, koska jokaisen muutoksen kohdalla täytyy määritellä, onko kyseessä sovittuun sisältöön kuuluva vai ylimääräinen ominaisuus ja jälkimmäisessä tapauksessa laskea sen kustannusvaikutukset. Poppendieckin ja Poppendieckin (2003, s.163) mukaan sisältö sen sijaan voidaan sopia melko vapaasti kehittämisen edistyessä ja sopimusmalli jakaa muutosten aiheuttamat kustannusvaikutukset osapuolten kesken. Nämä sopimusmallit ovat hyvä vaihtoehto puhtaalle aika-ja-materiaalit -sopimusmallille esimerkiksi tilanteessa, jossa toimittaja vaatisi nimenomaan aika-ja-materiaalit -sopimusmallin käyttöä. Kustannusten ollessa etusijalla tulee näistä sopimusmalleista valita tavoitehintasopimus ja aikataulun ollessa etusijalla puolestaan tavoiteaikasopimus.

Yhteistyösopimus on hybridisopimusmalli. Sen rakenne on niin ketterään kehittämiseen optimaalinen kuin myös molemmille osapuolille taloudellisesti tasapuolinen eli toimitus osissa, maksu osissa ja hinta osaksi kiinteä ja osaksi kulujen mukainen. Yhteistyösopimusta voidaan ominaisuuksiensa perusteella varauksetta suositella käytettäväksi ketterässä kehittämisessä ja sen tulisi olla vapaan valinnan mahdollistavassa tilanteessa ensisijainen valinta.

3.5 Yhteenveto sopimusmalleista

Sopimus on osapuolten välinen oikeustoimi, jolla ne järjestelevät keskinäiset oikeussuhteensa (Hyvärinen, Hulkko & Ohvo, 2002). Osapuolilla tarkoitetaan asiakasta ja toimittajaa. Ketterässä kehittämisessä arvostetaan asiakasyhteistyötä enemmän kuin sopimusneuvotteluita (Agile Manifesto, 2001), mutta yritysten välisessä yhteistyössä sopimukset ovat kuitenkin välttämättömiä (Stevens, 2009a). Perinteisen kehittämisen yhteydessä vakiintuneiden sopimusmallien soveltaminen ketterään kehittämiseen voi osoittautua ongelmalliseksi ja ketterän kehittämisen tarpeisiin pyritäänkin kehittämään sopimusmalleja, jotka ottavat huomioon sen tuomat uudentyypiset tarpeet ja haasteet.

Sopimusmallit voidaan jaotella kolmeen ryhmään. Kiinteähintaisiin sopimusmalleihin kuuluvat puhtaasti kiinteähintaisen sopimusmallin lisäksi monivaiheinen sopimusmalli, vaiheistettu kehitys -sopimusmalli sekä "rahaa tyhjästä ja muutokset ilmaiseksi" -sopimusmalli. Kaikki edellä mainitut perustuvat hinnan, sisällön ja aikataulun tarkkaan ja pääosin kiinteään etukäteismäärittelyyn. Aika-ja-materiaalit -sopimusmalleihin kuuluvat puhtaan aika-ja-materiaalit -sopimusmallin lisäksi aika-ja-materiaalit -sopimusmalli joustavalla tai kiinteällä sisällöllä ja kustannuskatolla sekä tavoitehintasopimus ja tavoiteaikasopimus. Ne perustuvat toimittajan kulujen ja niiden päälle tulevan voitto-osuuden perusteella tapahtuvaan laskutukseen. Hybridisopimusmallit yhdistelevät kiin-

teähintaisen sekä aika-ja-materiaalit -sopimusmallin ominaisuuksia. Niihin kuuluvat yhteistyösopimus sekä adVantage-sopimusmalli.

Käytännön kokemusten mukaan asiakkaat vaativat usein kehittämissopimuksissa kiinteähintaisen sopimusmallin käyttöä haluten näin hallita hankkeeseen liittyvää riskiä omalta puoleltaan. Kiinteähintainen sopimusmalli ei kuitenkaan ole ketterään kehittämiseen hyvin soveltuva ja lisäksi se ei suinkaan ole niin turvallinen vaihtoehto kuin asiakkaat saattavat kuvitella vaan jopa monin tavoin haitallinen koko kehittämissopimuksen onnistumiselle. Aika-ja-materiaalit -sopimusmalli on ketterään kehittämiseen hyvin soveltuva, mutta sen käyttö vaatii luottamusta ja ei useinkaan ole asiakkaan - usein aiheellisenkin - varautuneisuuden vuoksi mahdollista. Hybridisopimusmallit ovat nimenomaan ketterään kehittämiseen kehitettyjä ja niiden ominaisuudet on pyritty optimoimaan ketterää kehittämistä tukeviksi.

Sopimusmallien arviointi asiakkaan näkökulmasta pitää sisällään neljä kriteeriä: sopimuksen rakenne, riskin jakautuminen osapuolten kesken, muutostenhallinta sekä yhteistyömalli (kilpailullinen, yhteistyö, välinpitämätön, riippuvainen). Sopimusmallien arviointi edellämainittujen kriteereiden perusteella antaa lopputuloksena käsityksen asiakkaan näkökulmasta parhaiten ketterään kehittämiseen sopivista sopimusmalleista. Näitä ovat yhteistyösopimus, "rahaa tyhjästä ja muutokset ilmaiseksi", tavoitehintasopimus sekä tavoiteaika-sopimus. "Rahaa tyhjästä ja muutokset ilmaiseksi" -sopimusmalli on hyvä vaihtoehto kiinteähintaiselle sopimusmallille. Tavoitehintasopimus tai tavoiteaika-sopimus puolestaan ovat hyviä vaihtoehtoja aika-ja-materiaalit -sopimusmallille. Yhteistyösopimus taas on hyvä valinta mihin tahansa tilanteeseen ja sen tulisi mikäli mahdollista olla ensisijainen valinta.

4 ARVON MÄÄRITTÄMINEN

Tässä luvussa tarkastellaan toista osaa asiakkaan näkökulmasta ketterään kehittämiseen, eli arvon määrittämistä. Ensiksi tarkastellaan arvon merkitystä ketterässä kehittämisessä sekä arvon määritelmää. Tämän jälkeen esitellään erilaisia arvon määrittämisen menetelmiä. Seuraavaksi määritellään arvonmäärittämismenetelmien käsittelyn teoreettisena viitekehyksenä toimiva asiakkaan näkökulma ja arvioidaan eri menetelmiä tästä näkökulmasta. Lopuksi esitetään yhteenveto.

4.1 Arvo ketterässä kehittämisessä

Arvon käsite on keskeinen piirre ketterässä kehittämisessä. Ketterä kehittäminen on määritelmänsä mukaisesti arvontuottamisprosessi. Agile-manifestin (Agile Manifesto, 2001) mukaan ketterän kehittämisen päätavoitteena on tyydyttää asiakkaan tarpeet toimittamalla arvoa tuottavaa ohjelmistoa aikaisin ja jatkuvasti. Abrahamssonin ym. (2002) mukaan ketterä kehittäminen keskittyy tuottamaan arvoa ja tekemään sen heti kehittämisprojektin alusta alkaen. Boehmin (2002) mukaan ketterä kehittäminen toimii parhaiten juuri tilanteissa, joissa tärkeintä on arvon kehittäminen nopeasti. Highsmithin (2002) mukaan asiakkaalle merkitsevän arvon toimittaminen määrittelee ketterän kehittämisen onnistumisen.

Arvon tärkeydestä huolimatta sitä käsitellään ketterän kehittämisen yhteydessä käytännön tasolla usein melko ylimalkaisesti, määrittelemättä tarkemmin, mitä sillä tarkoitetaan tai miten sitä voidaan mitata. Esimerkiksi ketterän kehittämisen menetelmistä Scrumia ja XP:tä koskevassa kirjallisuudessa arvon käsite esiintyy useasti, mutta sitä käsitellään vain kehittäjiä kiinnostavassa määrin. Schwaber ja Sutherland (2011) määrittelevät Scrumissa tuoteomistajan olevan vastuussa kehitettävän ohjelmistotuotteen arvon optimoinnista ja tuotteen kehitysjonon järjestämisestä arvon ja prioriteettien mukaan. Beck (1999a) puolestaan määrittelee XP:n suunnittelupeli-käytänteen yhteydessä liiketoimintaihmissa (business people) vastuuksi sisällön priorisoinnin arvon

perusteella. Kummassakaan tapauksessa asiasta päättävälle ei kuitenkaan tarjota tarkempaa ohjeistusta päätöksenteon tueksi.

Jossain määrin erilaisen lähestymistavan arvon käsitteeseen sen sijaan antaa Lean-ajatteluun perustuva lähestymistapa, jossa arvolla on keskeisempi rooli myös käytännön tasolla. Poppendieckin ja Poppendieckin (2003) mukaisessa Lean-ohjelmistokehityksessä näkyy Lean-tuotannon perintönä keskittyminen ohjelmistokehityksen tuotannon tehostamiseen. Lean-ohjelmistokehityksessä esitelläänkin esimerkiksi hukan eliminointi – periaatteen yhteydessä tuotannon tehostamisen menetelmiin perustuva arvovirran kartoittaminen – työkalu, joka on käytännössä toimiva ohjelmistotuotteen arvon parantamisen menetelmä. Tästä huolimatta myöskään Lean-ohjelmistokehitys ei tarjoa asiakkaalle työkaluja varsinaiseen arvon määrittämiseen ja sen pohjalta sisällön perusteltuun priorisointiin.

Edellä käsiteltyjen kolmen yleisimmän ketterän kehittämisen menetelmän ja lähestymistavan perusteella voidaan todeta arvon olevan nimellisestä keskeytydestään huolimatta selvästikin käytännön tasolla laiminlyöty piirre ketterässä kehittämisessä. Kaikissa käsitellyissä menetelmissä ja lähestymistavoissa asiakkaan tehtäväksi annetaan sisällöstä päättäminen, kuitenkin tarjoamatta työkaluja tehdä perusteltuja päätöksiä. Rachevan, Danevan, Sikkelin, Herrmannin ja Wieringan (2010a) vaatimusten priorisointia ketterässä kehittämisessä käsittelevän tutkimuksen mukaan priorisointi kuitenkin edellyttää asiakkaalta tarkkaa tietoa kunkin yksittäisen vaatimuksen tuottamasta arvosta. Näin ollen arvon määrittämismenetelmien puuttuminen vaikuttaa olevan asiakkaan näkökulmasta hyvin todellinen ja ketterän kehittämisen onnistumista uhkaava ongelma. Lopputuloksena onkin mitä todennäköisimmin asiakkaan parhaiden arvauksiensa perusteella suorittama sisällön priorisointi. Racheva ym. (2010a) päätyivät tutkimuksessaan samaan päätelmään toteamalla, että objektiivisten tietojen puutteesta johtuen priorisointi tapahtuu subjektiivisen arvon määrittämisen perusteella. Subjektiivisiin tietoihin perustuva priorisointiprosessi saattaa kylläkin suorituksena tyydyttää toimittajaa, mutta ei varmastikaan ole tie asiakkaan kannalta optimaaliseen lopputulokseen.

4.2 Arvon määritelmä

Taloustieteen alalla arvon määritelmä on vakiintunut ja yksiselitteinen. Bannock, Baxter ja Davis (1998, s.425) esittävät tämän yleisesti tunnetun määritelmän muodossa ”jonkin arvo omistajalleen”. Bannock ym. (1998, s.425) tuovat esille myös kaksi arvon alalajia eli käyttöarvon ja vaihtoarvon, joista ensimmäinen merkitsee omistajalle tuotetun hyödyn arvoa ja jälkimmäinen arvoa vaihdettaessa johonkin muuhun. Liiketoiminnan tarkoitus on tuottaa voittoa ja tähän tavoitteeseen pääseminen edellyttää liiketoiminnan kuluja suurempaa tuottoa ja näiden erotus eli saavutettu voitto on tulosta liiketoiminnassa syntyneestä arvonalisästä (Bannock ym., 1998, s. 425). Taloussanasto (1998, s. 392) määrittelee liikeyrityksen tarkoituksen seuraavasti: ”Talouksyksikkö, joka ostaa tuotanto-

panoksia (tuotannontekijöitä), yhdistää ne tuotantoprosessissa ja myy näin aikaansaadut tuotokset eli suoritteet tulonhankkimiseksi”. Mainitut tuotannontekijät käsittävät mm. työn, luonnonvarat ja pääoman (Taloussanasto, 1998, s. 330). Voiton saamiseksi yrityksen tulee siis myydä tuottamansa suoritteet kalliimmalla kuin millä se ostaa tarvittavat tuotannontekijät eli toisin sanoen saada tuotannontekijöitä yhdistämällä tuotannossaan aikaan arvoa, josta asiakkaat ovat valmiita maksamaan. Arvon syntymistä liiketoiminnassa havainnollistaa Porterin (1985) esittelemä ja laajalti tunnettu arvoketjuajattelu.

Taloustieteen alalla käytettävän arvon määritelmän selkeydestä huolimatta – tai mahdollisesti nimenomaan siitä johtuen (Racheva ym., 2009) – arvolla informaatioteknologian (IT) ratkaisujen ja erityisesti ketterän kehittämisen yhteydessä ei vaikuta olevan yksiselitteistä määritelmää. Melville, Kraemer ja Gurbaxani (2004) toteavat käsittelemiensä tutkimusten perusteella IT-ratkaisujen arvolla viitattavan yleensä niiden vaikutuksiin liiketoiminnan tuottavuuden parantamisessa, voiton parantamisessa, kulujen vähentämisessä, kilpailuedun saavuttamisessa, varaston pienentämisessä tai muissa tehokkuuden mittareissa. Tutkimuksensa perusteella Melville ym. (2004) määrittelevät informaatioteknologian arvon määräytyvän sen perusteella, miten se vaikuttaa organisaation suorituskykyyn prosessikohtaisesti ja organisaation laajuisesti tehokkuuden ja kilpailukyvyn osalta. IT-ratkaisuja käsitellään siis liiketoiminnassa yhtenä tuotannontekijänä, jonka tehtävänä on tehostaa tuotantoa ja saada aikaan enemmän arvoa ja siten parantaa liiketoiminnasta saatavaa voittoa.

Ketterän kehittämisen tarkoituksena on yrityksen tuotantoa ja sitä kautta liiketoiminnan voittoa parantavan ohjelmiston eli IT-ratkaisun kehittäminen. Ketterän kehittämisen yhteydessä käytettävä arvon määritelmä onkin siis lähikohtaisesti läheisesti yhteydessä edellä esiteltyyn IT-ratkaisun arvoon. Ketterän kehittämisen yhteydessä käytettävän arvon tarkan määritelmän löytämistä varten tutustumme Rachevan ym. (2009) tutkimuksessa esitettyyn yhteenveetoon kirjallisuudessa esitetyistä määritelmistä. Rachevan ym. (2009) mukaan ketterän kehittämisen yhteydessä arvo määritellään eri lähteissä toisistaan poikkeavilla tavoilla, jotka on esitetty taulukossa 6. Agile Journal - verkkosivustolla sijainneet lähteet eivät ole enää löydettävissä, ja niihin viitataan vain Rachevan ym. (2009) ilmoittamassa muodossa.

Racheva ym. (2009) toteavat kaikkien löytämiensä arvon määritelmää ketterän kehittämisen yhteydessä käsittelevien lähteiden olevan tieteellisten tutkimusten sijasta ohjelmistokehityksen ammattinharjoittajien (practitioners) artikkeleita ja arvelevat tieteenharjoittajien pitävän arvon käsitettä itsestään selvänä taloustieteen alalla suoritettun tutkimuksen perusteella. Rachevan ym. (2009) mukaan kirjallisuudesta löytyneille arvon määritelmille yhteisiä piirteitä ovat:

- 1) Arvo on yleensä kvalitatiivisesti mitattavaa.
- 2) Arvo on yleensä subjektiivista.
- 3) Arvo hallitsee vaatimusten priorisointia.
- 4) IT-ratkaisun arvo vaatii luottamusta, koska sitä ei voida määritellä varmasti ja tarkasti.
- 5) IT-ratkaisun arvo riippuu yleensä myös sen ulkopuolisista prosesseista.

TAULUKKO 6 Arvon määritelmiä ketterän kehittämisen yhteydessä (Racheva ym., 2009, s. 146)

Lähde	Määritelmä
Barnett, L. (2007). Agile projects must measure business value. Agile Journal (January 2007). Lähdettä ei ole enää löydettävissä.	...liiketoiminta-arvo, mitattuna liiketoiminnan tuoton, osakekurssin, markkinaosuu- den ja muiden liiketoiminnan mittareiden perusteella. Arvo on asiakkaan silmissä...
Patton (2008)	Liiketoiminta-arvo tuottaa voittoa ohjelmis- tosta maksavalle organisaatiolle voiton kas- vamisen, kustannusten välttämisen tai pal- velun parantumisen muodossa.
Pettit, R. (2007). Business Value Applied: Aligning Day To Day With Business Imper- ative. Agile Journal (January 2007). Lähdettä ei ole enää löydettävissä.	Liiketoiminta-arvo on kommunikoinnin väline: käytämme liiketoiminta-arvoa kommunikoidaksemme arvoa, prioriteetteja ja motivaatiota.
Rawsthorne (2004)	Liiketoiminta-arvo on sitä mistä johto on valmis maksamaan; arvo voidaan määritellä vain lopullisen asiakkaan toimesta. Se on mielenkiintoista vain tuotteen, palvelun tai molempien muodossa ilmaistuna, täyttäes- sään asiakkaan tarpeet tietyllä hinnalla ja tiettyinä aikana.
Poole, M. (2007). Business and IT - A Mar- riage Made in Heaven? Agile Journal (Octo- ber 2007). Lähdettä ei ole enää löydettävissä.	Informaatioteknologian arvoa ei ehkä ole mahdollista määritellä irrallisena. Liiketoi- minta-arvo = $F(x) + F(y) + F(z) + \dots$ eli monimutkainen kaava jonka täytyy ottaa huo- mioon useita seikkoja niiden ollessa samalla muutoksessa.

Rachevan ym. (2009) yhteenvedossa mukana olevien lähteiden lisäksi määri- telmiä arvolle ketterän kehittämisen yhteydessä ovat esittäneet ainakin Cohn (2005) sekä Larman ja Vodde (2009). Cohnin (2005) mukaan ohjelmiston omi- naisuuden arvo merkitsee organisaation sen avulla ansaitsemaa tai säästämää rahasummaa. Larman ja Vodde (2009) puolestaan antavat Lean- ohjelmistokehityksen käsittelyn yhteydessä arvolle määritelmän, jonka mukaan arvo on tuotteen kehittämisessä kaikki se, mistä asiakas on valmis maksamaan, kun taas kaikki muu tuotteen kehittämisessä tehtävä on Lean-ajattelun mukai- sesti hukkaa. Näiden määritelmien sisällön voidaan katsoa olevan pääosin yh- teneväinen Rachevan ym. (2009) esittämän yhteenvedon kanssa, joskin Cohnin (2005) määritelmä painottaa tarkkaa rahamääräistä arvoa kun taas Racheva ym. (2009) painottavat arvon olevan yleensä kvalitatiivista ja suhteellista. Rachevan ym. (2009) määritelmä ei kuitenkaan sulje pois tarkan rahamääräisen arvon mahdollisuutta.

Esitellyistä määritelmistä voidaan nyt yhteenvetona johtaa tässä tutkiel- massa käytettävä määritelmä, jonka mukaan *ketterän kehittämisen tuottama arvo* on kehittämisen tuloksena syntyneen ohjelmiston sitä käyttävälle asiakkaalle tuottama arvo asiakkaan kokemana ja tämä arvo on yleensä subjektiivista ja kvalitatiivisesti mitattavaa.

4.3 Arvon määrittämisen menetelmiä

Rachevan ym. (2009) tutkimuksessa on esitetty yhteenveto arvon käsitettä käsittelevästä kirjallisuudesta ja kyseisen tutkimuksen käyttämät lähteet ovatkin hyvä lähtökohta myös arvon määrittämisen menetelmien etsimiselle. Racheva ym. (2009) kiinnittävät huomiota siihen, että tutkimuksissa on käytetty osittain eri termejä arvolle kuten esimerkiksi ansaittu arvo (earned value), ansaittu liiketoiminta-arvo (earned business value) tai koettu liiketoiminta-arvo (perceived business value). Agile-manifestissa (Agile Manifesto, 2001) arvo on mainittu muodossa "arvokas" (valuable), joka ei rajaa pois mitään edellä esitetyistä termeistä. Käsittelyyn onkin otettu kaikki Rachevan ym. (2009) mainitsemat arvoon viittaavat tutkimukset. Rachevan ym. (2009) arvon käsittelyn yhteydessä käyttämät lähteet sekä kunkin lähteen käyttämä arvoon viittaava termi ja mahdollisesti sisältämä menetelmä arvon määrittämiseen on esitetty taulukossa 7.

TAULUKKO 7 Arvon määrittämisen menetelmiä

Lähde	Arvoon viittaava termi	Arvon määrittämisen menetelmä
Patton (2008)	Liiketoiminta-arvo	Liiketoimintatavoitemalli, jonka tavoitteena on tuoda esiin ja priorisoida liiketoiminnan tavoitteet. Ei sisällä ehdotusta arvon määrittämisen menetelmäksi.
Rawsthorne (2004)	Liiketoiminta-arvo ja ansaittu liiketoiminta-arvo	Sisältää menetelmän kehittämissuorituksen liiketoiminta-arvosta tietyllä hetkellä toteutetun osan eli ansaitun liiketoiminta-arvon määrittämiseksi ketterän kehittämisen yhteydessä ja lisäksi ehdotuksia koko kehittämissuorituksen liiketoiminta-arvon määrittämisen menetelmäksi.
Rawsthorne (2006)	Sama kuin Rawsthorne (2004)	Sama kuin Rawsthorne (2004)
Alleman, Henderson & Seggelke (2003)	Ansaittu arvo	Sisältää ehdotuksen perinteisen kehittämisen yhteydessä käytetyn ansaitun arvon määrittämisen soveltamisesta ketterään kehittämiseen.
Cabri & Griffiths (2006)	Ansaittu arvo	Sama kuin Alleman ym. (2003)
Sulaiman, Barton & Blackburn (2006)	Ansaittu arvo	Sama kuin Alleman ym. (2003)
Favaro (2005)	Taloudellinen arvo (economic value)	Hyvin suppea artikkeli eikä sisällä ehdotusta arvon määrittämisen menetelmäksi.
Müller & Padberg (2003)	Taloudellinen arvo (economic value)	Sisältää ehdotuksen nykyisen nettoarvon (net present value) määrittämiseksi diskonttaamalla tuleva arvo nykyhetkeen.
Pinheiro, Maurer & Sillito (2008)	Koettu liiketoiminta-arvo	Käsittelee ketterän kehittämisen sijasta iteratiivista Resource Unified Process (RUP) -menetelmää eikä sisällä ehdotusta arvon määrittämisen menetelmäksi.

Rachevan ym. (2009) käyttämistä lähteistä Agile Journal -verkkosivustolla sijainneet Barnettin (2007), Pettitin (2007) ja Poolen (2007) artikkelit eivät ole enää löydettävissä, eikä niihin voida tarkemmin tutustua, josta syystä niitä ei ole otettu mukaan käsittelyyn.

Taulukon perusteella artikkelit voidaan jaotella neljään ryhmään niiden käyttämän arvoon viittaavan termin ja sitä vastaavan arvon määrittämisen menetelmän perusteella:

- Ansaitun arvon määrittäminen ketterään kehittämiseen sovellettuna (Alleman ym., 2003; Cabri & Griffiths, 2006; Sulaiman ym., 2006)
- Taloudellisen arvon määrittäminen (Müller & Padberg, 2003)
- Liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittäminen (Rawsthorne 2004; 2006)
- Arvon määrittämisen menetelmää sisältämättömät artikkelit (Patton, 2008; Favaro, 2005; Pinheiro ym., 2008)

Seuraavaksi tutkitaan tarkemmin kutakin kolmesta ensimmäisestä ryhmästä ja niiden esille tuomista arvon määrittämisen menetelmistä. Lisäksi tutkittavaksi otetaan edellä käsitellyn listan ulkopuolelta Wiegertsin (1999) esittelemä suhteellisen painotuksen (relative weighting) menetelmä. Sen yhteydessä käsitellään lisäksi Rachevan ym. (2010a) sekä Rachevan, Danevan, Sikkelin ja Buglionen (2010b) esittämä negatiivisen arvon käsite ja sen merkitys arvon määrittämisessä.

4.3.1 Ansaitun arvon määrittäminen ketterään kehittämiseen sovellettuna

Allemanin ym. (2003), Cabrin ja Griffithsin (2006) sekä Sulaimanin ym. (2006) tutkimukset käsittelevät *ansaitun arvon määrittämistä* (earned value management, EVM) ketterän kehittämisen yhteydessä. Allemanin ym. (2003) mukaan ansaitun arvon määrittämistä on käytetty kauan ennen ketterän kehittämisen menetelmiä mittaamaan kehittämissuunnitelmissa tuotettua arvoa suhteessa kustannuksiin. Sulaimanin ym. (2006) mukaan ansaitun arvon määrittämistä on käytetty projektinhallinnassa 1960-luvulta lähtien erityisesti puolustus- ja rakennusteollisuuden projekteissa. PMI Standards Committee (1996) mukaan ansaitun arvon määrittäminen on menetelmä projektin tehokkuuden mittaamiseksi, ja se mittaa tehdyn työn määrää suhteessa suunnitellun työn määrään tarkoituksena selvittää, etenevätkö kustannukset ja aikataulu suunnitellusti.

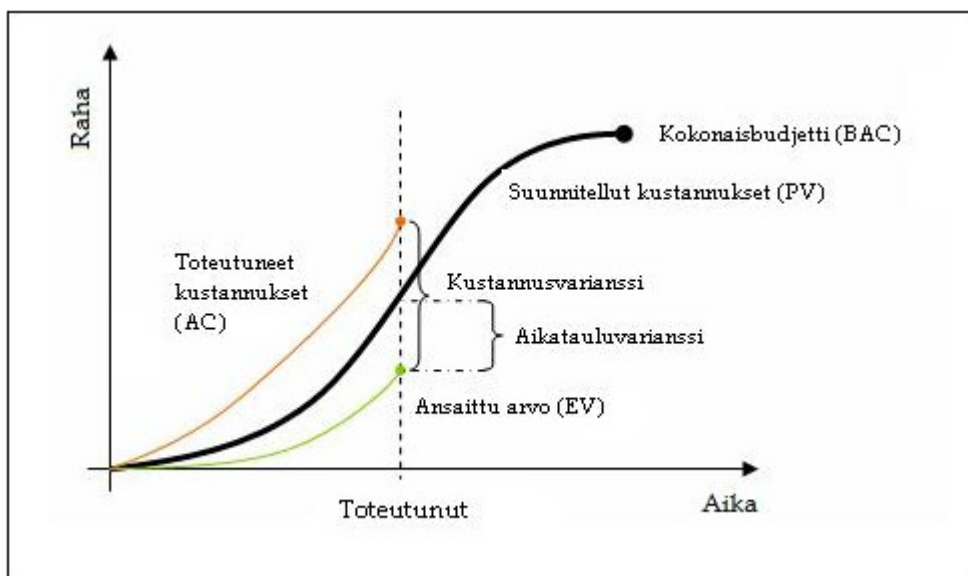
Allemanin ym. (2003) sekä Cabrin ja Griffithsin (2006) mukaan ansaitun arvon määrittäminen käsittelee kolmea projektin kustannuksiin liittyvää arvoa eli suunniteltuja kustannuksia (Planned Value, PV), toteutuneita kustannuksia (Actual Cost, AC) sekä toteutetulle työlle suunniteltuja kustannuksia eli ansaittua arvoa (Earned Value, EV). Taulukossa 8 esitetään edellä mainittujen arvojen perusteella tehtävät laskelmat eli kustannusvarianssi, aikatauluvarianssi, kustannusten suorituskykyindeksi, aikataulun suorituskykyindeksi, arvio valmistuessa ja arvio valmistumiseen. Taulukossa on esitetty lisäksi laskelmien seli-

tykset ja laskentakaava. Laskelmissa on käytetty edellä mainittujen arvojen lisäksi lähtötietona projektin kokonaisbudjettia (Budget at Completion, BAC).

TAULUKKO 8 Ansaitun arvon määrittämisen laskelmat

Laskelma	Selitys	Kaava
Kustannusvarianssi (Cost Variance, CV)	Suunniteltujen ja toteutuneiden kustannusten ero	$CV = EV - AC$
Aikatauluvarianssi (Schedule Variance, SV)	Tehdyn investoinnin ja ansaitun arvon ero	$SV = EV - PV$
Kustannusten suorituskykyindeksi (Cost Performance Index, CPI)	Kustannusten normalisoitu suorituskykyindeksi	$CPI = EV / AC$
Aikataulun suorituskykyindeksi (Schedule Performance Index, SPI)	Aikataulun normalisoitu suorituskykyindeksi	$SPI = EV / PV$
Arvio valmistuessa (Estimate at Completion, EAC)	Arvio kokonaiskustannuksista	$EAC = AC + ((BAC - EV) / CPI)$
Arvio valmistumiseen (Estimate to Completion, ETC)	Arvio jäljellä olevista kustannuksista	$ETC = (BAC - EV) / CPI$

Cabri ja Griffiths (2006) puolestaan ovat havainnollistaneet ansaitun arvon määrittämisen arvojen ja laskelmien merkitystä käytännössä kuvion 11 mukaisesti. Kuviossa on esitetty esimerkkitapauksen suunnitellut kustannukset, toteutuneet kustannukset, ansaittu arvo, kokonaisbudjetti, kustannusvarianssi ja aikatauluvarianssi.



KUVIO 11 Ansaitun arvon määrittäminen (Cabri & Griffiths, 2006, s.18)

Sulaimanin ym. (2006) mukaan ansaitun arvon määrittäminen ohjelmistotuotannon yhteydessä vaatii kuitenkin mittavaa etukäteissuunnittelua ja työnosituksista, jotka taas edellyttävät sisällön pysymistä kiinteänä. Sen vuoksi kyseinen menetelmä soveltuu sellaisenaan perinteiseen kehittämiseen, mutta ei ketterään kehittämiseen. Allemanin ym. (2003), Cabrin ja Griffithsin (2006) ja Sulaimanin

ym. (2006) tutkimukset käsittelevätkin nimenomaan ansaitun arvon määrittämisen soveltamista ketterän kehittämisen erityistarpeisiin. Merkittävin ero perinteisen ja ketterän ansaitun arvon määrittämisen välillä on laskelmien aikajänne, joka perinteisessä on koko projektin pituinen, mutta ketterässä vain yhden iteraation (Alleman ym., 2003, Cabri & Griffiths, 2006) tai julkaisun (Sulaiman ym., 2006) pituinen. Näin myös ketterässä kehittämisessä voidaan käyttää hyväksi ansaitun arvon määrittämistä, menettämättä kuitenkaan muutoksen hyväksymisen periaatetta. Lisäksi ketterässä ansaitun arvon määrittämisessä sisältö määritellään testattavina vaatimuksina, käyttäjätarinoina ja tehtävinä ja valmistunut osa työstä lasketaan rahamääräisen arvon sijasta tarinapisteinä (story points) eli käyttäjätarinoiden suhteellisten kokojen perusteella (Alleman ym., 2003; Cabri & Griffiths, 2006; Sulaiman ym., 2006). Sulaimanin ym. (2006) esittelemät perinteisen ja ketterän ansaitun arvon määrittämisen väliset erot on esitetty tarkemmin taulukossa 9.

TAULUKKO 9 Perinteisen ja ketterän ansaitun arvon määrittämisen erot (Sulaiman ym., 2006, s. 3)

Termi	Perinteinen EVM	Ketterä EVM
Suorituskyvyn mittauksen lähtökohta (Performance Measurement Baseline, PMB)	Kaikkien sisällön osien työmääräarviot yhteenlaskettuna	Julkaisuun suunniteltujen tarinapisteiden yhteismäärä
Aikataulun lähtökohta (Schedule Baseline)	Kaikkien sisällön osien aikatauluarviosta laskettu kokonaiskesto	Suunniteltujen sprinttien kokonaismäärä kerrottuna sprinttien pituudella
Kokonaisbudjetti (Budget at Complete, BAC)	Julkaisun tai projektin kokonaisbudjetti	Julkaisun kokonaisbudjetti
Suunniteltu valmistumisaste (Planned Percent Complete, PPC)	Tietyllä hetkellä valmistuneeksi suunniteltu osa sisällöstä	Tietyllä hetkellä käynnissä olevan sprintin järjestysnumero jaettuna suunniteltujen sprinttien kokonaismäärällä
Toteutunut valmistumisaste (Actual Percent Complete, APC)	Tietyllä hetkellä valmiina olevien sisällön osien rahamääräinen arvo jaettuna kokonaisbudjetilla	Tietyllä hetkellä valmiina oleva tarinapisteiden määrä jaettuna tarinapisteiden kokonaismäärällä

4.3.2 Taloudellisen arvon määrittäminen

Müllerin ja Padbergin (2003) tutkimus käsittelee *taloudellisen arvon määrittämistä* (economic value) XP-menetelmällä toteutetuissa projekteissa. Müllerin ja Padbergin (2003) esittelemä arvon määrittämisen menetelmä perustuu nykyisen nettoarvon (net present value) käsitteeseen. Nykyisen nettoarvon käsite perustuu kehittämisprojektin tulevaisuudessa tuottaman voiton diskonttaukseen, joka on taloustieteessä yleisesti käytetty menetelmä ja merkitsee sen nykyarvon eli tarkasteluhetken arvon laskemista (Taloussanasto, 1998, s.39). Diskonttauksen käyttäminen tässä yhteydessä havainnollistaa kehittämiseen kuluvan ajan

merkitystä valmiista ohjelmistosta saatavaan voittoon eli nopea valmistuminen lisää voittoa ja hidas valmistuminen puolestaan vähentää sitä. Menetelmän mukaan nykyisen nettoarvon (net present value, NPV) määrittely tapahtuu kuviossa 12 esitetyllä kaavalla:

$$\text{NPV} = \frac{\text{varallisuusarvo}}{(1 + \text{diskonttauskurssi})^{\text{kehitysaika}}} - \text{kehityskustannukset}$$

KUVIO 12 Nykyisen nettoarvon kaava (Müller & Padberg, 2003, s.170)

Nykyisen nettoarvon kaavassa varallisuusarvo on kehitettävästä ohjelmistosta saatava voitto eli sen lopulta realisoituva arvo. Diskonttauskurssille asetettava arvo määrittää ohjelmiston valmistumisnopeuden merkityksen, ja kehitysaajan pituus vaikuttaa diskonttauskorvaan arvoon eksponentiaalisesti. Kaavassa otetaan huomioon myös kehityskustannukset (diskonttaamattomina) ja näin lopputuloksena saadaan nimenomaan nettoarvo eli arvo vähennettynä kustannuksilla. (Müller & Padberg, 2003.)

Mutschler, Zarvic ja Reichert (2007) huomauttavat nykyiseen nettoarvoon liittyen, että yleisesti ottaen toteutettavaksi tulisi valita vain sellaisia kehittämisprojekteja, joiden nykyinen nettoarvo on positiivinen, koska silloin niiden tuotto ylittää diskonttauskurssin eli niiden voidaan laskelman perusteella odottaa tuottavan voittoa.

4.3.3 Liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittäminen

Rawsthornen (2004, 2006) tutkimukset käsittelevät *liiketoiminta-arvoa* (*business value*) ja *ansaittua liiketoiminta-arvoa*. Molemmat tutkimukset käsittelevät samaa menetelmää, jonka tavoitteena on kehitettävän ohjelmiston arvon jakaminen osiin ja tietyllä hetkellä ansaitun arvon määrittäminen toteutettujen osien summana. Menetelmä voidaan jakaa kolmeen vaiheeseen, joista ensimmäisessä määritellään koko kehittämisprojektin eli kehitettävän ohjelmiston arvo, toisessa lasketaan arvo sisältöosille ja kolmannessa määritellään tietyllä hetkellä ansaittu arvo. (Rawsthorne, 2004 & 2006.)

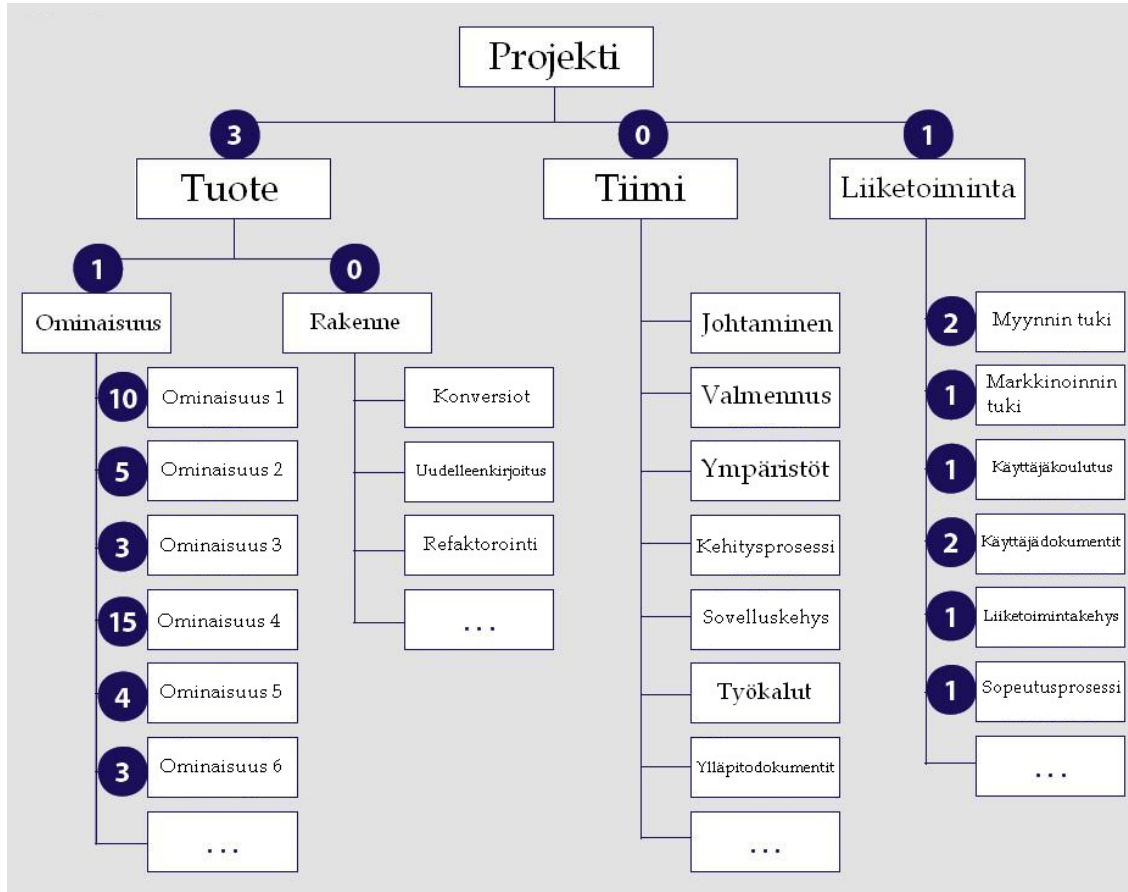
Rawsthorne (2006) mukaan aluksi on tärkeintä saada kehittämisen kohteen arvolle jonkinlainen yhteenlaskukelpoinen numeerinen määritelmä ja hänen mukaansa sellaisen määrittämiseen organisaatioissa käytetään yleensä formaaleja sijoitetun pääoman tuottolaskelmia (return on investment, ROI) tai tuloslaskelmia (profit and loss, PNL) tai epäformaaleja projektien suhteelliseen arvoon perustuvia laskelmia. Sijoitetun pääoman tuottolaskelmia käsittelevät lukuisten muiden lähteiden ohella Harris, Herron ja Iwanicki (2008, s.16), joiden mukaan ROI ilmaisee sijoituksella ansaittavaa tuottoa tai säästöä ja ilmaistaan yleensä prosenttimääräisenä lukemana sijoituksesta (eli pääoman tuotto prosenttina) vuosittaisena tai projektin keston aikaisena ja tulot muutettuna nykyiseen nettoarvoonsa. Investointi on perusteltavissa vain, mikäli sen voidaan to-

distaa tuottavan toteutuskustannustensa ja voittomarginaalin verran tuottoa tai säästöä. Tuloslaskelma puolestaan on Taloussanaston (1998, s.326) mukaan "tilinpäätöksessä tilikauden tuottojen ja kulujen yhteenveto vähennyslaskuasetelman muodossa". Rawsthornen (2006) ehdottamassa yhteydessä kyseistä laskelmaa voitaisiin käyttää yhden projektin osalta ilmaisemaan sen aiheuttamien kustannusten ja voittojen tai säästöjen erotusta. Mikäli laskelman lopputulos olisi tällöin positiivinen, olisi investoinnista odotettavissa voittoa. Toinen ehdotettu vaihtoehto koko kehittämisprojektin arvon määrittämiseen, eli epäformaalin ja suhteelliseen arvoon perustuva laskelma, ei edellytä rahassa mitattavia tietoja tuotoista tai kustannuksista. Sen toteuttamiseen riittää aiemmin mainitun mukaisesti jonkinlainen yhteenlaskukelpoinen numeerinen arvo, joka mahdollistaa eri kehittämisprojektien arvioitujen arvojen vertailemisen toisiinsa. Kaikkien edellä mainittujen laskelmien taustalla on nähtävä muut vaihtoehdot kyseisten investointivarojen käytölle, ja niitä voidaan hyödyntää joko arvioitaessa tietyn investoinnin kannattavuutta tai vertailtaessa useita mahdollisia investointeja keskenään.

Rawsthornen (2004, 2006) menetelmässä on koko kehittämisprojektin arvon määrittämisen jälkeen tärkeällä sijalla työnositurakenne (work breakdown structure, WBS), joka on perinteisessä projektinhallinnassa usein käytetty työkalu. PMI Standards Committee (1996) mukaan työnositurakenne on tuotosuuntautunut projektiosien ryhmittely, joka organisoii ja määrittelee kokonaissisällön. Työnositurakenne auttaa hallitsemaan sisältöä ja sen osia sekä hahmotamaan osien välisiä suhteita. Esimerkki työnositurakenteesta on esitetty Rawsthornen (2006) mukaisesti kuviossa 13. Sisältö on jaettu kolmeen haaraan (tuotteeseen liittyvät, kehitystiimiin liittyvät ja liiketoimintaan liittyvät) ja niin haaroille kuin niiden sisältämille yksittäisille sisältöosillekin on merkitty lisäksi suhteelliset painoarvot eli käytetty epäformaalia ja suhteellista menetelmää arvon määrittämiseksi.

Työnositurakenteen mukaisesti projektin tuotehaaran painoarvo on kolme kertaa suurempi kuin liiketoimintahaaran ja tiimihaaralla tai tuotehaaran rakennealahaaralla ei ole lainkaan painoarvoa. Näin ollen lisätyön tekeminen tuotteen rakenteen parantamiseksi tai tiimin työn tehostamiseksi ei tuota lisää arvoa, kun taas uusien ominaisuuksien tekeminen tuotteeseen tai liiketoiminnan tehostaminen tuottavat. Edelleen yksittäisten sisältöosien tapauksessa esimerkiksi ominaisuus 4 tuottaa 1.5 kertaa enemmän arvoa kuin ominaisuus 1 ja jopa 5 kertaa enemmän arvoa kuin ominaisuus 3 tai ominaisuus 6.

Suhteellisten painoarvojen antaminen ominaisuuksille on helposti hahmotettavissa, mutta sen käyttömahdollisuudet ovat merkittävät. Seuraavaksi esitetäänkin sisältöosan arvon laskeminen käyttäen hyväksi edellä määritellyjä suhteellisia painoarvoja. Osan arvo saadaan osan vanhemman eli sitä ylempään osan arvosta kerrottuna kyseisen osan ja muiden osien suhteellisen painoarvon suhteesta saatavalla kertoimella. Sisältöosan x liiketoiminta-arvo $BV(x)$ saadaan kuviossa 14 esitetyllä kaavalla.



KUVIO 13 Työnosituskaavio (mukaillen Rawsthorne, 2006, s.4)

$$BV(x) = BV(\text{vanhempi}) * \left[\frac{\text{painoarvo}(x)}{\text{painoarvo}(x) + \sum_{\text{sisarukset}} \text{painoarvo}(\text{sisarukset})} \right]$$

KUVIO 14 Sisältöosan liiketoiminta-arvon kaava (mukaillen Rawsthorne, 2006, s.5)

Arvon laskenta tulee suorittaa koko työnositusrakenteelle, jolloin ylimmän tason vanhemman eli koko puurakenteen arvo on koko kehittämisprojektin eli kehitettävän ohjelmiston arvo. Koko kehittämisprojektissa tietyllä hetkellä ansaittu liiketoiminta-arvo EBV(projekti) saadaan kyseisellä hetkellä toteutettujen sisältöosien edellä mainitusta kaavasta saatujen arvojen summana kuviossa 15 esitetyllä kaavalla:

$$EBV(\text{projekti}) = \sum_{\text{toteutetut}} BV(\text{osa})$$

KUVIO 15 Ansaitun liiketoiminta-arvon kaava (mukaillen Rawsthorne, 2006, s.5)

Rawsthornen (2004, 2006) esittämän arvon määrittämisen menetelmän yhteydessä tulee sen monista hyvistä puolista huolimatta huomioida myös sen käyttämän työnositurakenteen mahdollinen yhteensopimattomuus ketterän kehittämisen kanssa. Työnositurakenteen käyttäminen koko projektiin suunnitellun työn jakamiseksi osiin viittaa perinteiselle kehittämiselle ominaiseen mittavaan etukäteissuunnitteluun ja sisällön aikaiseen kiinnittämiseen. Sulaiman ym. (2006) viittaavat tutkimuksessaan nimenomaan työnositurakenteeseen eräänä perinteisen ansaitun arvon määrittämisen huonona puolena. Työnositurakennetta voidaan käyttää myös ketterän kehittämisen yhteydessä, mutta silloin sillä voidaan määritellä vain yhdessä iteraatiossa tai julkaisussa tehtävä osa työstä.

4.3.4 Suhteellinen painotus ja negatiivinen arvo

Wiegiers (1999) esittelee ominaisuuksien suhteelliseen priorisointiin tarkoitettua suhteellisen painotuksen (relative weighting) menetelmän, joka perustuu ominaisuuksien arvoon, kustannuksiin ja riskiin. Wiegiersin (1999) mukaan kyseessä on priorisointimenetelmä, mutta priorisointi suoritetaan tässä menetelmässä selvästi määritellyn arvon perusteella ja sitä voidaan siksi pitää myös arvomääritysmenetelmänä. Menetelmässä ominaisuuksille määritellään aluksi suhteellinen hyöty (benefit) asteikolla 1-9. Sen jälkeen ominaisuuksille määritellään samoin asteikolla 1-9 rangaistus (penalty), joka merkitsee asiakkaan liiketoiminnalle aiheutuvaa haittaa, mikäli kyseistä ominaisuutta ei toteutettaisi. Tämän jälkeen kunkin ominaisuuden arvo lasketaan määritellyn hyödyn ja rangaistuksen summana. Kokonaisarvo saadaan laskemalla kaikkien ominaisuuksien arvot yhteen ja kunkin ominaisuuden prosentuaalinen osuus kokonaisarvosta saadaan jakamalla kyseisen ominaisuuden arvo kokonaisarvolla. Seuraavaksi kullekin ominaisuudelle määritellään suhteellinen kustannus asteikolla 1-9, joiden yhteissummana saadaan kokonaiskustannus ja edelleen kunkin ominaisuuden prosentuaalinen osuus kokonaiskustannuksesta jakamalla kyseisen ominaisuuden kustannus kokonaiskustannuksella. Lopuksi lasketaan kunkin ominaisuuden prioriteetti jakamalla prosentuaalinen arvo prosentuaalisen kustannuksen ja prosentuaalisen riskin summalla (prioriteetti = arvo % / (kustannus % + riski %). Ominaisuuksien lista järjestetään lasketun prioriteetin perusteella laskevaan järjestykseen, jonka seurauksena listan ylimpänä olevilla ominaisuuksilla on suotuisin arvon, kustannusten ja riskin välinen suhde. (Wiegiers, 1999.) Cohn (2005) referoi suhteellisen painotuksen menetelmää ja käsittelee sen arvon ja kustannusten osalta jättäen pois riskin, jolloin myös prioriteetin laskukaavasta jää pois riskin osuus. Esimerkki suhteellisen painotuksen menetelmän käytöstä on esitetty taulukossa 10 perustuen Wiegiersin (1999) esimerkkiin ja siitä on jätetty pois riski-sarake Cohnin (2005) esittämän esimerkin mukaisesti. Riski-sarakkeen puuttumista lukuun ottamatta taulukon sarakkeet vastaavat suhteellisen painotuksen menetelmän vaiheita eli kullekin ominaisuudelle lasketaan suhteellisen hyödyn ja rangaistuksen perusteella arvo ja arvon osuus sekä suhteellisen kustannuksen perusteella kustannusten osuus sekä näiden perusteella prioriteetti.

TAULUKKO 10 Suhteellisen painotuksen menetelmä (mukaillen Wiegers, 1999, s.6)

Ominaisuus	Suhteellinen hyöty	Suhteellinen rangaistus	Arvo	Arvo %	Suhteellinen kustannus	Kustannus %	Prioriteetti
Ominaisuus 1	5	3	8	23.5	2	20	1.175
Ominaisuus 2	9	7	16	47.1	5	50	0.942
Ominaisuus 3	5	5	10	29.4	3	30	0.333
Yhteensä	19	15	34	100	10	100	-----

Suhteellisen painotuksen menetelmässä erityisen mielenkiintoista on sen käyttämä rangaistuksen käsite. Samasta käsitteestä käytetään kirjallisuudessa myös termiä negatiivinen arvo. Esimerkiksi Rachevan ym. (2010a, 2010b) mukaan lähes kaikissa heidän tutkimissaan käytännön tapauksissa korostettiin tiettyjen ominaisuuksien tuottaman arvon määrittämisen sijasta negatiivisen arvon käsitettä eli tietyn ominaisuuden puuttumisen aiheuttamaa arvon alenemista. Negatiivisen arvon käsite tuo esille mielenkiintoisen huomion siitä, ettei jonkin puuttuvan ominaisuuden arvo lopulliselle ohjelmistolle olekaan suinkaan nolla vaan negatiivinen eli kokonaisuudelle haitallinen ja merkitsee käytännössä ohjelmiston käytön hankaloitumista puuttuvan ominaisuuden vuoksi. Rachevan ym. (2010a, 2010b) mukaan negatiivisen arvon käsitteen keskeinen osa käytännön tapauksissa tuo esille pragmaattisen suhtautumisen arvon käsitteeseen käytettävissä olevien resurssien ollessa rajalliset. Eli kaikkien mahdollisten arvoa kasvattavien ominaisuuksien toteuttamisen sijasta pyritään rajallisten resurssien puitteissa toteuttamaan ne ominaisuudet, joita ilman ohjelmisto ei kykene suorittamaan tarkoitustaan. Rachevan ym. (2010a, 2010b) mukaan negatiivisen arvon käsitteen merkitys osoittaa tarpeen määrittellä uudelleen koko arvon käsite ketterän kehittämisen yhteydessä.

4.4 Arvon määrittämisen menetelmät asiakkaan näkökulmasta

Tässä alaluvussa arvioidaan aiemmin esiteltyjä arvon määrittämisen menetelmiä asiakkaan näkökulmasta. Aluksi esitellään menetelmien arvioinnin kriteerit ja niiden muodostama asiakkaan näkökulma ja sen jälkeen käytetään esiteltyjä kriteerejä menetelmien arvioimiseksi.

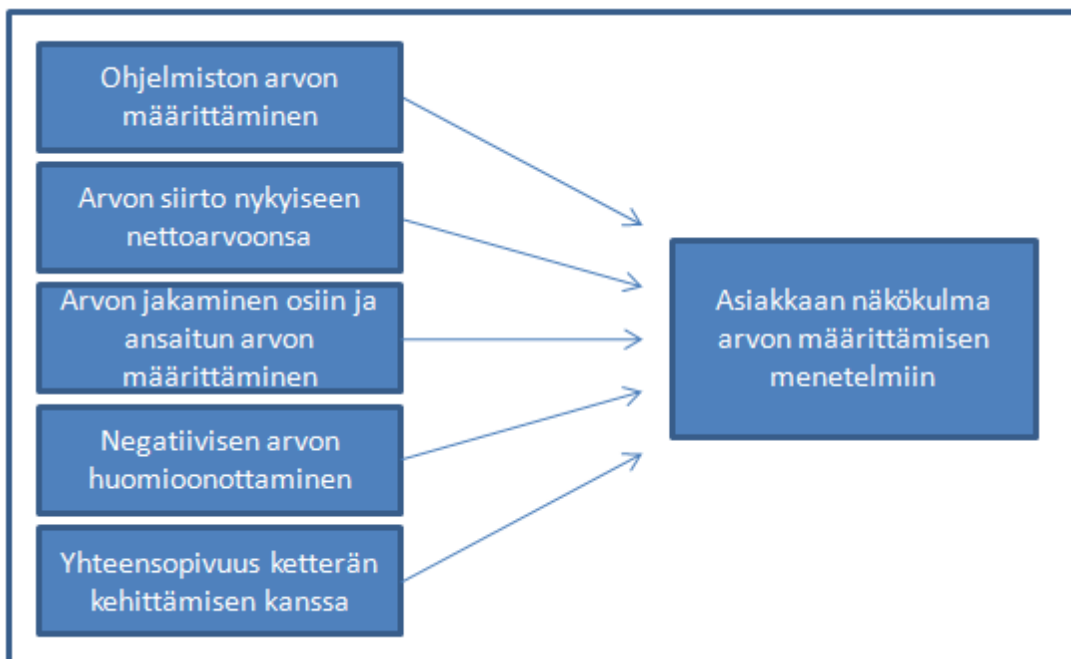
4.4.1 Arvon määrittämisen menetelmien arvioinnin kriteerit

Edellä esitetyn perusteella voidaan todeta arvon olevan ketterän kehittämisen yhteydessä hyvin moniulotteinen ja vaikeasti hahmotettava käsite. Eri lähteissä esitetyt arvon määrittämisen menetelmät lähestyvät aihetta hyvin eri tavoin ja samasta päämäärästä huolimatta käyttävät toisistaan huomattavasti poikkeavia

menetelmiä. Kokonaiskuva arvon määrittämisestä voidaan kuitenkin katsoa syntyvän ottamalla huomioon kaikki käsiteltyjen menetelmien korostamat arvon ulottuvuudet:

- ohjelmiston arvon määrittäminen
- arvon siirto nykyiseen nettoarvoonsa
- arvon jakaminen osiin ja ansaitun arvon määrittäminen
- negatiivisen arvon huomioonottaminen

Näistä ulottuvuuksista voidaan nyt johtaa asiakkaan näkökulma arvon määrittämisen menetelmien arviointiin eli toimivan menetelmän tulisi pitää sisällään kaikki neljä mainittua arvon ulottuvuutta. Näiden ulottuvuuksien lisäksi kriteeriksi määritellään yhteensopivuus ketterän kehittämisen kanssa, koska se havaittiin aiemmin mainitun mukaisesti mahdollisena ongelmana ainakin Rawsthornen (2004, 2006) menetelmän kohdalla. Näiden ulottuvuuksien muodostama asiakkaan näkökulma on esitetty kuviossa 16, ja se toimii arvon määrittämisen menetelmien vertailun teoreettisena viitekehyksenä.



KUVIO 16 Asiakkaan näkökulma arvon määrittämisen menetelmiin

Ohjelmiston arvon määrittäminen on perustavanlaatuinen lähtökohta kaikelle muulle arvon määrittämiseen liittyvälle. Aiemmin mainitun mukaisesti Rawsthornen (2006) mukaan on aluksi tärkeintä saada kehittämisen kohteen arvolle jonkinlainen yhteenlaskukelpoinen numeerinen määritelmä, joko rahamääräisenä tai suhteellisenä, jonka perusteella eri kehittämisprojektien arvoja voidaan vertailla keskenään.

Arvon siirto nykyiseen nettoarvoonsa perustuu nykyisen nettoarvon käsitteeseen. Nettoarvon laskenta merkitsee kehitys- ja muiden mahdollisten kustan-

nusten vähentämistä ohjelmiston tuottamasta hyödystä. Ohjelmiston arvon siirto nykyhetkeen puolestaan tuo esille kehitykseen kuluvan ajan merkityksen ohjelmiston arvolle ja on siksi hyvin tärkeä piirre arvon määrittämisessä. Ohjelmiston arvon siirrossa nykyhetkeen voidaan käyttää diskonttausta, mutta innovatiivisuuteen perustuvien tuotteiden markkinoilla tulee ottaa huomioon myös vaara diskonttauskurssia radikaalisti nopeammasta arvon alenemisesta, mikäli markkinoille ilmaantuu kilpailijoiden parempia tuotteita. Nykyisen nettoarvon laskentaa voidaan käyttää rahamääräisissä laskelmissa, mutta sen avulla voidaan hahmottaa myös suhteellisesti määriteltyä ohjelmiston arvoa. Tosin jälkimmäisessä tapauksessa kehityskustannusten vaikutuksen huomioiminen tuottaa ongelmia.

Arvon jakaminen osiin ja ansaitun arvon määrittäminen muodostavat yhdessä kolmannen ulottuvuuden. Nämä kaksi seikkaa käsitellään yhdessä, koska tietyllä hetkellä ansaitun arvon määrittäminen perustuu kyseisellä hetkellä toteutettuihin osiin ja tarvitsee siksi lähtökohdaksi kyseisille osille määritellyn arvon.

Negatiivisen arvon huomioonottaminen merkitsee täysin muista poikkeavaa ulottuvuutta arvon määrittämiseen. Sen mukaan positiivisen ja nolla-arvon lisäksi ominaisuuksien puuttuminen voi aiheuttaa myös negatiivista arvoa. Negatiivisen arvon käsite on tärkeää ottaa huomioon esimerkiksi vaihtoehtokustannusten vuoksi. Koska yhden ominaisuuden toteuttamiseen käytetyt resurssit ovat aina poissa jonkin toisen ominaisuuden toteuttamisesta, tulee näiden välillä tehtävän valinnan perustua toisesta saatavan positiivisen arvon lisäksi myös toisen puuttumisesta seuraavaan negatiiviseen arvoon. Negatiivisen arvon käsite on Rachevan ym. (2010a, 2010b) mainitseman käsittelyn vähäisyytensä sekä käytännön merkityksellisyytensä vuoksi hyvin mielenkiintoinen.

Yhteensopivuus ketterän kehittämisen kanssa on viimeinen arvioinnin kriteeri ja sen yhteydessä arvioidaan onko arvon määrittämisen menetelmä kaikilta osin ketterän kehittämisen periaatteiden kanssa yhteensopiva ja sen toteuttamista haittaamaton. Agile-manifestin (Agile Manifesto, 2001) mukaan ketterässä kehittämisessä arvostetaan muutokseen vastaamista enemmän kuin suunnitelman seuraamista. Ketterässä kehittämisessä sisältö siis muuttuu ja tarkentuu kehittämisen kuluessa eikä alkuvaiheessa ole sen enempää järkevää kuin mahdollistakaan tehdä sisältöön perustuvia tarkkoja laskelmia. Lisäksi Agile-manifestin (Agile Manifesto, 2001) mukaan ketterässä kehittämisessä arvostetaan toimivaa ohjelmistoa enemmän kuin kattavaa dokumentaatiota, mikä olisi ristiriidassa arvonmääritykseen liittyvän mittavan ja itse ohjelmiston toiminnan kannalta täysin ylimääräisen suunnittelutyön kanssa.

4.4.2 Arvon määrittämisen menetelmien arviointi

Määritellyn asiakkaan näkökulman ja sen sisältämien viiden kriteerin (ohjelmiston arvon määrittäminen, arvon siirto nykyiseen nettoarvoonsa, arvon jakaminen osiin ja ansaitun arvon määrittäminen, negatiivisen arvon huomioonottaminen, yhteensopivuus ketterän kehittämisen kanssa) perusteella voidaan nyt luokitella edellä käsitellyt arvon määrittämisen menetelmät niiden sisältämien

ulottuvuuksien perusteella. Luokittelu on esitetty taulukossa 11, jossa kutakin kriteeriä vastaa yksi taulukon sarake.

TAULUKKO 11 Arvon määrittämisen menetelmien luokittelu

Menetelmä	Ohjelmiston arvon määrittäminen	Arvon siirto nykyiseen nettoarvoonsa	Arvon jakaminen osiin ja ansaitun arvon määrittäminen	Negatiivisen arvon huomioonottaminen	Yhteensopivuus ketterän kehittämisen kanssa
Ansaittu arvo	Ei sisällä	Ei sisällä	Sisältää	Ei sisällä	Kyllä
Taloudellinen arvo	Ei sisällä	Sisältää	Ei sisällä	Ei sisällä	Kyllä
Liiketoiminta-arvo ja ansaittu liiketoiminta-arvo	Sisältää ehdotuksia, esim. ROI, PNL tai suhteelliset arvot	Sisältää maininnan	Sisältää	Ei sisällä	Ei automaattisesti
Suhteellinen painotus	Sisältää	Osittain	Sisältää (arvon määrittäminen tapahtuu alhaalta ylöspäin)	Sisältää	Kyllä

Luokittelun perusteella voidaan todeta kahden menetelmän kattavan suhteellisen hyvin kaikki ulottuvuudet. Nämä ovat Rawsthornen (2004, 2006) liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittämisen menetelmä sekä Wiegiersin (1999) suhteellisen painotuksen menetelmä.

Rawsthornen (2004, 2006) menetelmä sisältää toimivan arvon osiin jakamisen ja ansaitun arvon määrittämisen lisäksi ehdotuksia koko ohjelmiston arvon määrittämiseksi sekä maininnan tarpeesta siirtää kyseinen arvo nykyiseen nettoarvoonsa, mutta siitä puuttuu negatiivisen arvon huomioonottaminen. Lisäksi kyseisen menetelmän kohdalla huomioon tulee ottaa jo aiemmin mainittu menetelmän käyttämän työnositusrakenteen mahdollinen yhteensopimattomuus ketterän kehittämisen kanssa. Työnositusrakennetta voidaan käyttää myös ketterän kehittämisen yhteydessä, mutta silloin sillä voidaan määrittää vain yhdessä iteraatiossa tai julkaisussa tehtävä osa työstä.

Siinä missä Rawsthornen (2004, 2006) menetelmä lähestyy arvon määrittämistä koko ohjelmiston arvosta ja sen jakamisesta osiin eli ylhäältä alaspäin, on Wiegiersin (1999) suhteellisen painotuksen menetelmän lähestymistapa sen sijaan alhaalta ylöspäin. Suhteellisen painotuksen menetelmän käyttämä arvonmäärittäminen perustuu yksittäisten ominaisuuksien suhteellisen arvon määrittämiseen ja koko ohjelmiston arvon määrittämisen niiden yhteissummaksi. Täten voidaan katsoa arvon osiin jakamisen kuuluvan automaattisesti suhteellisen painotuksen menetelmään, koska ominaisuuksien osa kokonaisarvosta on läh-

tökohta eikä päätepiste. Suhteellisen painotuksen menetelmässä otetaan huomioon kustannukset eli määritellyt prioriteetit perustuvat nettoarvoihin, mutta arvon siirtoa nykyhetkeen menetelmässä ei käsitellä. Yhteensopivuuden ketterän menetelmän kanssa voidaan katsoa olevan hyvä, koska vaikka menetelmä on kehitetty ennen ketterän kehittämisen yleistymistä, niin se ei sisällä mitään ketterän kehittämisen piirteiden kanssa ristiriidassa olevia käytänteitä. Menetelmässä ei oteta kantaa ominaisuuksien priorisoinnin aikajänteeseen ja koska suhteellisten hyödyn, rangaistuksen, kustannusten sekä riskin määrittely ja niiden perusteella priorisoinnin laskeminen on suhteellisen vähän kuormittavaa, voidaan se helposti tehdä esimerkiksi jokaiselle sprintille tai julkaisulle uudelleen. Kaikkein mielenkiintoisinta menetelmässä on, kaikista muista menetelmistä poiketen, negatiivisen arvon olennainen osa.

Huomattavasti rajoitetumpia menetelmiä sen sijaan ovat ansaitun arvon määrittäminen ja taloudellisen arvon määrittäminen. Ansaitun arvon määrittäminen sisältää vain arvon jakamisen osiin ja niiden perusteella ansaitun arvon määrittämisen. Ansaitun arvon määrittäminen onkin määritelmänsä mukaisesti projektin etenemistä projektisuunnitelmaa vasten mittaava projektinhallinnan työkalu. Se antaa yksityiskohtaista tietoa projektin etenemisen tilanteesta sekä toteutetun työn perusteella ansaitusta arvosta suhteessa projektin kokonaisbudjettiin. Ansaitun arvon määrittäminen on kuitenkin todellakin tarkoitettu ensisijaisesti projektinhallinnan työkaluksi eikä se soveltamismahdollisuuksiltaan täysin vastaa tässä yhteydessä tavoiteltua kehitettävän ohjelmiston ja sen osien arvon määrittämistä asiakkaan kannalta. Ansaitun arvon määrittäminen perinteisessä muodossaan on myös ristiriidassa ketterän kehittämisen periaatteiden kanssa edellyttäessään mittavaa etukäteissuunnittelua ja työnositusta ja siten sisällön pysymistä kiinteänä. Sen sijaan perinteisestä ansaitun arvon määrittämisestä ketterän kehittämisen erikoistarpeisiin sovelletut versiot ovat yhteensopivia ketterän kehittämisen kanssa.

Taloudellisen arvon määrittäminen puolestaan käsittää pelkästään ohjelmiston arvon siirtämisen nykyiseen nettoarvoonsa. Se ei sisällä edes ehdotusta koko ohjelmiston arvon määrittämiseksi tai arvon jakamiseksi osiin. Nykyisen nettoarvon laskentakaavan käyttö tässä tarkastelussa käsiteltyyn arvon määrittämiseen asiakkaan näkökulmasta ei siis ole mahdollista. Nykyisen nettoarvon laskentakaava sen sijaan soveltuu muilla menetelmillä määritellyn tulevaisuuden ennustetun arvon siirtämiseen tulevaisuuden arvostaan nykyhetken arvoon ja sitä voidaan käyttää esimerkiksi Rawsthornen (2004, 2006) liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittämisen menetelmän yhteydessä. Taloudellisen arvon määrittäminen ei ota varsinaisesti kantaa sisältöön tai projektinhallintaan vaan käsittelee puhtaasti korkeamman tason taloudellisia näkökohtia, joten sen voidaan katsoa toimivan täysin eri abstraktiotasolla verrattuna ketterään kehittämiseen eikä se siksi voi olla ristiriidassa ketterän kehittämisen periaatteiden kanssa.

4.4.3 Arvon määrittämisen kokonaiskuva

Racheva ym. (2009) tiivistävät arvon käsitteen tilan melko surullisesti toteamalla kysymyksen arvon määritelmästä ketterän kehittämisen yhteydessä jäävän heidän suorittamansa systemaattisen tutkimuksen jälkeenkin avoimeksi. Edellä suoritetun tarkastelun jälkeen voimme todeta myös arvon määrittämisen menetelmien olevan hyvin vaihtelevia ja valitettavan rajoittuneita. Toimiva kokonaiskuva arvon määrittämisestä vaikuttaisi olevan saatavilla vain yhdistämällä eri lähteistä löytyneitä arvon määrittämisen menetelmiä. Tällainen kokonaiskuva on mahdollista rakentaa ottamalla eri menetelmistä käytäntöön sovellettavimmat osat.

Näin muodostuva arvon määrittämisen menetelmä perustuisi ensiksi Rawsthornen (2004, 2006) ehdotuksille ohjelmiston arvon määrittämiseksi rahamääräisen (esim. ROI tai PNL) tai suhteellisen arvon mukaisesti. Tämän jälkeen arvo tulisi siirtää nykyiseen nettoarvoonsa kehitysajan vaikutuksen huomioonottamiseksi Müllerin ja Padbergin (2003) ehdotuksen mukaisesti. Tämän jälkeen tuloksena saatu arvo jaetaan osiin käyttäen Rawsthornen (2004, 2006) menetelmää, kuitenkin soveltaen työnositurakennetta ketterän kehittämisen yhteydessä samalla tavoin kuin Alleman ym. (2003), Cabri ja Griffiths (2006) sekä Sulaiman ym. (2006) soveltavat ehdotuksissaan perinteisen ansaitun arvon määrittämisen menetelmää ketterään kehittämiseen. Työnositurakenne tehtäisiin siten projektin sijaan iteraatio- tai julkaisukohtaisesti ja työn jakamisen yksikköinä käytettäisiin testattavia vaatimuksia, käyttäjätarinoita ja tehtäviä.

Lopuksi tässä menetelmässä tulisi ottaa huomioon Wiegernin (1999) ja Rachevan (2010a, 2010b) esiin tuoma rangaistus tai negatiivinen arvo eli sen tulisi sisältää mekanismi ohjelmiston keskeisen toiminnan kannalta haittaavien puutteiden sisällyttämiseksi arvon määrittämiseen. Hyvä menetelmä tähän olisi Wiegernin (1999) menetelmän mukainen ominaisuuden suhteellisen arvon ja suhteellisen rangaistuksen yhteenlaskeminen ominaisuuden kokonaisarvon saamiseksi. Näin ollen Rawsthornen (2004, 2006) menetelmään tulisi lisätä koko ohjelmiston arvon osiin jakamisen jälkeen uusi vaihe, jossa ohjelmiston osille määritellyn arvon lisäksi niille määritettäisiin ominaisuuden puuttumisesta aiheutuva rangaistus ja tämä laskettaisiin mukaan ominaisuuden arvoon. Rachevan ym. (2010a, 2010b) huomiota puuttuvien ominaisuuksien negatiivisesta arvosta voisi olla mahdollista laajentaa puuttuvien ominaisuuksien lisäksi kattamaan myös muut ohjelmiston toiminnan kannalta haitalliset piirteet kuten erilaiset toimintaa haittaavat ohjelmointivirheet tai toiminnan kannalta epäoptimaalinen ohjelmistoarkkitehtuuri. Näiden huomioonottaminen arvon määrittelyssä tekisi arvon määrittelystä huomattavasti nykyistä kattavamman. Tällöin arvon määrittämisessä tulisivat huomioonotetuiksi myös suoranaisesti arvoa tuottamattomat, mutta kokonaisuuden kannalta tärkeät kuten esimerkiksi turvallisuuden tai toimintavarmuuteen liittyvät ominaisuudet. Negatiivista arvoa voitaisiin myös käyttää havainnollistamaan ns. teknisen velan eli nopean aikataulun vuoksi teknisesti huonolla tavalla tehtyjen ratkaisujen vaikutusta ohjelmiston arvoon.

Edellä esitetyn kaltaisen menetelmän voidaan katsoa olevan yhteensopiva ketterän kehittämisen kanssa, koska kaikki sen osat ovat yhteensopivia ketterän kehittämisen kanssa. Työnositurakenteen soveltaminen Allemanin ym. (2003), Cabrin ja Griffithsin (2006) sekä Sulaimanin ym. (2006) ehdotusten mukaisesti tekee työnositurakenteesta ja siten myös Rawshthornen (2004, 2006) menetelmästä yhteensopivan ketterän kehittämisen kanssa.

Edellä esitetty ajatus arvon määrittämisen kaikki ulottuvuudet käsittäväs-tä menetelmästä on tässä vaiheessa pelkästään teoreettisen ajatuksen tasolla eikä sitä tämän työn puitteissa ole mahdollista viedä pidemmälle, mutta kysei-sen mahdollisuuden tutkiminen olisi sopiva aihe jatkotutkimukselle.

4.5 Yhteenveto arvon määrittämisestä

Arvon käsite on keskeinen piirre ketterässä kehittämisessä, mutta tästä huoli-matta sen määritelmä ja käytännön käsittely ketterän kehittämisen yhteydessä on epäselvää ja ylimalkaista. Kirjallisuuden perusteella johdetun määritelmän mukaan ketterän kehittämisen tuottama arvo on kehittämisen tuloksena synty-neen ohjelmiston sitä käyttävälle asiakkaalle tuottama arvo asiakkaan kokema-na ja tämä arvo on yleensä subjektiivista ja kvalitatiivisesti mitattavaa.

Arvon määrittämistä käsittelevät tutkimukset jakautuvat kolmeen ryh-mään niiden käyttämän arvoon viittaavan termin perusteella. Nämä kolme ryhmää ovat ansaitun arvon määrittäminen ketterään kehittämiseen sovellettu-na, taloudellisen arvon määrittäminen sekä liiketoiminta-arvon ja ansaitun lii-ketoiminta-arvon määrittäminen. Näiden lisäksi käsiteltiin kirjallisuudesta löy-tynyttä negatiivisen arvon käsitettä ja sen merkitystä arvon määrittämisessä.

Ansaitun arvon määrittämistä käytetään projekteissa tuotetun arvon mit-taamiseksi suhteessa kustannuksiin, mutta se ei vaatimansa mittavan etukäteis-suunnittelun vuoksi sovi suoraan ketterään kehittämiseen. Taloudellisen arvon määrittäminen perustuu nykyisen nettoarvon käsitteeseen ja se havainnollistaa kehittämisessä kuluvan ajan merkitystä valmiista ohjelmistosta saatavaan voit-toon. Liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittämisen tavoit-teena on kehitettävän ohjelmiston arvon jakaminen osiin ja tietyllä hetkellä an-saitun arvon määrittäminen toteutettujen osien summana. Negatiivinen arvo puolestaan määritellään tietyn ominaisuuden puuttumisen aiheuttamana arvon alenemisena.

Arvon määrittämisen menetelmien arviointi asiakkaan näkökulmasta pi-tää sisällään kaikki esitelyjen menetelmien käsittelemät ulottuvuudet eli oh-jelmiston arvon määrittäminen, arvon siirto nykyiseen nettoarvoonsa, arvon jakaminen osiin ja ansaitun arvon määrittäminen, negatiivisen arvon huomi-onottaminen sekä lisäksi yhteensopivuuden ketterän kehittämisen kanssa. Ar-vioinnin perusteella kaikista menetelmistä löytyy selviä puutteita. Lupaavim-pina menetelminä voidaan arvioinnin perusteella pitää liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittämisen menetelmää sekä suhteellisen pai-notuksen menetelmää. Näistä ensin mainittua tulee kuitenkin mahdollisen ket-

terän kehittämisen ja työnositusrakenteen välisen ristiriidan vuoksi soveltaa esimerkiksi samoin kuin perinteistä ansaitun arvon menetelmää on esitetty sovellettavaksi ketterään kehittämiseen. Löydettyjen menetelmien arvioinnin lisäksi esiteltiin käsiteltyjen lähteiden perusteella muodostunutta arvon määrittämisen kokonaiskuvaa ja kaikki sen ulottuvuudet huomioon ottavan menetelmän hahmotelma.

5 OSALLISTUMINEN KEHITTÄMISEN AIKANA

Tässä luvussa tarkastellaan kolmatta osaa asiakkaan näkökulmasta ketterään kehittämiseen, eli osallistumista kehittämisen aikana. Ensiksi käsitellään yleisesti asiakkaan osallistumista ketterässä kehittämisessä, määritellään asiakkaan osallistuminen kehittämiseen, esitetään tavoitteita osallistumiselle sekä kuvataan osallistumista jäsentävä viitekehys. Tämän jälkeen esitellään erilaisia asiakkaan osallistumisen aktiviteetteja ja tarkastellaan kirjallisuudesta löytyviä teoreettisia arvioita sekä empiirisiä kokemuksia esitellyistä aktiviteeteista. Seuraavaksi määritellään aktiviteettien käsittelyn teoreettisena viitekehysenä toimiva asiakkaan näkökulma ja arvioidaan eri aktiviteetteja tästä näkökulmasta. Lopuksi esitetään yhteenveto.

5.1 Yleisesti asiakkaan osallistumisesta

Agile-manifestin (Agile Manifesto, 2001) mukaan ketterässä kehittämisessä arvostetaan asiakasyhteistyötä enemmän kuin sopimusneuvotteluita ja muutokseen vastaamista enemmän kuin suunnitelman seuraamista. Käytännössä näiden arvojen korostaminen ilmenee asiakkaan osallistumismahdollisuuksien merkittävänä kasvamisena perinteiseen kehittämiseen verrattuna sekä asiakkaan osallistumisen laajentumisena kattamaan koko projektin keston. Tämä on merkittävä ero verrattuna perinteiseen kehittämiseen, jossa asiakkaan osallistuminen on yleensä rajoittunut pelkästään alkuvaiheessa tapahtuvaan vaatimusten esille tuontiin ja loppuvaiheeseen tapahtuvaan hyväksymistestaukseen (Grisham & Perry, 2005).

Mumford (1983) määrittelee asiakkaan osallistumisen järjestelmäkehitykseen merkitsevän prosessia, jossa kaksi tai useampi osapuoli vaikuttaa toisiinsa selllaisten suunnitelmien, käytäntöjen tai päätösten tekemisessä, joilla on tulevaisuudessa vaikutusta kaikkiin päätösten tekemiseen osallistuneisiin tai heidän edustamiinsa sidosryhmiin. Kyseistä määritelmää tämän tutkielman aiheeseen ja sen rajoituksiin soveltaen voidaan *asiakkaan osallistuminen* tässä yhtey-

dessä määritellä prosessiksi, jossa asiakas ja toimittaja vaikuttavat toisiinsa sel-
laisissa projektiin liittyvissä päätöksissä, joilla on tulevaisuudessa vaikutusta
näihin molempiin osapuoliin.

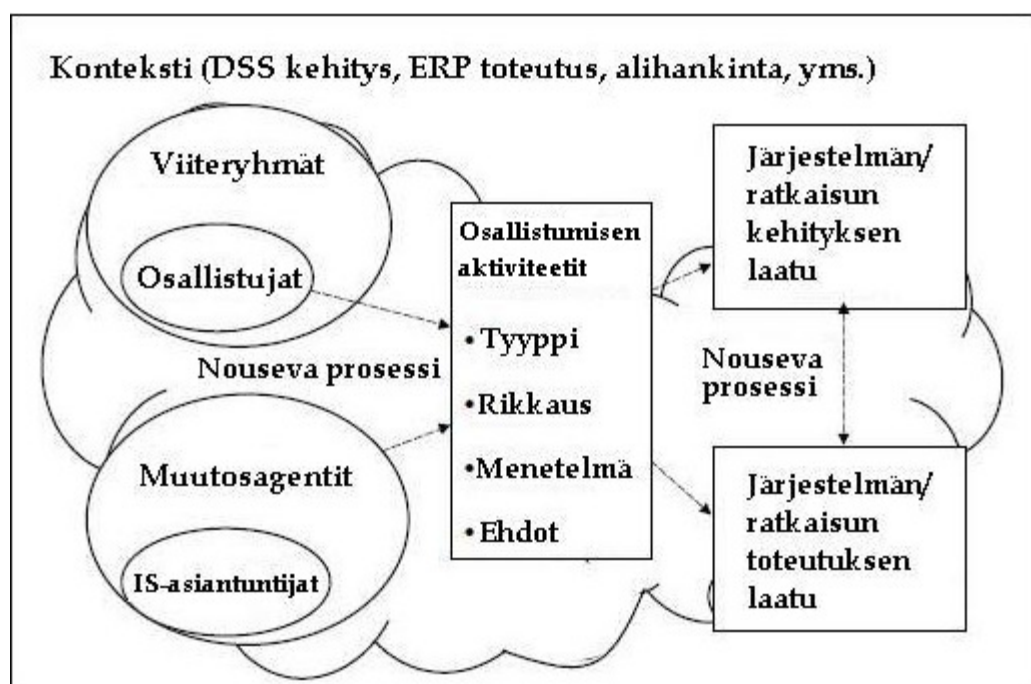
Asiakkaan osallistuminen on ketterässä kehittämisessä hyvin tärkeää ja
olennainen menestystekijä (Hoda ym., 2010). Ketterissä menetelmissä koroste-
taan asiakkaan suoraa ja säännöllistä osallistumista kehittämiseen (Grisham &
Perry, 2005). Osallistumalla tehokkaasti koko projektin ajan asiakkaalla on
mahdollisuus vaikuttaa kehitettävän ohjelmiston sisältöön ja laatuun sekä siten
sen tuottamaan arvoon. Mahdollisuuksien lisäksi asiakkaan oikeus ja velvolli-
suus osallistua merkitsee asiakkaalle kuitenkin myös runsaasti haasteita. Asi-
akkaan keskeinen rooli tarkoittaa myös suurta vastuuta ja työkuormaa. Fraserin
ym. (2004) mukaan XP-projektissa asiakkaat, olivatpa kuinka päteviä ja valmis-
tautuneita hyvänsä, kohtaavat tehtävän, joka on vaikea, aikaavievä ja kriittinen
projektin onnistumiselle.

Asiakkaan roolin vaatavuus ja sen vaatima työmäärä tulee ottaa huomioon,
ja osallistumisen hyödyt suhteessa sen vaatimiin resursseihin tulee optimoida.
Svenbrant (2008) korostaa asiakkaalle syntyvän arvoa vain silloin, kun osallis-
tumisen hyödyt ovat suurempia kuin sen asiakkaalta vaatimat uhraukset. Asi-
akkaan osallistumista tulisikin käsitellä, kuten ketterää kehittämistä yleensä,
arvontuottamisprosessina. Ketterässä kehittämisessä toimenpiteet valitaan ja
priorisoidaan niiden tuottaman arvon perusteella ja samoin myös asiakkaan
osallistumisessa toimenpiteistä tulisi päättää niiden tuottaman arvon perusteel-
la.

Asiakkaan osallistumisesta päättämiseksi tulee ensin määritellä osallistu-
misella tavoiteltava – asiakkaalle arvoa tuottava – päämäärä ja sen jälkeen mää-
ritellä tuohon päämäärään johtavat tekijät. Markus ja Mao (2004) esittelevät
kaksi päämäärää, jotka asiakkaan osallistumisella voidaan saavuttaa eli järjes-
telmän onnistuminen mitattuna ensiksikin kehityksen laadulla ja toisekseen
toteutuksen laadulla. Näistä järjestelmän kehityksen laatu merkitsee kehitettä-
vän järjestelmän tai kehityksessä käytettävän prosessin laatua. Järjestelmän to-
teutuksen laatu puolestaan merkitsee loppukäyttäjien valmistamista kehitettä-
vän järjestelmän käyttöönottoon. Vaikka tässä tutkielmassa keskitytäänkin näis-
tä kahdesta päämäärästä pääosin kehityksen laatuun eli varsinaiseen järjestel-
mään liittyviin seikkoihin, tulee kuitenkin huomata myös loppukäyttäjien val-
mistamisen olevan asiakkaan kannalta tärkeää ja sekin tulee siksi ottaa tässä
yhteydessä huomioon.

Näitä kahta päämäärää ja niihin johtavia tekijöitä on havainnollistettu ku-
viossa 17, joka esittää Markusin ja Maon (2004) teoreettisen viitekehyksen järjes-
telmäkehityksen osallistumiseen liittyvien piirteiden jäsentämiseksi. Viiteke-
hyksessä järjestelmäkehityksen toimijat on jaettu muutosagentteihin ja sidos-
ryhmiin. Näistä sidosryhmät määrittävät käsittämään niitä tahoja, joihin kehi-
tettävän järjestelmän voidaan odottaa vaikuttavan. Osallistujat (participants)
puolestaan on sidosryhmien osajoukko, jolla on mahdollisuus vaikuttaa järjes-
telmän kehitykseen, eli se ryhmä, johon tässä tutkielmassa määritelty asiakkaan
rooli kuuluu. Markusin ja Maon (2004) viitekehyksessä toimijoiden ja päämää-

rien välisenä linkkinä toimivat osallistumisen aktiviteetit (participation activities). Nämä osallistumisen aktiviteetit ovat toimijoille mahdolliset keinot vaikuttaa kehittämiseen ja siten johdattaa sitä kohti päämääriä. Lisäksi viitekehys jäsentää toimijoiden ulottuvilla olevat osallistumisen aktiviteetit neljän kriteerin mukaan, joita ovat tyyppi, rikkaus, menetelmä ja ehdot. Viitekehyksessä on myös kuvattu niin toimijoiden ja osallistumisen kuin myös osallistumisen ja lopputulosten välistä yhteyttä nousevina prosesseina, joiden tarkoituksena on osoittaa näiden vallitsevat yhteydet ja samalla kuitenkin korostaa kyseisten yhteyksien laatua pikemminkin toisiinsa vaikuttavina (influential) kuin suoranaisina vaatimuksina tai seurauksina. (Markus & Mao, 2004.)



KUVIO 17 Järjestelmäkehityksen osallistumisen teoreettinen viitekehys (Markus & Mao, 2004, s.537)

Markusin ja Maon (2004) viitekehystä ja erityisesti sen osallistumisen aktiviteeteille esittämää jäsenystä tarkastellaan myöhemmin tarkemmin, mutta sitä ennen tulee selvittää kirjallisuudesta, millaisia osallistumisen aktiviteetteja ketterässä kehittämisessä on käytettävissä. Tämän tutkielman puitteissa tarkastelu rajataan pelkästään asiakkaan osallistumisen aktiviteetteihin.

5.2 Osallistumisen aktiviteetteja

Ketterän kehittämisen asiakkaan osallistumisen aktiviteetteja koskevan kirjallisuushaun tuloksena löytyi kahdeksan aihetta käsittelevää lähdettä, joista kolme on empiirisiä ja viisi teoreettisia. Osa löydettyistä lähteistä käsittelee aktiviteetteja syvällisemmin, kun taas osa tyytyy vain mainitsemaan asiakkaan vastuut.

Löydettyjen lähteiden esittämät aktiviteetit, niitä vastaavat lähteet tyyppineen sekä kyseisissä yhteyksissä käsitellyt ketterät menetelmät on esitetty taulukossa 12. Löydetyistä kahdeksasta lähteestä kaksi käsittelee Scrumia ja kuusi XP:tä. Scrumia koskevien lähteiden osalta on käsittelyyn otettu tuoteomistajan rooli, joka on Scrumissa nimenomaan asiakkaan osallistumisen mahdollistava rooli. XP:n ylikorostuminen otannassa johtunee menetelmän alkuperäisessä versiossa mukana olleesta ja asiakkaan osallistumista korostaneesta läsnäolevan asiakkaan käytänteestä (Beck, 1999a). Kyseinen käytenne on tosin jätetty pois XP:n toisessa versiossa, mutta sen tilalla korostetaan kuitenkin yhtä merkittävästi todellista asiakkaan osallistumista (real customer involvement) (Beck & Andres, 2004, s. 61). Lisäksi XP:n ylikorostuminen otannassa kertoo tutkimuksen painotumisesta vahvasti nimenomaan XP:n käsittelyyn (Dybå & Dingsøyr, 2008).

Eri lähteissä esitetyt osallistumisen aktiviteetit eroavat huomattavasti toisistaan, ja kaikkien läpikäynti tämän tutkielman puitteissa ei olisi mahdollista. Lisäksi eri lähteissä painotetaan eri ketteriä menetelmiä, ja useat esitetyt osallistumisen aktiviteetit ovat hyvin menetelmäkohtaisia. Tämän tutkielman tarkoituksena on kuitenkin käsitellä ketterää kehittämistä yleisellä tasolla ilman menetelmäkohtaisia eroja. Näistä syistä johtuen pyritään seuraavaksi yhdistelemään eri lähteistä löytyneitä osallistumisen aktiviteetteja ja siten myös häivyttämään menetelmäkohtaisia rajoja. Yhdistettyjen osallistumisen aktiviteettien nimiksi on pyritty valitsemaan siihen kuuluvista, alun perin kirjallisuudessa mainituista aktiviteeteista parhaiten kyseistä yhdistelmää kuvaava. Osa yhdistettyjen osallistumisen aktiviteettien nimistä on jouduttu menetelmäkohtaisuuden häivyttämiseksi muotoilemaan uusiksi ja silloin on pyritty valitsemaan mahdollisimman hyvin kyseisen yhdistetyn osallistumisen aktiviteetin sisältöä kuvaava nimi. Osa yhdistetyistä aktiviteeteista on pohjimmiltaan toisiaan vastaavia, mutta ne on jätetty erilleen toisistaan eroavan osallistumisen muodon (suora/epäsuora) (Schwaber, 2004) vuoksi. Tällaisia ovat esimerkiksi kehitystiimiin kuuluminen ja kehitystiimin ohjaaminen sekä testaukseen osallistuminen ja testien tuottaminen.

Yhdistellyistä osallistumisen aktiviteeteista on lisäksi tunnistettavissa selkeitä yhteisiä teemoja, joiden perusteella ne voidaan kategorisoida ja siten muodostaa helpommin hahmotettavia kokonaisuuksia. Selkeimmin yhdistettyjen aktiviteettien kokonaisuudesta esille nousevat kategoriat ovat projektinhallintaan, sisällöstä päättämiseen, testaukseen ja hyväksyntään sekä yhteydenpitoon ja konsultointiin liittyvät. *Projektinhallinnan kategoria* käsittää projektin aloitukseen ja etenemisen seurantaan liittyvät aktiviteetit. *Sisällöstä päättämisen kategoria* puolestaan käsittää käyttäjien tarpeiden tuntemisen, käyttäjätarinoiden tuottamisen ja priorisoinnin sekä iteraatioiden ja julkaisujen suunnittelun. *Testauksen ja hyväksymisen kategoria* taas käsittää korkean tason testien kirjoittamisen ja suorituksen valvomisen sekä iteraatioiden ja valmiin ohjelmiston hyväksymisen. Lopuksi *yhteydenpidon ja konsultoinnin kategoria* käsittää yhteydenpidon eri sidosryhmiin ja niiden välisenä kontaktina toimimisen sekä erityisesti yhteydenpidon kehitystiimiin ja sen kysymyksiin vastaamisen.

TAULUKKO 12 Asiakkaan osallistumisen aktiviteetteja

Lähde	Tyyppi	Menetelmä	Osallistumisen aktiviteetti
Beck (1999a)	Teor.	XP	Läsnäoleva asiakas Metafora Suunnittelupeli Testausvetoinen kehitys Jatkuva integraatio
Van Deur- sen (2001)	Teor.	XP	Yhteydenpito loppukäyttäjiin ja loppukäyttäjien toiveiden ymmärtäminen Yhteydenpito kehittäjiin, epäselvyyksien selvittäminen ja kysymyksiin vastaaminen Toiminnallisten testien kirjoittaminen ja valvominen Iteraatioiden ja julkaisujen suunnitteluun osallistuminen Yhteydenpito johtoon ja edistymisestä raportointi
Wang, Wu & Zhao (2008)	Teor.	XP	Asiakastiimin muodostaminen Tilojen varaaminen kehitystiimille Projektisuunnitelman katselmointi Palautteen antaminen Vaatimustenhallinta Toteuttajien kysymyksiin vastaaminen Testaukseen osallistuminen Julkaisun hyväksyminen Kehitysprosessin seuraaminen Valmiin ohjelmiston testaus ja hyväksyntä
Sutherland (2008)	Teor.	Scrum	Priorisointi arvon perusteella Työmääräarvioiden hyväksyminen Sprinttien suunnittelupalaveriin osallistuminen Ominaisuuksien hyväksymisehtojen kirjoittaminen Sprinttikatselmuksiin osallistuminen
Schwa- ber & Sutherland (2011)	Teor.	Scrum	Vastuussa tuotteen arvosta Vastuussa tuotteen kehitysjonosta Sprinttien suunnittelupalaveriin osallistuminen Julkaisuista päättäminen Sprinttikatselmuksiin osallistuminen Scrum-tiimiin kuuluminen Projektinhallinta, sprinttien hallinta ja edistymisen seuranta
Griffin (2001)	Emp.	XP	Toimiminen kontaktina sidosryhmien välillä Tarinoiden kirjoittaminen Työmääräarvioiden hankkiminen Julkaisujen ja iteraatioiden sisällön päättäminen
Martin ym. (2004a)	Emp.	XP	Tarinoiden tuottaminen ja priorisointi Testauksen suorittaminen Kehitystiimiin kuuluminen
Kautz (2009)	Emp.	XP	Suunnittelupeli Käyttäjätarinat ja tarinakortit Toimivan ohjelmiston esittelyyn osallistuminen Yhteydenpito loppukäyttäjiin Hyväksymistestit Kehitystiimiin kuuluminen

Kirjallisuudesta löydetty asiakkaan osallistumisen aktiviteetit, toisiaan vastavien kanssa yhdistettyinä ja kokonaisuudesta esiin nousseiden kategorioiden mukaisesti ryhmiteltyinä on esitetty taulukossa 13.

TAULUKKO 13 Asiakkaan osallistumisen aktiviteetit yhdisteltyinä

Kategoria	Yhdistetty osallistumisen aktiviteetti
Projektinhallinta	Tilojen varaaminen kehitystiimille (Wang ym., 2008)
	Projektisuunnitelman katselmointi (Wang ym., 2008)
	Työmääräarvioiden käsittely (Sutherland, 2008; Griffin, 2001)
	Kehitysprosessin seuraaminen (Wang ym., 2008; Schwaber & Sutherland, 2011)
Sisällöstä päätäminen	Yhteydenpito loppukäyttäjiiin (Van Deursen, 2001; Wang ym., 2008; Martin ym., 2004a; Kautz, 2009)
	Tarinoiden tuottaminen ja priorisointi (Wang ym., 2008; Sutherland, 2008; Schwaber & Sutherland, 2011; Griffin, 2001; Martin ym., 2004a; Kautz, 2009)
	Iteraatioiden ja julkaisujen suunnittelu (Beck, 1999a; Van Deursen, 2001; Sutherland, 2008; Schwaber & Sutherland, 2011; Griffin, 2001; Kautz, 2009)
Testaus ja hyväksyntä	Testien tuottaminen (Beck, 1999a; Van Deursen, 2001)
	Testaukseen osallistuminen (Beck, 1999a; Wang ym., 2008)
	Testauksen suorittaminen (Martin ym., 2004a; Kautz, 2009)
	Iteraatioiden hyväksyminen (Sutherland, 2008; Schwaber & Sutherland, 2011; Kautz, 2009)
	Julkaisujen hyväksyminen (Wang ym., 2008; Kautz, 2009)
Yhteydenpito ja konsultointi	Valmiin ohjelmiston hyväksyminen (Wang ym., 2008; Sutherland, 2008)
	Kehitystiimiin kuuluminen (Beck, 1999a; Schwaber & Sutherland, 2011; Martin ym., 2004a; Kautz, 2009)
	Kehitystiimin ohjaaminen (Beck, 1999a; Van Deursen, 2001; Wang ym., 2008)
	Yhteydenpito johtoon (Van Deursen, 2001)
	Toimiminen kontaktina sidosryhmien välillä (Griffin, 2001)

Seuraavaksi käsitellään tarkemmin kutakin taulukossa 13 luetelluista yhdistetyistä osallistumisen aktiviteeteista. Aluksi perehdytään kirjallisuudesta löytyviin teoreettisiin näkemyksiin ja sen jälkeen käytännön kokemuksiin eri osallistumisen aktiviteeteista.

5.3 Teoreettisia näkemyksiä osallistumisen aktiviteeteista

Tässä alaluvussa esitellään kirjallisuudesta löytyviä teoreettisia näkemyksiä asiakkaan osallistumisen aktiviteeteista. Lähteinä käytetään osallistumisen aktiviteettien etsimisessä käytettyjä teoreettisia lähteitä, joita ovat Beck (1999a), Van Deursen (2001), Wang ym. (2008), Sutherland (2008) sekä Schwaber ja Sutherland (2011). Kyseisistä lähteistä etsitään tietoa siitä, mitä aktiviteetteja ja mi-

ten suositellaan asiakkaan suoritettavaksi sekä millaisia tuloksia niiden suorittamisesta voidaan odottaa. Osallistumisen aktiviteetit käsitellään kategorioitain alkaen projektinhallinnasta, jonka jälkeen käsitellään sisällöstä päättäminen, testaus ja hyväksyntä, sekä lopuksi konsultointi ja yhteydenpito.

5.3.1 Projektinhallinta

Wangin ym. (2008) mukaan asiakkaan tehtäviin kuuluu *kehittämiseen käytettävien tilojen varaaminen*. Tämä on tarpeellista silloin, kun kehittäjiä odotetaan työskentelevän asiakkaan luona. Tällöin asiakas ja kehittäjät voivat toimia yhdessä ja kommunikoida helposti.

Wang ym. (2008) laskevat asiakkaan tehtäviin myös *projektisuunnitelman katselmoinnin* ja siitä keskustelemisen. Projektisuunnitelmaa tulisi lisäksi ylläpitää projektin edetessä ja suunnitelmien tarkentuessa. Tämä edellyttää asiakkaan osallistumista projektisuunnitelman katselmointiin koko projektin ajan pelkän alkuvaiheen hyväksymisen sijasta.

Sutherlandin (2008) mukaan asiakkaan tulee osallistua *työmääräarvioiden käsittelyyn* hyväksymällä virallisesti kunkin sisällönosan työmääräarvio. Lisäksi kehittämisen aikana tapahtuvien sisällön muutosten osalta tulee hyväksytyt työmääräarviot liittää alkuperäisen sopimuksen liitteeksi.

Wang ym. (2008) esittävät asiakkaan vastuuksi edelleen myös *kehitysprosessin seuraamisen*. Asiakkaan tulisi olla tietoinen kehittämisen tilanteesta ja kohdatuista vaikeuksista sekä tunnistaa vaatimuksiin tehdyt muutokset ja kehittäjiä vastaukset muutoksiin.

5.3.2 Sisällöstä päättäminen

Van Deursenin (2001) mukaan asiakkaan tehtäviin kuuluu *yhteydenpito loppukäyttäjiiin* ja hän kiinnittää erityisesti huomiota tarpeeseen tasapainoilla mahdollisesti keskenään ristiriitaisten loppukäyttäjien tarpeiden välillä. Van Deursen (2001) tuo kuitenkin myös esille päätoimisen asiakkaan roolin ongelmana loppukäyttäjien ja kehittäjiä välisen suoran kontaktin puuttumisen ja tämän vuoksi asiakkaan roolissa toimivan tulisi rohkaista myös suoraa kommunikointia loppukäyttäjän ja kehittäjiä välillä sekä henkilökohtaisesti toimia tällaisen kommunikaation laadun varmistamiseksi. Wang ym. (2008) puolestaan esittävät asiakkaan tehtäviin kuuluvan asiakastiimin muodostamisen. Asiakastiimi koostuu asiakkaan edustajista ja se käsittää kehitysprojektin eri tasoilla toimivia asiakkaan edustajia. Asiakastiimi tulee muodostaa jo ennen projektin alkua ja sen pääasiallisena tehtävänä on huolehtia vaatimuksista. Asiakastiimi pitää yhteyttä loppukäyttäjiiin selvittääkseen heidän vaatimuksensa ja kehittäjiin kommunikoidakseen ja tarvittaessa selvittääkseen vaatimuksia heille. (Wang ym., 2008.)

Tarinoiden tuottaminen ja priorisointi antaa asiakkaalle mahdollisuuden päättää kehittämisen sisällöstä. Wangin ym. (2008) mukaan asiakkaan tehtävänä on esittää vaatimuksia kunkin iteraation alussa ja priorisoida ne sekä päättää

mitkä muutokset ovat niiden tekemiseen tarvittavien resurssien arvoisia. Sutherland (2008) määrittelee asiakkaan vastuuksi ominaisuuksien priorisoinnin arvon perusteella ja niiden toteuttamisen arvon mukaisessa järjestyksessä. Schwaber ja Sutherland (2011) puolestaan määrittelevät Scrumin tuoteomistajan pääasialliseksi vastuuksi tuotteen arvosta huolehtimisen. Tuoteomistaja on yksinomaisesti vastuussa tuotteen kehitysjonosta eli sen sisällöstä, priorisoinnista, näkyvyydestä kaikille ja ymmärrettävyydestä tai tarpeen vaatiessa lisäselvityksistä. Lisäksi tuoteomistaja on vastuussa kehitystiimin työn ohjaamisesta tuotteen arvoa lisääväksi sekä sprinttien ja julkaisujen hallinnasta. Vastuun lisäksi tuoteomistajalla on yksinomainen valta hänelle kuuluvissa tehtävissä. (Schwaber & Sutherland, 2011.)

Iteraatioiden ja julkaisujen suunnitteluun liittyy XP:n suunnittelupeli-käytänne. Beck (1999a) käsittelee suunnittelupeli-käytänteen esittelyn yhteydessä siihen kuuluvia liiketoimintahenkilöiden (business people) vastuita, jotka voidaan tässä yhteydessä käsittää koskemaan nimenomaan asiakkaan vastuita. Beckin (1999a) mukaan liiketoimintahenkilöiden eli asiakkaan vastuisiin kuuluu päättää sisällöstä, prioriteeteista, julkaisujen koostumuksesta ja julkaisujen ajankohdista. Sisällöstä päättämiseen kuuluu vastaaminen kysymykseen siitä, kuinka paljon toiminnallisuutta ohjelmiston tulee sisältää voidakseen tuottaa todellista arvoa. Prioriteeteista päättämiseen kuuluu valitseminen eri vaihtoehtojen välillä, joista kaikkia ei ole mahdollista toteuttaa. Julkaisujen sisällöstä päättämiseen kuuluu vastaaminen kysymykseen siitä, kuinka paljon tai kuinka vähän tulee olla tehtynä, jotta ohjelmisto voi edistää liiketoimintaa. Ja lopuksi julkaisujen ajankohdista päättämiseen kuuluu ohjelmistojulkaisun olemassaolon kannalta merkittävien ajankohtien tunnistaminen. (Beck, 1999a.) Schwaberin ja Sutherlandin (2011) mukaan Scrumissa tuoteomistaja osallistuu sprinttien suunnittelupalaveriin ja esittelee tuotteen kehitysjonon, joka toimii pohjana sprintin suunnittelulle. Samoin myös Sutherlandin (2008) mukaan asiakkaan tehtävänä on Scrumissa osallistua sprinttien suunnittelupalaveriin.

5.3.3 Testaus ja hyväksyntä

Testien tuottamiseen liittyen Beck (1999a) mainitsee XP:n testauskäytänteen käsittelyn yhteydessä funktionaalisten testien kirjoittamisen kuuluvan asiakkaan vastuulle.

Testaukseen osallistumisesta puolestaan Wang ym. (2008) mainitsevat, että asiakkaan tehtävänä on ottaa osaa käyttäjätestaukseen ja koekäyttöön. Asiakkaan tulee tuottaa testitapauksia todellisen ympäristön perusteella, osallistua integraatiotestaukseen, koekäyttää julkaisuja ohjelmiston kehitysvaiheessa sekä koekäyttää lopullista julkaistua versiota ohjelmistosta rajoitetusti ja toimittaa koekäytön tulokset mahdollisimman nopeasti. Lisäksi asiakkaan tulee kehittämisen kuluessa osallistua jatkuvaan integraatioon, jossa asiakkaan tehtävänä on päättää kunkin uuden ohjelmiston osan kohdalla sen mahdollisesta ohjelmistoon mukaan ottamisesta. (Wang ym., 2008.)

Iteraatioiden hyväksymiseen liittyy Scrumissa sprinttien katselmointi. Sutherland (2008) esittää asiakkaan tehtäväksi Scrumissa osallistumisen sprinttikatselmuksiin ja palautteen antamisen valmistuneesta sekä myös keskeneräisestä työstä. Schwaberin ja Sutherlandin (2011) mukaan Scrumin tuoteomistaja toteaa sprinttikatselmuksessa sprintin aikana valmistuneet tehtävät ja päättää sprintin aikana syntyneen uuden ohjelmiston inkrementin mahdollisesta julkaisusta.

Wang ym. (2008) esittävät asiakkaan vastuuksi *julkaisun hyväksymisen*. Julkaisu voidaan toimittaa asiakkaalle käyttäjätestaukseen ja koekäyttöön. Asiakas hyväksyy julkaisun, mikäli se vastaa julkaisusuunnitelmassa sovittua.

Wangin ym. (2008) mukaan asiakkaan vastuisiin kuuluu *valmiin ohjelmiston hyväksyminen*. Asiakkaan tai asiakastiimin tulee tarkistaa valmiin ohjelmiston vastaavuus esitettyihin vaatimuksiin. Vaatimuksia vastaava ohjelmisto voidaan tämän jälkeen hyväksyä. Sutherlandin (2008) mukaan puolestaan asiakkaan vastuisiin kuuluu kunkin ominaisuuden hyväksymisehtojen kirjoittaminen, jotta asiakkaalla ja kehitystiimillä on yhtenevä käsitys ominaisuuksista.

5.3.4 Yhteydenpito ja konsultointi

Beck (1999a) korostaa *kehitystiimiin kuulumisen* tarvetta XP:hen kuuluvan läsnäolevan asiakkaan käytännön käsittelemisen yhteydessä. Todellisen asiakkaan tulee työskennellä fyysisesti yhdessä kehitystiimin kanssa valmiina vastaamaan kysymyksiin ja selvittämään epäselvyyksiä. Ongelmana tämän suhteen voi olla mahdollisuus irrottaa todellinen asiakas työstään kehittämiseen osallistumisen ajaksi.

Kehitystiimin ohjaaminen merkitsee kehitystiimin tukemista, kuulumatta kuitenkaan itse siihen ja osallistumatta sen jokapäiväiseen työhön välittömän tuen tarpeen täyttämistä enempää. Van Deursenin (2001) mukaan asiakkaan tehtäviin kuuluu kehittäjien kanssa keskustelu ja ominaisuuksien selvittäminen tarvittaessa sekä myös kehittäjien teknisten huolien ymmärtäminen. Wang ym. (2008) puolestaan määrittävät asiakkaan tehtäviin kuuluvan palautteen antamisen kehitystiimille sekä konsulttina toimimisen kehittäjien kysymyksiin vastaamiseksi. Kehitystiimin ohjaamiseen liittyy myös Beckin (1999a) käsittelemä XP:n metafora-käytänne, jonka ajatuksena on asiakkaan ajatuksissa sijaitsevan vision siirtäminen kehittäjille metaforan välityksellä.

Van Deursenin (2001) mukaan asiakkaan tulee huolehtia *yhteydenpidosta johtoon*. Asiakkaan tulee kommunikoida johdon kanssa kehittämisen edistymisestä ja esittää oikeutus kehitystiimin parissa käyttämälleen ajalle (Van Deursen, 2001). Johdolla tarkoitetaan tässä selvästikin asiakkaan esimiestasoa tai muuta asiakkaan organisaation johtoa.

5.4 Käytännön kokemuksia osallistumisen aktiviteeteista

Tässä alaluvussa esitellään kirjallisuudesta löytyviä käytännön kokemuksia eri osallistumisen aktiviteeteista. Lähteinä toimivat Griffinin (2001), Martinin ym. (2004a) sekä Kautzin (2009) tutkimukset. Kyseisistä tutkimuksista etsitään tietoa siitä, mitä aktiviteetteja ja miten asiakas on kussakin tapauksessa suorittanut ja millaisin kokemuksin. Osallistumisen aktiviteetit käsitellään kategorioittain alkaen projektinhallinnasta, jonka jälkeen käsitellään sisällöstä päättäminen, testaus ja hyväksyntä sekä lopuksi konsultointi ja yhteydenpito. Lisäksi näiden jälkeen on käsitelty erikseen ne aktiviteetit, joiden vaatimasta resurssien tarpeesta on kirjallisuudesta löydettävissä erityisiä havaintoja. Tarkastelu tässä alaluvussa on varsin suppeaa johtuen aiheesta tehtyjen empiiristen tutkimusten vähäisyydestä. Samantapaista tarkastelutapaa voidaan kuitenkin käyttää myös tulevien tapaustutkimusten analysointiin.

5.4.1 Projektinhallinta

Griffinin (2001) käsittelemässä tapauksessa käsitellään *työmääräarvioiden käsittelyä*. Korkean tason työmääräarvioiden saamisen kerrotaan olleen aluksi haastavaa, koska kehittäjät eivät ymmärtäneet kehitettävän järjestelmän kokonaisuutta eivätkä siksi osanneet hahmottaa ja arvioida yksittäisiä käyttäjätarinoita. Yksittäisten tehtävien työmääräarvioiden tuottamisen kerrotaan samoin olleen aluksi hyvin haastavaa. Toisen iteraation alussa kehittämisen sisällön kerrotaan yhtäkkiä kokeneen täydellisen muutoksen, jonka seurauksena oli tarve saada uusia korkean tason työmääräarvioita hyvin nopeasti. Nopeasti tehdyt korkean tason työmääräarviot paljastuivat myöhemmin hyvin suuriksi verrattuna tarkempiin tehtävätason työmääräarvioihin, mikä aiheutti ongelmia iteraatioiden sisällön suhteen. (Griffin, 2001.)

5.4.2 Sisällöstä päättäminen

Yhteydenpito asiakkaisiin mainitaan Martinin ym. (2004a) käsittelemistä tapauksista toisen kohdalla, jossa tuodaan ilmi asiakkaan rooli jopa tuhansien loppukäyttäjien edustajana. Samoin Martinin ym. (2004a) käsittelemistä tapauksista kolmannen kohdalla mainitaan asiakkaan tuotepäälliköiden pitäneen yhteyttä sisäisten ja ulkoisten sidosryhmien kanssa ymmärtääkseen heidän tarpeitaan. Myös Kautzin (2009) käsittelemässä tapauksessa asiakkaan edustajina toimineet projektipäälliköt korostivat rooliinsa kuuluneen yhteydenpidon loppukäyttäjiiin ja vaatimusten tulleen heidän itsensä sijaan nimenomaan loppukäyttäjiltä. Asiakkaan edustajina toimineiden projektipäälliköiden kerrotaan vierailleen loppukäyttäjien luona tavoitteenaan kerätä palautetta, ideoita ja parannusehdotuksia. Tämän käytännön kerrotaan auttaneen huomaamaan mahdolliset väärinkäsitykset ja vähentäneen niistä johtuneita ongelmia. (Kautz, 2009.)

Tarinoiden tuottamiseen ja priorisointiin liittyen Kautzin (2009) käsittelemässä tapauksessa kerrotaan kehittäjien tuottaneen tarinakortit sekä priorisoineen ne yhdessä asiakkaiden kanssa. Samoin Martinin ym. (2004a) käsittelemistä tapauksista ensimmäisen kohdalla kerrotaan asiakkaan sijasta kehitystiimin kirjoittaneen käyttäjätarinat toiminnallisen määrittelyn perusteella, koska asiakas ei olisi kyennyt tuottamaan riittävää määrää käyttäjätarinoita riittävän nopeasti. Martin ym. (2004a) arvelevat tämän johtuneen joko asiakkaan ajanpuutteesta tai osaamattomuudesta tuottaa käyttäjätarinoita. Martinin ym. (2004a) käsittelemistä tapauksista toisen kohdalla sen sijaan mainitaan, ettei sisällön päättäminen ollut vaikeata toimialueensa vahvasti tuntevalle asiakkaalle. Sen sijaan kyseisessä tapauksessa ongelmana mainitaan olleen sopivan yksityiskohtaisuuden tason löytäminen käyttäjätarinoihin ja todetaan olleen vaikeata vetää rajaa kehittäjien suunnittelua varten riittävän yksityiskohtaisuuden ja toisaalta taas kehittäjien suunnittelupäätöksiin vaikuttavan yksityiskohtaisuuden tason välillä (Martin ym., 2004a). Samoin myös Griffinin (2001) käsittelemässä tapauksessa mainitaan alussa ongelmaksi muodostuneen asiakkaan epätietoisuus käyttäjätarinoiden sopivasta yksityiskohtaisuuden tasosta työmääräarvioita varten. Kyseisessä tapauksessa käyttäjätarinat olivatkin aluksi huonolaatuisia, koska ne eivät olleet riittävän yksityiskohtaisia tai loppuun asti mietittyjä. Kolmanteen iteraatioon mennessä käyttäjätarinoiden laatu oli kuitenkin parantunut merkittävästi. Työmääräarvioiden epätarkkuuden vuoksi käyttäjätarinoita päädyttiin kuitenkin varmuuden vuoksi tuottamaan arvioitua tarvetta suurempi määrä. (Griffin, 2001.)

Martinin ym. (2004a) käsittelemistä tapauksista kolmannen tapauksen kohdalla kerrotaan asiakkaan edustajina toimineiden tuotepäälliköiden aluksi koonneen priorisoidun listan tarvittavista ominaisuuksista olemassa olleen järjestelmän sekä loppukäyttäjien tarpeiden perusteella ja tämän jälkeen asiakkaan tuotepäälliköiden tuottaneen käyttäjätarinat ominaisuuslistan pohjalta. Martinin ym. (2004a) käsittelemistä tapauksista ensimmäisessä kerrotaan käyttäjätarinoiden priorisoinnin - eli erään asiakkaan päätehtävistä - olleen asiakkaan itsensä mukaan hyvin vaikeata siihen sisältyvän suuren vastuun vuoksi. Myös Martinin ym. (2004a) käsittelemistä tapauksista toisessa mainitaan ongelmana priorisointi ja siinä nimenomaisesti liian suuri keskittyminen alkuvaiheessa teknisiin seikkoihin varsinaisten liiketoiminnallisten ominaisuuksien sijasta. Sen sijaan Griffinin (2001) käsittelemässä tapauksessa kerrotaan korkean tason työmääräarvioiden saamisen jälkeen priorisoinnin olleen asiakkaana toimineille tuotepäälliköille suhteellisen helppoa. Martinin ym. (2004a) tutkimuksen käsittelemistä tapauksista ensimmäisen kohdalla mainitaan asiakkaan olleen yleisesti ottaen hyvin tyytyväinen mahdollisuuteen muokata vaatimuksiaan kehittämisen kuluessa ja tiedon lisääntyessä

Iteraatioiden ja julkaisujen suunnitteluun liittyen Kautz (2009) kertoo käsittelemässään tapauksessa toimittajan projektipäälliköiden suunnitelleen kunkin iteraation sisällön suunnittelupelin muodossa perustuen kehitettävän järjestelmän konseptiin ja vaatimuslistaan, jotka myös olivat toimittajan projektipäälliköiden yhdessä asiakkaan kanssa tuottamia. Griffinin (2001) käsittelemässä ta-

pauksessa mainitaan ongelmaksi toisen iteraation jääminen työmäärältään va-
jaaksi, koska asiakkaat eivät harhaanjohtavien työmääräarvioiden vuoksi olleet
tuottaneet riittävää määrää tarinoita iteraatiota varten. Griffin (2001) kertoo
suunnittelun helpottuneen kolmannen iteraation kohdalla ja suunnitteluun käy-
tetyn ajan lyhentyneen merkittävästi.

5.4.3 Testaus ja hyväksyntä

Testauksen suorittamista käsittelevät Martin ym. (2004a), joiden tutkimista tapa-
uksista ensimmäisen ja toisen kohdalla korostetaan asiakkaan suorittaneen hy-
väksymistestauksen ja kolmannen tapauksen kohdalla mainitaan asiakkaan
suorittaneen lopullisen valmiin tuotteen järjestelmä- ja integraatiotestauksen.
Kaikkien kolmen tapauksen yhteydessä mainitaan asiakkaan apuna olleen joko
testausasiantuntija tai kokonainen testaustiimi (Martin ym., 2004a). Tässä yh-
teydessä viitataan Van Deursenin (2001) käsittelemään tarpeeseen yhdistää asi-
akkaan roolissa toimivan avuksi erillinen laadunvarmistustiimi (quality assu-
rance team), jonka vastuulla on tarinoiden hyväksymistestien tuottaminen ja
suorittaminen, ympäristön ylläpitäminen automaattisten hyväksymistestien
ajamiseksi sekä laadunvarmistustoimien toteutuksen valvominen. Toisen tapa-
uksen kohdalla käytössä oli automatisoitu hyväksymistestauskehys, joka mah-
dollisti toimialueen hyvin tunteneiden ja ohjelmiston toteuttaman algoritmin
suunnitteleiden tutkijoiden osallistumisen algoritmien hyväksymistestaukseen
iteraatiokohtaisesti. Asiakkaan nopeasti suorittaman hyväksymistestauksen
kerrotaan tässä tapauksessa korottaneen tuotteen laatutasoa merkittävästi.
Kolmannessa tapauksessa puolestaan kerrotaan asiakkaan edustajien jokaisen
päivän päätteeksi katselmoineen ja testanneen jokaisen toteutetun käyttäjätari-
nan, jonka jälkeen hyväksytyt tarinat merkattiin valmistuneiksi. (Martin ym.,
2004a.) Kautzin (2009) käsittelemässä tapauksessa hyväksymistestien suoritta-
misen kerrotaan olleen asiakkaan vastuulla ja hyväksymistestausta suoritta-
massa olleen aina loppukäyttäjiiin kuuluvia. Asiakkaan edustajina toimineiden
projektipäälliköiden kerrotaan olleen tyytyväisiä kyseiseen osallistumisen muo-
toon ja kokeneen itsensä osallisiksi kehittämiseen.

Iteraatioiden hyväksyminen tunnistetaan erääksi asiakkaan osallistumisen
muodoista Kautzin (2009) tutkimuksessa. Kyseisen tutkimuksen käsittelemässä
tapauksessa kerrotaan kunkin julkaisun koostuneen 3-6 viikkoa kestävästä ite-
raatioista, joiden aikaansaannoksia esiteltiin asiakkaalle iteraation jälkeen tai
jopa jo sen aikana. Kahden iteraation välissä kerrotaan olleen päättyneeseen
iteraatioon jälkiaktiiviteettina kuulunut testausvaihe.

Kautz (2009) mainitsee asiakkaan osallistuneen myös *julkaisujen hyväksy-
miseen*. Julkaisuja kerrotaan tehdyn 3-6 kuukauden välein ja kukin julkaisu esi-
teltiin suuremmalle käyttäjistä koostuneelle ryhmälle, koska esityksiä yksittäis-
selle asiakkaan edustajalle ei pidetty riittävinä.

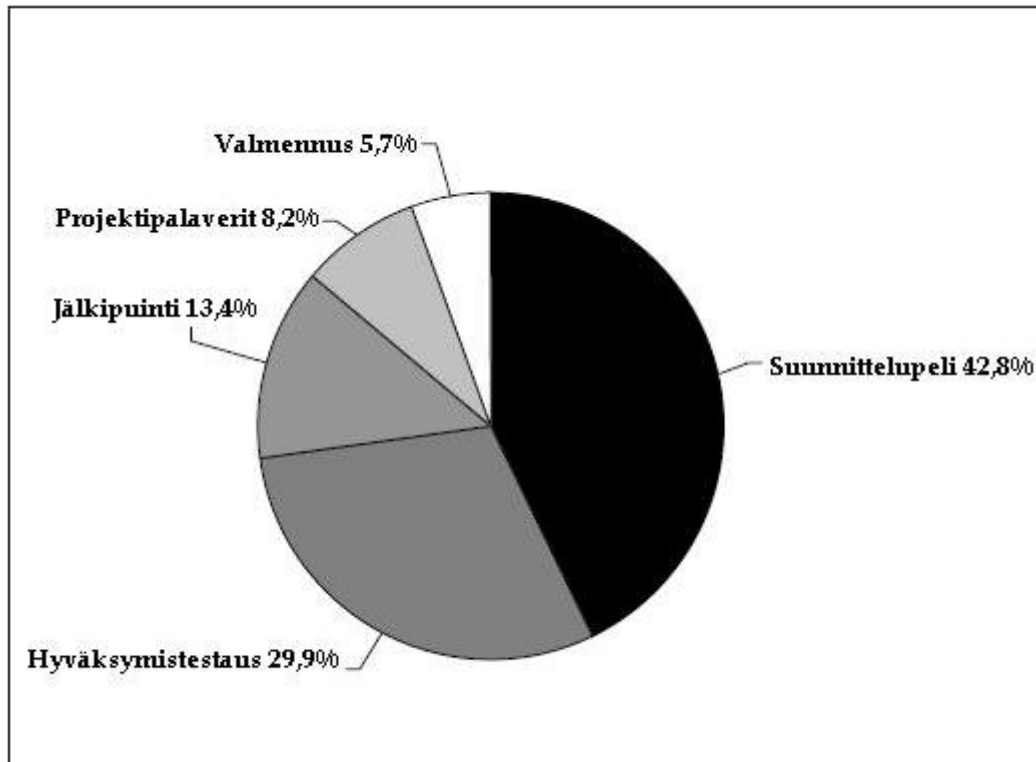
5.4.4 Yhteydenpito ja konsultointi

Kehitystiimiin kuulumisen tuodaan esille Martinin ym. (2004a) käsittelemistä tapauksista toisen kohdalla. Siinä XP:n läsnäolevan asiakkaan roolin kuvaillaan koostuneen läheisestä yhteistyöstä kehitystiimin kanssa. Jatkuvan läheisen yhteistyön kerrotaan kuitenkin vieneen hyvin paljon aikaa ja asiakkaan kerrotaan käytännössä joutuneen suorittamaan mahdolliset muut tehtävät normaalin työajan ulkopuolella kehittäjiä neuvomisen kuluttaessa normaalin työajan kokonaan. Kautzin (2009) käsittelemässä tapauksessa asiakkaan edustajina toimivat projektipäälliköt puolestaan kuvasivat työtään konsultointina, jossa he ohjasivat kehittämistä ja olivat kehittäjiä käytettävissä tarvittaessa. Asiakkaan projektipäälliköt arvioivat, että asiakkaan panosta tarvittiin lisäselvityksen muodossa lähes jokaisen tarinakortin osalta. Tarinakorteista arviolta 20% vaati lisäselvitystä kesken iteraation ja toinen 20% iteraation lopussa, mikä korostaa toimivan ohjelmiston merkitystä väärinkäsitysten havaitsemisessa. Sekä toimittajan että asiakkaan projektipäälliköt korostivat sen tärkeyttä, että nimenomaan asiakas todella päättää kehittämisestä ja sisällöstä. Toimittajan tehtävänä ei ole tehdä päätöksiä asiakkaan puolesta vaikka asiakas niin jossain tapauksessa olettaisikin. Kehittäjiä puolelta mainittiin asiakkaan kiinteän osallistumisen ja viikoittaisen palautteen antamisen huonona puolena sen aiheuttama jatkuva pienten muutospyyntöjen virta. (Kautz, 2009.)

5.4.5 Resurssien tarve

Martin ym. (2004a) tuovat kaikkien kolmen käsittelemänsä tapauksen yhteydessä esille *kehitystiimiin kuulumisen* kuormittavuuden asiakkaalle. Martinin ym. (2004a) käsittelemistä tapauksista ensimmäisessä tuodaan ilmi asiakkaan roolin merkittävyys sekä sen vaatima aika. Toisessa tapauksessa tuodaan ilmi jo yhteydenpidon ja konsultoinnin yhteydessä mainittu koko normaalin työajan kulumisen kehittäjiä neuvomiseen ja muiden työtehtävien jääminen sen vuoksi normaalin työajan ulkopuolella suoritettaviksi. Kolmannessa tapauksessa puolestaan ongelmana mainitaan jatkuvien iteraatioiden asiakkaan edustajille aiheuttama suuri työkuorma sekä heidän merkittävän roolinsa vuoksi kärsimä henkinen paine. (Martin ym., 2004a.)

Aktiviteettien resurssien tarpeeseen liittyy myös Koskelan ja Abrahamssonin (2004) empiirisessä tutkimuksessaan saavuttama tulos asiakkaan työmäärän jakautumisesta, joka on esitetty kuviossa 18. Kuvioista selviää suunnittelupelin olleen käsitellyssä tapauksessa merkittävästi eniten asiakasta työllistänyt aktiviteetti, hyväksymistestauksen tullessa selvällä erolla toisena ja muiden aktiviteettien yhteenlasketun työmäärän ollessa edelleen hieman testausta pienempi. Näin ollen voidaan todeta suunnittelupeliin sisältyvien aktiviteettien eli tarinoiden tuottamisen ja priorisoinnin sekä iteraatioiden ja julkaisujen suunnittelun olevan resurssien tarpeeltaan merkittäviä.



KUVIO 18 Asiakkaan työmäärän jakautuminen (Koskela & Abrahamsson, 2004, s.8)

5.5 Osallistumisen aktiviteettien arviointi asiakkaan näkökulmasta

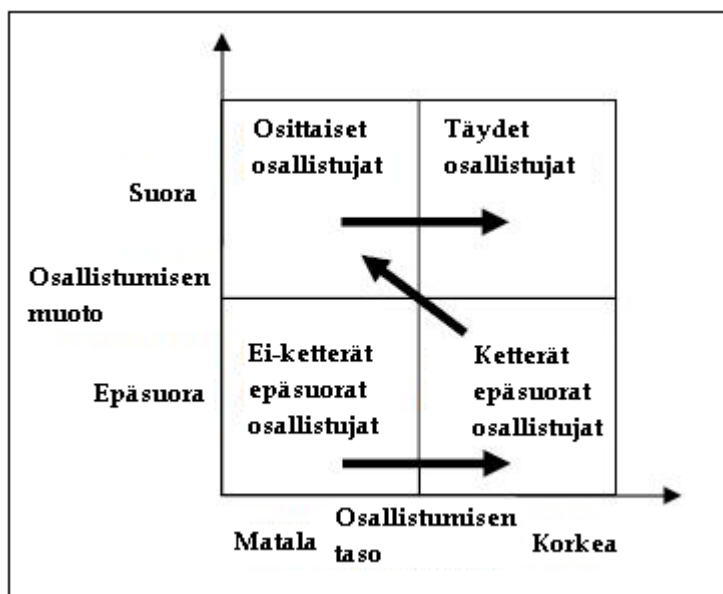
Tässä alaluvussa arvioidaan aiemmin esiteltyjä osallistumisen aktiviteetteja asiakkaan näkökulmasta. Aluksi esitellään osallistumisen aktiviteettien arvioinnin kriteerit ja niiden muodostama asiakkaan näkökulma. Sen jälkeen käytetään esiteltyjä kriteerejä arvioitaessa osallistumisen aktiviteettien suorittamisen hyödyllisyyttä asiakkaalle.

5.5.1 Osallistumisen aktiviteettien arvioinnin kriteerit

Osallistumisen aktiviteettien arviointikriteereiden lähteinä toimivat Markusin ja Maon (2004) sekä Svenbrantin (2008) tutkimukset. Markusin ja Maon (2004) teoreettinen viitekehys jäsentää asiakkaan ulottuvilla olevat osallistumisen aktiviteetit neljän kriteerin mukaan, joita ovat tyyppi, rikkaus, menetelmä ja ehdot. Näistä kaksi ensimmäistä koskee osallistujan näkökulmaa ja kaksi jälkimmäistä muutosagentin näkökulmaa (Markus & Mao, 2004). Näin ollen asiakkaan näkökulmasta merkityksellisiä kriteereitä ovat kaksi ensimmäistä eli tyyppi ja rikkaus.

Svenbrant (2008) puolestaan esittää Scrum-asiakkaiden luokittelujärjestelmän kuviossa 19 esitetyn mukaisesti. Hänen mukaansa Scrum-projektien asiakkaat voidaan jakaa kahteen ryhmään osallistumisen muodon (suo-

ra/epäsuora) ja edelleen kahteen ryhmään osallistumisen tason (korkea/matala) perusteella, jolloin saadaan kokonaisuudessaan neljä osallistumisen perusteella toisistaan eroavaa asiakkaiden ryhmää.



KUVIO 19 Scrum-asiakkaiden luokittelujärjestelmä (Svenbrant, 2008, s.3)

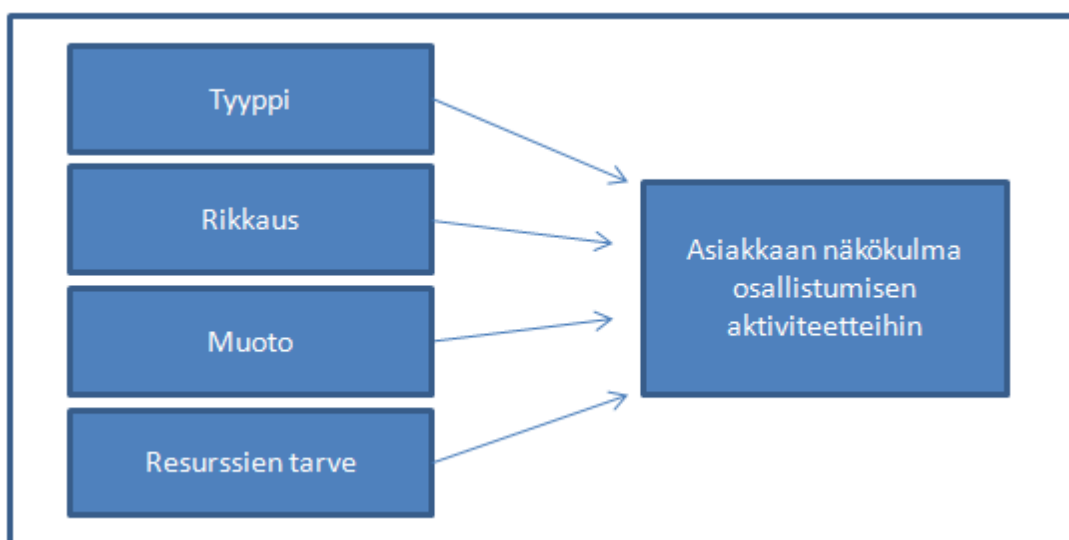
Svenbrantin (2008) esittämistä kahdesta asiakkaan osallistumisen aktiviteetteja erottavasta kriteeristä osallistumisen taso vastaa Markusin ja Maon (2004) esittämää osallistumisen rikkautta. Svenbrantin (2008) esittämistä kriteereistä toinen eli osallistumisen muoto on sen sijaan lisäys Markusin ja Maon (2004) esittämiin nähden ja se voidaan ottaa käyttöön kolmantena asiakkaan osallistumisen aktiviteettien arvioinnin kriteerinä.

Koska aktiviteettien arvioinnin tarkoituksena on auttaa asiakasta löytämään optimaalinen tasapaino aktiviteettien suorittamisen vaatimien resurssien ja niiden tuottaman arvon välillä ja opastaa asiakasta valitsemaan vaadittavien resurssien ja tuotettavan arvon suhteen kannalta tehokkaimmat aktiviteetit, tulee myös aktiviteettien suorittamisen vaatimat resurssit ottaa arvioinnissa huomioon. Tästä syystä arviointiin otetaan mukaan neljäntenä kriteerinä aktiviteettien suorittamisen asiakkaalta vaatimat resurssit.

Edellä mainitut neljä seikkaa eli tyyppi, rikkaus, muoto ja resurssien tarve toimivat kriteereinä osallistumisen aktiviteettien arviointiin asiakkaan näkökulmasta. Näiden kriteereiden muodostama asiakkaan näkökulma on esitetty kuviossa 20 ja se toimii osallistumisen aktiviteettien arvioinnin teoreettisena viitekehystenä.

Tyyppin perusteella osallistumisen aktiviteetit voidaan Markusin ja Maon (2004) mukaan jakaa suunnitteluun liittyviin, toteutukseen liittyviin sekä projektinhallintaan liittyviin aktiviteetteihin. Suunnitteluun liittyvät aktiviteetit pitävät sisällään kaiken suoranaisesti järjestelmän laatuun vaikuttavan osallistumisen. Toteutukseen liittyvät aktiviteetit puolestaan pitävät sisällään kaiken järjestelmän käyttöönottoon ja loppukäyttäjien valmistamiseen liittyvän osallis-

tumisen. Ja lopuksi projektinhallintaan liittyvät aktiviteetit pitävät sisällään kaiken projektin valmistumiseen ja kustannustehokkuuteen tai aikatauluun vaikuttavan. (Markus & Mao, 2004). Tässä työssä tarkennetaan tätä Markusin ja Maon (2004) esittämää kolmijakoista tyyppijäsennystä tavalla, joka perustuu aiemmin esitettyyn osallistumisen aktiviteettien kategorisointiin. Aktiviteetit jaetaan tyyppin perusteella *projektinhallintaan* liittyviin (projektin aloitus ja etenemisen seuranta), *sisällöstä päättämiseen* liittyviin (käyttäjien tarpeiden tunteminen, käyttäjätarinoiden tuottaminen ja priorisointi sekä iteraatioiden ja julkaisujen suunnittelu), *testaukseen ja hyväksyntään* liittyviin (korkean tason testien kirjoittaminen ja suorituksen valvominen sekä iteraatioiden ja valmiin ohjelmiston hyväksyminen) sekä *yhteydenpitoon ja konsultointiin* liittyviin (yhteydenpito eri sidosryhmiin ja niiden välisenä kontaktina toimiminen sekä erityisesti yhteydenpito kehitystiimiin ja heidän kysymyksiinsä vastaaminen).



KUVIO 20 Asiakkaan näkökulma osallistumisen aktiviteetteihin

Rikkaus voidaan selittää osallistumisen syvyytenä, ja se pitää sisällään osallistumisen merkittävyyden ja mahdollisuuden todelliseen vaikuttamiseen. Sen perusteella voidaan erottaa toisistaan esimerkiksi vaatimuksia koskevaan lyhyeen kyselyyn vastaaminen tai kokoaikainen vaatimustenhallintatiimissä toimiminen. Jälkimmäisessä tapauksessa sekä osallistujan henkilökohtainen panos että mahdollisuus vaikuttaa järjestelmän laatuun ovat huomattavasti suurempia. Lisäksi toimiminen päätäntämahdollisuuksia omaavassa roolissa tuottaa rikkaamman osallistumiskokemuksen kuin toimiminen pelkästään toteuttavassa roolissa. (Markus & Mao, 2004.) Myös Hanssenin ja Fægriin (2006) mukaan asiakkaan osallistuminen voi tapahtua eri tasoilla. Asiakkaat voivat *olla mukana* (involved) kehittämisessä, mutta vain ajatustasolla, jolloin heille ei kenties ole osoitettu tarkkoja tehtäviä tai vastuuta. Toisaalta asiakkaat voivat *ottaa osaa* (participation) kehittämiseen, jolloin heille on määrätty tietyt tehtävät tai vastuut. (Hanssen & Fægri, 2006.)

Muodon perusteella asiakkaan osallistuminen voidaan jakaa suoraan ja epäsuoraan osallistumiseen. *Suoraan* osallistuva asiakas johtaa kehittämistä hallinnoimalla suoraan kehityksen sisältöä ja osallistuu kehittämiseen osana kehitystiimiä. Tähän kuuluu esimerkiksi Scrumin tapauksessa tuotteen kehitysjonosta vastaaminen sekä osallistuminen Scrum-tiimin kokouksiin. *Epäsuorasti* osallistuva asiakas puolestaan ei suoranaisesti johda kehittämistä, ei itse osallistu kehittämiseen osana kehitystiimiä eikä myöskään toimi suoraan missään ketterän kehittämisen menetelmien mukaisissa rooleissa. Sen sijaan epäsuorasti osallistuvan asiakkaan tapauksessa ketterää kehittämistä toteutetaan kehittävän yrityksen sisäisenä menetelmänä ja yrityksellä voi olla käytössä sisäinen asiakas, jonka tehtävänä on edustaa varsinaista asiakasta. Tässä tapauksessa kehittämisen läpinäkyvyys asiakkaalle on vähäisempi. (Svenbrant, 2008.)

Resurssien tarve tuo tarkasteluun pelkän aktiviteettien etujen arvioinnin lisäksi myös näkökulman niiden kustannusten tarkastelemiseksi. Eri aktiviteettien resurssien tarve on huomattavan erilainen ja sen voidaan katsoa olevan käänteisesti riippuvainen aktiviteetin rikkaudesta. Rikkaudeltaan suuremmissa aktiviteeteissa asiakkaan osallistuminen on syvempää ja se heijastuu myös suurempana resurssien tarpeena. Osallistumisen rikkauden ja resurssien tarpeen riippuessa toisistaan voidaan myös nähdä tarve tehdä valintoja (tradeoffs) sen suhteen, kuinka paljon ja miten hyödyllistä on osallistua milläkin rikkauden tasolla. (Markus & Mao, 2004.)

5.5.2 Osallistumisen aktiviteettien arviointi

Osallistumisen aktiviteettien arviointi aloitetaan luokittelemalla aktiviteetit taulukossa 14 esitetyn mukaisesti. Luokittelun tarkoituksena on tiivistää eri lähteiden aktiviteetteja koskeva sisältö sekä integroida teoreettiset ajatukset ja empirinen tietämys toisiinsa. Taulukossa kukin asiakkaan osallistumisen aktiviteetti on luokiteltu tyyppin perusteella pohjautuen nelijakoiseen aktiviteettien kategorisointiin ja lisäksi kunkin kohdalla on esitetty teoreettisiin lähteisiin perustuvat rikkaus ja muoto. Osallistumisen aktiviteettien keskinäisen vertailun mahdollistamiseksi kunkin aktiviteetin rikkaus on määritelty Svenbrantin (2008) esittämän mukaisesti kaksiporaisella asteikolla (korkea/matala). Aktiviteettien muoto on puolestaan määritelty sen perusteella, kuuluuko siihen suora vaikutusmahdollisuus kehittämisen kohteeseen vai tapahtuuko vaikuttaminen välillisesti kehitystiimin kautta. Kaikki aktiviteetit eivät kuitenkaan suuntaudu suoraan kehittämisen kohteeseen ja koska tällaisille aktiviteeteille muodon määrittäminen on mahdotonta, on se niiden osalta jätetty määrittelemättä. Aktiviteettien vaatima resurssien tarve on määritelty nelijakoisella luokittelulla (matala / keskisuuri / suuri / erittäin suuri). Resurssien tarpeen määrittelyssä on pyritty osoittamaan eri aktiviteettien vaatimien resurssien määrä suhteellisesti verrattuna toisiin. Lisäksi taulukossa on kolmessa viimeisessä sarakkeessa esitetty tieto siitä, missä käsiteltyjen empiristen tutkimusten tapauksissa kyseinen aktiviteetti on ollut käytössä ja ovatko kokemukset aktiviteetin osalta kyseisen tut-

kimuksen mukaan olleet positiivisia (esitetty merkinnällä "+"), negatiivisia (esitetty merkinnällä "-") vai neutraaleja (esitetty merkinnällä "x").

TAULUKKO 14 Osallistumisen aktiviteettien luokittelu

Tyyppi	Aktiviteetti	Rikkaus/Muoto	Resurssien tarve	Griffin (2001)	Martin ym. (2004a)	Kautz (2009)
Projektinhallinta	Tilojen varaaminen kehitystiimille	Matala/-	Matala			
	Projektisuunnitelman katselmointi	Matala/ -	Matala			
	Työmääräarvioiden käsittely	Matala/ -	Matala	-		
	Kehitysprosessin seuraaminen	Matala/ -	Matala			
Sisällöstä päättäminen	Yhteydenpito loppukäyttäjiin	Korkea/-	Keskisuuri		+	+
	Tarinoiden tuottaminen ja priorisointi	Korkea/suora	Suuri	x	x	x
	Iteraatioiden ja julkaisujen suunnittelu	Korkea/epäsuora	Suuri	x		x
Testaus ja hyväksyntä	Testien tuottaminen	Korkea/epäsuora	Keskisuuri			
	Testaukseen osallistuminen	Korkea/suora	Keskisuuri			
	Testauksen suorittaminen	Korkea/suora	Keskisuuri		+	+
	Iteraatioiden hyväksyminen	Korkea/epäsuora	Keskisuuri			x
	Julkaisujen hyväksyminen	Korkea/epäsuora	Keskisuuri			x
	Valmiin ohjelmiston hyväksyminen	Korkea/epäsuora	Keskisuuri			
Yhteydenpito ja konsultointi	Kehitystiimiin kuuluminen	Korkea/suora	Erittäin suuri		-	-
	Kehitystiimin ohjaaminen	Korkea/epäsuora	Keskisuuri			
	Yhteydenpito johtoon	Matala/-	Keskisuuri			
	Toimiminen kontaktina sidosryhmien välillä	Matala/-	Keskisuuri	x		

Taulukossa esitetyn luokittelun perusteella voidaan tehdä johtopäätöksiä siitä, mitkä aktiviteetit vaikuttavat teoreettisten lähteiden käsittelyn perusteella asiakkaan näkökulmasta tehokkaimmilla tavoilla vaikuttaa kehittämiseen sekä

siitä, mitkä aktiviteetit ovat empiiristen tutkimusten mukaan käytännön työskentelyssä korostuneet ja minkä suorittamisesta on kokemusten mukaan ollut todellista hyötyä kehittämiselle. Asiakkaan suoritettavaksi valittavien aktiviteettien tulisi olla teoreettisen tarkastelun perusteella ominaisuuksiltaan mahdollisimman optimaalisia ja lisäksi niiden tukena tulisi olla empiiristen tutkimusten osoittamaa käytännön hyötyä.

Osallistumisen aktiviteettien analysointi edellä esitetyn luokittelun perusteella suoritetaan kolmessa osassa. Ensiksi keskitytään teoreettiseen tarkasteluun rikkauden ja muodon perusteella, toiseksi tarkastellaan osallistumisen aktiviteettien resurssien tarvetta ja kolmanneksi tarkastellaan empiirisiä havaintoja osallistumisen aktiviteeteista. Kussakin kolmesta osasta keskitytään löytämään kyseiseen osaan kuuluvien kriteereiden perusteella asiakkaan suoritettavaksi suositeltavimmat eli vaikutusmahdollisuuksiltaan suurimmat ja resurssien tarpeeltaan pienimmät sekä lisäksi empiiristen kokemusten perusteella positiivisimmin arvioidut osallistumisen aktiviteetit. Lopuksi kaikkien kolmen osan perusteella esitetään joukko asiakkaan suoritettavaksi suositeltavimpia osallistumisen aktiviteetteja.

Teoreettisessa tarkastelussa todetaan rikkauden ja muodon perusteella optimaalisimmiksi osallistumisen aktiviteeteiksi rikkaudeltaan korkeat ja muodoltaan suorat. Tällaisia ovat tarinoiden tuottaminen ja priorisointi, testaukseen osallistuminen, testauksen suorittaminen sekä kehitystiimiin kuuluminen. Nämä mahdollistavat asiakkaalle paitsi merkittävät vaikutusmahdollisuudet kehittämisen sisältöön, myös mahdollisuuden vaikuttaa suoraan kehittämisen kohteeseen, ilman tarvetta käyttää kehitystiimiä välittäjänä. Lisäksi rikkauden ja muodon perusteella optimaalisimpien aktiviteettien listaan lisätään rikkaudeltaan korkea ja muodoltaan määrittelemätön yhteydenpito loppukäyttöihin.

Rikkauden ja muodon perusteella seuraavaksi optimaalisimmat osallistumisen aktiviteetit ovat rikkaudeltaan korkeat ja muodoltaan epäsuorat. Tällaisia ovat iteraatioiden ja julkaisujen suunnittelu, iteraatioiden hyväksyminen, julkaisujen hyväksyminen, valmiin ohjelmiston hyväksyminen sekä kehitystiimin ohjaaminen. Myös nämä aktiviteetit mahdollistavat asiakkaalle merkittävät vaikutusmahdollisuudet kehittämisen sisältöön, mutta edellyttävät kehitystiimin käyttämistä välittäjänä asiakkaan ja kehittämisen kohteen välillä. Iteraatioiden, julkaisujen ja valmiin ohjelmiston hyväksymistä voitaisiin pitää muodoltaan myös suorina samoin perustein kuin testauksen suorittamistakin, koska niissä asiakas tekee hyväksyvän tai hylkäävän päätöksen suoraan ilman kehitystiimin toimimista välittäjänä. Koska ne kuitenkin ovat huomattavasti testaus- ta korkeammalla tasolla suoritettavia aktiviteetteja eikä asiakas niiden yhteydessä varsinaisesti itse työskentele kehittämisen kohteen kanssa, on ne päätetty luokitella muodoltaan epäsuoriksi. Sama perustelu koskee myös iteraatioiden ja julkaisujen suunnittelun määrittelemistä muodoltaan epäsuoraksi

Rikkaudeltaan korkeiden ja muodoltaan joko suorien, määrittelemättömiin tai epäsuorien osallistumisen aktiviteettien joukon voidaan todeta sisältävän riittävän valikoiman tyypiltään sisällöstä päättämiseen liittyviä, testaukseen ja

hyväksyntään liittyviä sekä yhteydenpitoon ja konsultointiin liittyviä aktiviteetteja. Projektinhallintaan liittyvät aktiviteetit ovat kaikki rikkaukseltaan matalia ja muodoltaan epäsuoria, joten niitä ei voida pitää teoreettisen tarkastelun perusteella kovinkaan suositeltavina.

Resurssien tarpeen osalta merkittäviä havaintoja ovat kehitystiimiin kuulumisen erittäin suuri resurssien tarve. Tämän lisäksi tarinoiden tuottaminen ja priorisointi sekä iteraatioiden ja julkaisujen suunnittelu ovat resurssien tarpeeltaan suuria. Muut aktiviteetit eivät resurssien tarpeen perusteella nouse merkittävästi esille. Kehitystiimiin kuulumisen osalta asiakkaalla on mahdollista valita resurssitarpeeltaan matalampi, mutta samalla myös muodoltaan epäsuora ja siten vaikutusmahdollisuuksiltaan rajoittuneempi kehitystiimin ohjaaminen. Tarinoiden tuottamisen ja priorisoinnin tai iteraatioiden ja julkaisujen suunnittelun kohdalla vähemmän resursseja vaativaa vaihtoehtoa sen sijaan ei ole.

Empiiristen tutkimusten perusteella käytännössä korostuneita aktiviteetteja ovat erityisesti tarinoiden tuottaminen ja priorisointi, jota käsitellään kaikissa kolmessa empiirisessä tutkimuksessa. Kahdessa tutkimuksessa kolmesta käsiteltyjä aktiviteetteja puolestaan ovat yhteydenpito loppukäyttäjiiin, iteraatioiden ja julkaisujen suunnittelu, testauksen suorittaminen sekä kehitystiimiin kuuluminen. Lopuksi yhdessä kolmesta tutkimuksesta käsiteltyjä aktiviteetteja ovat työmääräarvioiden käsittely, iteraatioiden hyväksyminen, julkaisujen hyväksyminen sekä toimiminen kontaktina sidosryhmien välillä. Positiivisia arvioita näistä aktiviteeteista ovat saaneet yhteydenpito loppukäyttäjiiin sekä testauksen suorittaminen, kun taas negatiivisia arvioita ovat saaneet työmääräarvioiden käsittely sekä kehitystiimiin kuuluminen. Kehitystiimiin kuulumisen osalta negatiivisten kokemusten syynä on ollut sen asiakkaalle aiheuttama merkittävä työkuorma. Tämä tulee ottaa huomioon tehtäessä valintaa teoreettisen tarkastelun perusteella optimaalisemman kehitystiimiin kuulumisen ja teoreettisen tarkastelun perusteella vähemmän optimaalisen, mutta myös vähemmän resursseja vaativan kehitystiimin ohjaamisen välillä. Testauksen suorittamisen kohdalla positiivisten empiiristen kokemusten voidaan katsoa suosittävän sitä testaukseen osallistumisen sijasta. Loput empiiristen tutkimusten käsittelemistä aktiviteeteista eivät ole saaneet selvästi positiivista tai negatiivista arviota.

Osallistumisen aktiviteettien analysoinnin yhteenvedona voidaan todeta asiakkaan suoritettavaksi suositeltavimpien aktiviteettien olevan yhteydenpito loppukäyttäjiiin, tarinoiden tuottaminen ja priorisointi, testauksen suorittaminen sekä asiakkaan käytössä olevien resurssien mukaan joko kehitystiimiin kuuluminen tai kehitystiimin ohjaaminen. Näiden lisäksi toissijaisesti suositeltavia aktiviteetteja voidaan katsoa olevan iteraatioiden ja julkaisujen suunnittelu, iteraatioiden hyväksyminen, julkaisujen hyväksyminen sekä valmiin ohjelmiston hyväksyminen.

5.6 Yhteenveto osallistumisesta kehittämisen aikana

Ketterässä kehittämisessä asiakkaan osallistumismahdollisuudet ovat merkittävästi perinteistä kehittämistä suuremmat ja kattavat koko projektin keston. Asiakkaalla on mahdollisuus osallistumisen välityksellä vaikuttaa kehitettävän ohjelmiston tuottamaan arvoon. Myös asiakkaan osallistumisesta tulee käsitellä arvontuottamisprosessina, jossa toimenpiteistä tulee päättää niiden tuottaman arvon perusteella. Asiakkaan keinoja osallistua kehittämiseen ovat erilaiset osallistumisen aktiviteetit, joiden ominaisuudet eroavat toisistaan.

Kirjallisuudesta löytyneet asiakkaan osallistumisen aktiviteetit jakautuvat neljään kategoriaan, joita ovat projektinhallinta, sisällöstä päättäminen, testaus ja hyväksyntä sekä yhteydenpito ja konsultointi. Projektinhallinnan kategoria käsittää projektin aloitukseen ja etenemisen seurantaan liittyvät aktiviteetit. Sisällöstä päättämisen kategoria puolestaan käsittää käyttäjien tarpeiden tuntemisen, käyttäjätarinoiden tuottamisen ja priorisoinnin sekä iteraatioiden ja julkaisujen suunnittelun. Testauksen ja hyväksymisen kategoria taas käsittää korkean tason testien kirjoittamisen ja suorituksen valvomisen sekä iteraatioiden ja valmiin ohjelmiston hyväksymisen. Lopuksi yhteydenpidon ja konsultoinnin kategoria käsittää yhteydenpidon eri sidosryhmiin ja niiden välisenä kontaktina toimimisen sekä erityisesti yhteydenpidon kehitystiimiin ja heidän kysymyksiinsä vastaamisen.

Käytännön kokemusten mukaan varsinkin yhteydenpito asiakkaisiin ja testauksen suorittaminen ovat kehittämiseen erittäin hyödyllisesti vaikuttavia osallistumisen aktiviteetteja. Merkittävästi asiakkaan resursseja vaativina aktiviteetteina puolestaan nousevat esille tarinoiden tuottaminen ja priorisointi, iteraatioiden ja julkaisujen suunnittelu sekä kehitystiimiin kuuluminen.

Osallistumisen aktiviteettien arviointi asiakkaan näkökulmasta pitää sisällään neljä kriteeriä: tyyppi, rikkaus, muoto ja resurssien tarve. Tämän lisäksi arvioinnissa tarkastellaan empiiristen tutkimusten esittämiä kokemuksia. Osallistumisen aktiviteettien arviointi edellä mainittujen kriteerien perusteella antaa lopputuloksena käsityksen asiakkaan suoritettavaksi suositeltavimmista osallistumisen aktiviteeteista, joita ovat yhteydenpito loppukäyttäjiiin, tarinoiden tuottaminen ja priorisointi, testauksen suorittaminen sekä asiakkaan käytössä olevien resurssien mukaan joko kehitystiimiin kuuluminen tai kehitystiimin ohjaaminen. Näiden lisäksi toissijaisesti suositeltavia aktiviteetteja ovat iteraatioiden ja julkaisujen suunnittelu, iteraatioiden hyväksyminen, julkaisujen hyväksyminen sekä valmiin ohjelmiston hyväksyminen.

6 KETTERÄN OHJELMISTOKEHITYKSEN KOKONAISKUVA ASIAKKAAN NÄKÖKULMASTA

Edellisissä luvuissa on asiakkaan roolia ketterässä kehittämisessä tarkasteltu kolmesta näkökulmasta, joista kukin vastaa yhtä asiakkaan kokonaisnäkökulman osaa. Ensimmäiseksi tarkastelun kohteena oli sopimuksen tekeminen, toiseksi arvon määrittäminen ja kolmanneksi osallistuminen kehittämisen aikana. Tämän luvun tarkoituksena on muodostaa kokonaiskuva asiakkaan näkökulmasta ketterään kehittämiseen kokoamalla yhteen edellisissä luvuissa käsitellyt asiat. Aluksi kerrataan kussakin kolmesta osasta saadut tulokset ja pohditaan sekä arvioidaan niitä. Toiseksi esitetään kolmen osan tulokset yhdistävä kokonaiskuva ketterästä kehittämisestä asiakkaan näkökulmasta sekä prosessimalli asiakkaan päätöksenteon tukemiseksi ketterään kehittämiseen liittyvissä päätöksentekotilanteissa.

6.1 Tulosten pohdinta

Tässä alaluvussa pohditaan tutkielman kolmessa aikaisemmassa luvussa saavutettuja tuloksia ja esitetään johtopäätöksiä tulosten laadusta ja luotettavuudesta. Aluksi käsitellään sopimuksen tekemistä käsittelevän luvun tulokset, seuraavaksi arvon määrittämistä käsittelevän luvun tulokset ja lopuksi kehittämisen aikaista osallistumista koskevan luvun tulokset.

6.1.1 Sopimuksen tekeminen

Sopimusmallien arvioinnin tuloksissa ilmeni odotetusti kiinteähintaisen ja sen kanssa oleellisesti samanlaisen monivaiheisen sopimusmallin sopimattomuus ketterään kehittämiseen asiakkaan näkökulmasta. Kirjallisuudesta löytyneiden arvioiden ja kokemusten perusteella on vaikea kuvitella muunlaista tulosta. Kysymyksiä herättää sen sijaan Franklinin (2008) empiirinen tutkimus, jonka

mukaan ketterää kehittämistä voitaisiin onnistuneesti suorittaa kiinteähintaisen sopimusmallin alaisuudessa.

Samoin aika-ja-materiaalit -sopimusmallin kannattamattomuus asiakkaan kannalta ei ole yllättävä tulos. Siihen perustuvien aika-ja-materiaalit kiinteällä sisällöllä ja kustannuskatolla sekä aika-ja-materiaalit joustavalla sisällöllä ja kustannuskatolla -sopimusmallien vastaava tulos kertoo myös siitä, ettei kyseisen sopimusmallin perusrakennetta ole mahdollista vähällä muutoksella tehdä asiakkaalle edulliseksi.

Yllättävää arvioinnin tuloksissa sen sijaan oli vaiheistetun kehityksen ja sen kanssa samoja ominaisuuksia jakavan adVantage-sopimusmallin jääminen suositeltujen sopimusmallien ulkopuolelle. Syynä tähän on kyseisten sopimusmallien riippuvainen yhteistyösuhde, jonka taustalla on usein uusiutuva riski toisen osapuolen irtaantumiseen. Asiakkaan kannalta tämä riski on erityisen merkittävä, koska toimittajan irtaantuminen ennen aikaisesti voi aiheuttaa asiakkaalle kalliin ohjelmiston jäämisen puolivalmiiksi ja käyttökelvottomaksi. Lyhyissä sopimusjaksoissa on hyötyjä esimerkiksi muutostenhallinnan suhteen, mutta ne tuovat myös turvattomuutta sopimussuhteeseen. Lyhyisiin jaksoihin perustuvissa sopimusmalleissa tulisi tämän ongelman välttämiseksi olla korkeamman tasoinen yleissopimus, jossa sopimusvelvoitteen tai kannustinten keinoin ehkäistäisiin toisen osapuolen mahdollisuus irtaantua toisen vahingoksi. Esimerkiksi yhteistyösopimuksen tapauksessa tällainen kannustin on saatu sisällytettyä sopimusmalliin onnistuneesti.

Arvioinnin perusteella asiakkaan kannalta ketterään kehittämiseen suositeltaviksi luokiteltavissa sopimusmalleissa sen sijaan ei ollut varsinaisesti yllätyksiä. Varsinkin yhteistyösopimuksen selvä ylivoimaisuus muihin sopimusmalleihin nähden oli kirjallisuuden perusteella odotettavissa (Thorup & Jensen, 2009). Myös tavoitehintasopimuksen ja tavoiteaikasopimuksen osalta löytyi kirjallisuudesta hyviä kokemuksia, mutta epäselvyyttä niiden kohdalla aiheuttaa Poppendieckin ja Poppendieckin (2003, s.163) teoreettisen arvion ja Eckfeldtin ym. (2005) empiirisen kokemuksen välinen eroavaisuus muutostenhallinnan suhteen. Muutostenhallinnan helppous ja vaikeus on tietysti subjektiivinen kysymys ja yhteenvetona eri tuloksista voitaneen sanoa muutostenhallinnan olevan näiden sopimusmallien kohdalla mahdollista, mutta sen toteuttaminen ei tapahdu automaattisesti vaan vaatii neuvottelua ja kustannusvaikutusten laskemista. "Rahaa tyhjästä ja muutokset ilmaiseksi" -sopimusmallin kohdalla mielenkiintoinen havainto on kiinteähintaisen sopimusmallin perusrakenteen muokkaantuminen suhteellisen vähällä muutoksilla ketterään kehittämiseen soveltuvaksi. Kyseessä on erittäin innovatiivinen ajatus ja ehdottomasti kokeilemisen arvoinen kiinteähintaista sopimusmallia vaativien asiakkaiden tapauksessa. Suositeltavien sopimusmallien kohdalla on myös merkillepantavaa se, että ne perustuvat kolmelle toisistaan poikkeavalle rakenteelle ja soveltuvat siten hyvin erilaisiin tilanteisiin. Tämän tulisi mahdollistaa oikean sopimusmallin löytyminen jokaisessa tilanteessa.

Sopimuksen tekemisen alueelta suositeltava jatkotutkimisaihe empiiriselle tutkimukselle olisi sen selvittäminen, miten hyvin "rahaa tyhjästä ja muutokset

ilmaiseksi” -sopimusmalli käytännössä soveltuu kiinteähintaisen sopimusmallin vaihtoehdoksi tai tavoitehinta- ja tavoiteaikasopimus aika-ja-materiaalit -sopimusmallin vaihtoehdoksi. Lisäksi jatkotutkimuksessa olisi mahdollista keskittyä riskin jakautumiseen osapuolten välillä ja tutkia sitä, missä määrin riskin tasainen jakaminen todellisuudessa hyödyttää molempia osapuolia verrattuna oman edun tavoitteluun perustuviin pyrkimyksiin sopimusneuvotteluissa. Sopimuksen tekemisen kokonaisuudessaan voidaan siihen liittyvien rahallisten seurauksien johdosta todeta olevan hyvin tärkeä tutkimusalue ohjelmistokehityksessä yleensä sekä varsinkin ketterässä kehittämisessä. Optimaalisen sopimuksen löytämiseen kohdistuvan tutkimuksen tärkeyttä korostaa Thorupin ja Jensenin (2009) loppupäätelmä, jonka mukaan suurissa kehittämisprojekteissa voidaan nähdä tuhlatavan paljon rahaa huonojen sopimusten vuoksi.

6.1.2 Arvon määrittäminen

Arvon määrittämistä koskevien tulosten voidaan kokonaisuudessaan katsoa jääneen valitettavan vajavaisiksi. Kirjallisuudesta löydettyjen menetelmien määrä ja ennen kaikkea laatu sekä koko arvon määrittämisen tutkimuksen ja käytännön nykytila vaikuttavat olevan merkittävästi odotettua vajavaisempia.

Tulosten vajavaisuuden voidaan katsoa juontavan juurensa arvon käsitteen abstraktiuteen sekä sen määritelmän puutteellisuuteen ketterän kehittämisen yhteydessä. Racheva ym. (2009) toteavat kysymyksen arvon määritelmästä ketterän kehittämisen yhteydessä jäävän heidän suorittamansa systemaattisen tutkimuksen jälkeenkin avoimeksi. Arvon määritelmä ketterän kehittämisen yhteydessä vaatisikin kirjallisuudessa täysin uuden ja tarkemman käsittelyn. Lisäksi tulosten vajavaisuuteen osasyynä lienee yhteenvetona esitetystä arvon määritelmästä (Racheva ym., 2009) tärkeällä sijalla olevat kvalitatiivisuus ja subjektiivisuus, koska on vaikeata määrittää tai edes tarkoin käsittää sellaista mitä ei voida mitata.

Tulosten vajavaisuudesta huolimatta tutkimuksen voidaan kuitenkin katsoa tarjoavan käsityksen aiheen tutkimuksen ja käytännön nykytilasta sekä käsityksen arvon määrittämisen alueesta mahdollista käytäntöön soveltamista tai jatkotutkimusta varten. Rawsthornen (2004, 2006) liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittämisen menetelmä näyttäisi tällä hetkellä lupavimmalta käytännössä sovellettavaksi, joskin siitäkin puuttuu negatiivisen arvon huomioon ottaminen ja lisäksi sen soveltaminen ketterään kehittämiseen ei ole sellaisenaan suositeltavaa. Rawsthornen (2004, 2006) menetelmän käyttö ketterässä kehittämisessä edellyttää sen soveltamista esimerkiksi samoin kuin Alleman ym. (2003), Cabri ja Griffiths (2006) sekä Sulaiman ym. (2006) soveltavat tutkimuksissaan perinteistä ansaitun arvon menetelmää ketterään kehittämiseen.

Arvon määrittämisen käsittelyn yhteydessä on esitetty alustava hahmotelma käsiteltyjen lähteiden perusteella muodostuvasta arvon määrittämisen kokonaiskuvasta sekä kaikki sen ulottuvuudet huomioon ottavasta menetelmästä, joka voisi toimia mahdollisena aiheena jatkotutkimukselle. Aiheesta voidaan

todeta muutenkin olevan huomattavan paljon jatkotutkittavaa, lähtien aiemmin mainitusta arvon määritelmästä ketterän kehittämisen yhteydessä ja edeten siitä aina käytännössä sovellettaviin arvon määrittämisen menetelmiin. Aiheen tärkeydestä huolimatta se näyttää olevan niin tutkimuksessa kuin käytännössäkin pahasti laiminlyöty. Arvon määrittämisen käsittelyn yhteenvedoksi sopii hyvin Allemanin ym. (2003) esittämä lainaus:

Kaikki mitattava ei ole merkittävää, ja kaikki merkittävä ei ole mitattavaa. – Albert Einstein

Arvon määrittämisen voidaan – ainakin tällä hetkellä – selvästi huomata kuuluvan näistä kahdesta ryhmästä jälkimmäiseen.

6.1.3 Osallistuminen kehittämisen aikana

Asiakkaan kehittämisen aikaista osallistumista koskevien tuloksien voidaan kokonaisuudessaan katsoa olevan odotetun kaltaisia. Ketterässä kehittämisessä nimenomaan tuotteen arvoon suoranaisesti vaikuttavat toimenpiteet on priorisoitu kaiken muun edelle (Agile Manifesto, 2001), mikä heijastuu teoreettisessa tarkastelussa varsinkin tyypiltään projektinhallintaan kuuluvien aktiviteettien jäämisessä asiakkaan suoritettavaksi suositeltavimpien aktiviteettien ulkopuolelle. Toisaalta taas suositeltavimpien aktiviteettien joukossa korostuvat nimenomaan asiakkaan vahvaa osallistumista vaativat aktiviteetit kuten asiakastiiimiin kuuluminen ja testauksen suorittaminen.

Empiirisiä tutkimuksia aiheesta on tehty yllättävän vähän. Tässä tutkielmassa käsitellyissä tutkimuksissa korostuivat erityisesti sisällöstä päättäminen (Griffin, 2001; Martin ym., 2004; Kautz, 2009) sekä jossain määrin myös testaus ja hyväksyntä (Martin ym., 2004; Kautz, 2009), projektinhallinnan sekä yhteydenpidon ja konsultoinnin jäädessä vähemmälle huomiolle. On täysin ymmärrettävää ja odotettua priorisoida rajallisten resurssien tapauksissa nimenomaan sisällöstä päättäminen sekä testaus ja hyväksyntä vähemmän arvoon vaikuttavien aktiviteettien edelle.

Mielenkiintoista kirjallisuudesta löydettyjen aktiviteettien suhteen oli saman asian toistuminen eri rikkauden tasoilla ja muodoissa kuten esimerkiksi testien tuottaminen (Beck, 1999a; Van Deursen, 2001), testaukseen osallistuminen (Beck, 1999a; Wang ym., 2008) ja testauksen suorittaminen (Martin ym., 2004a; Kautz, 2009) tai kehitystiimin ohjaaminen (Beck, 1999a; Van Deursen, 2001; Wang ym., 2008) ja kehitystiimiin kuuluminen (Beck, 1999a; Schwaber & Sutherland, 2011; Martin ym., 2004a; Kautz, 2009). Tällaisessa tapauksessa aktiviteettien valinnan tulee tapahtua käytössä olevien resurssien perusteella ja valintaperusteena voidaan tällöin katsoa myös olevan kehittämisessä tavoiteltava arvo sekä se paljonko tuon arvon saavuttamiseksi ollaan valmiita käyttämään resursseja.

Yllättävä huomio empiiristen tutkimusten käsittelyssä oli kehitystiimiin kuulumisen saamat merkittävän negatiiviset huomiot resurssien vaativuudes-

taan (Martin ym., 2004; Kautz, 2009). Vaikka kyseisen aktiviteetin työläys oli odotettavissa, ylittivät käytännön tapauksissa esille tulleet havainnot kuitenkin odotukset tässä suhteessa. Tämän perusteella tuli myös ilmi tarve suositella muodoltaan epäsuoraa ja siksi teoreettisen tarkastelun perusteella epäoptimaalisempaa asiakastiimin ohjaamista vähemmän resursseja vaativana vaihtoehtona asiakastiimiin kuulumiselle.

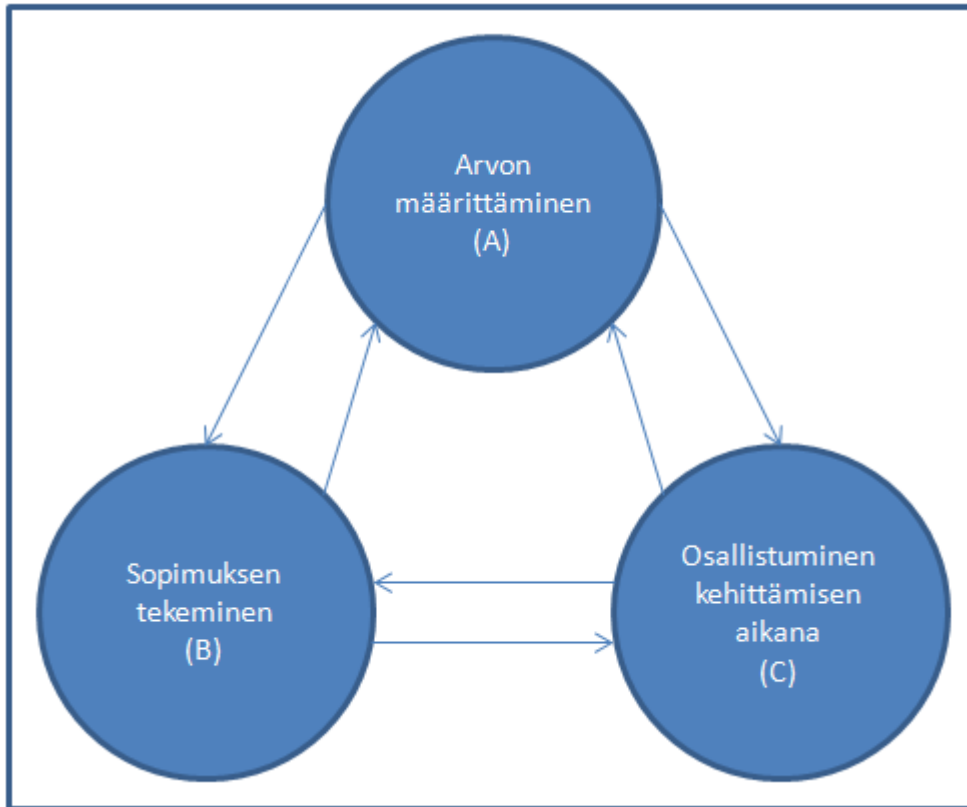
Teoreettisen tarkastelun perusteella tehdyissä johtopäätöksissä heijastuu jossain määrin arvioinnissa käytettyjen kriteereiden kvalitatiivisuus ja suhteellisuus, jonka vuoksi vasta empiiristen tutkimusten vahvistamat johtopäätökset voidaan katsoa perustelluiksi. Osasta teoreettisen tarkastelun perusteella varsin merkityksettömän roolin saaneita, pääosin projektinhallintaan sekä yhteydenpitoon ja konsultointiin liittyviä aktiviteetteja, tällaisia empiirisiä tutkimuksia ei ollut saatavilla ja tämän voidaan tulkita merkitsevän joko niiden käsittelyn jäämistä vajaaksi tai niiden merkityksettömyyttä käytännössä ja siten teoreettisen tarkastelun tulosten vahvistamista. Tuloksien tarkastelussa tulee myös huomioida kirjallisuudesta löytyneiden aktiviteettien osittainen päällekkäisyys ja mahdolliset aktiviteettien yhdistelyssä tehdyt luokitteluvirheet, jonka perusteella osa lähteistä on mahdollisesti kohdennettu väärään yhdistettyyn aktiviteettiin ja mikä olisi omiaan vääristämään lopullisia tuloksia.

Kehittämisen aikaisen osallistumisen alueelta jatkotutkimuksen olisi suositeltavaa painottua empiirisen tutkimuksen suuntaan. Suositeltava aihe tällaiselle empiiriselle tutkimukselle olisi - tällä hetkellä valitettavan suppean - asiakkaan osallistumisen aktiviteetteja koskevan tapaustutkimusten joukon jatkaminen.

6.2 Tuloksista muodostuva kokonaiskuva

Asiakkaan näkökulmaan kuuluvat kolme osaa eli sopimuksen tekeminen, arvon määrittäminen ja osallistuminen kehittämisen aikana, on tähän asti tässä tutkielmassa käsitelty toisistaan erillisinä osina. Lopuksi tarkastellaan kuitenkin vielä näiden osien suhteita toisiinsa ja niistä muodostuvaa kokonaiskuvaa ketterästä kehittämisestä asiakkaan näkökulmasta sekä esitetään tuloksien soveltamista havainnollistava prosessimalli asiakkaan toiminnalle ketterässä kehittämisessä

Asiakkaan näkökulman kokonaiskuvaa voidaan hahmottaa verkkoteorian mukaisesti kolmen solmun (näkökulman osat) ja niiden välisten polkujen (näkökulman osien väliset vuorovaikutussuhteet) muodostamana suunnattuna verkkona. Kyseessä on nimenomaan suunnattu verkko, koska vuorovaikutussuhteilla on suunta eli toinen osa vaikuttaa toiseen. Tällainen näkökulman osien ja niiden välisten vuorovaikutussuhteiden muodostama suunnattu verkko on esitetty kuviossa 21.



KUVIO 21 Asiakkaan näkökulman osien vuorovaikutussuhteet

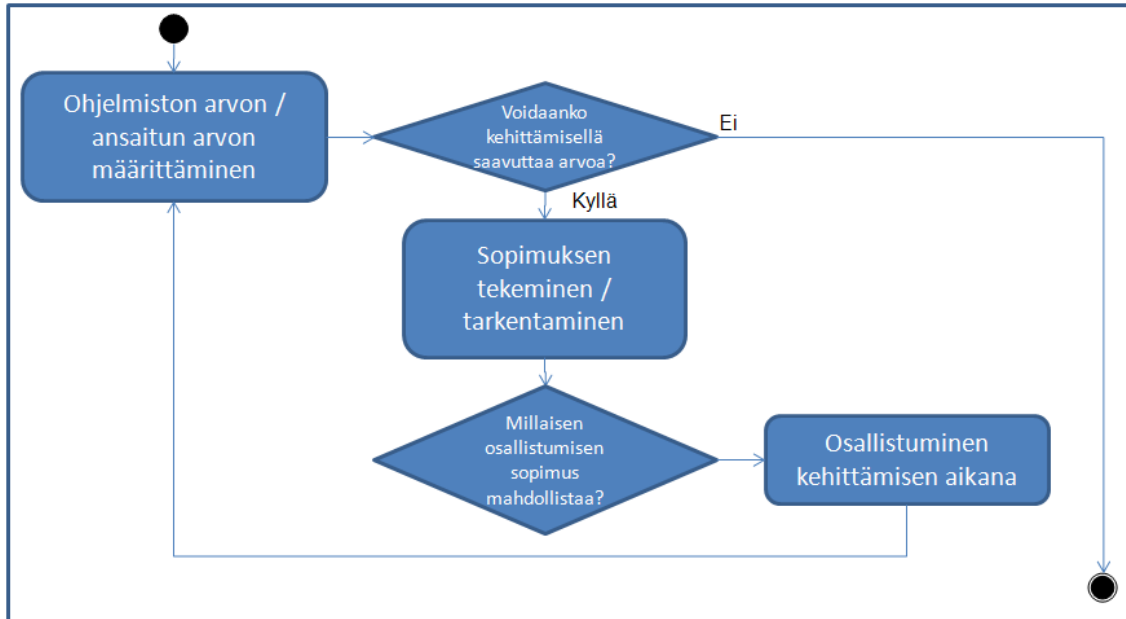
Verkon solmuilla ja niitä vastaavilla asiakkaan näkökulman osilla A (arvon määrittäminen), B (sopimuksen tekeminen) ja C (osallistuminen kehittämisen aikana) on keskenään yhteensä kuusi vuorovaikutussuhdetta eli verkon solmujen väliset polut A->B, B->A, B->C, C->B, A->C ja C->A. Nämä vuorovaikutussuhteet on esitetty taulukossa 15. Taulukossa on esitetty suunnatun verkon polku ja kyseistä polkua vastaavan vuorovaikutussuhteen lähteenä ja kohteena olevat näkökulman osat sekä näiden osien välisessä vuorovaikutussuhteessa tapahtuva vaikutus.

Vuorovaikutussuhteiden analysoinnin perusteella voidaan huomata kaikkien asiakkaan näkökulman kolmen osan vaikuttavan toisiinsa ja riippuvan toisistaan. Näin ollen kokonaisuudesta voi helposti muodostua asiakkaalle monimutkaisten riippuvaisuuksien verkosto, jossa on vaikea hahmottaa eri toimenpiteiden vaikutuksia toisiinsa. Asiakkaan toiminnan selkeyttämiseksi onkin tarpeen esittää yksiselitteisesti seurattavissa oleva prosessi, joka liittää yhteen asiakkaan näkökulman eri osat ja ohjaa asiakkaan toimintaa. Tällainen prosessimalli ketterästä kehittämisestä asiakkaan näkökulmasta katsottuna ja perustuen tämän tutkielman aikana tehtyihin havaintoihin ja johtopäätöksiin on kuvattu aktiviteettikaavion muodossa kuviossa 22. Ketterä kehittäminen on määritelmänsä mukaisesti iteratiivinen prosessi ja myös asiakkaan toiminta ketterässä kehittämisessä tapahtuu iteratiivisesti. Näin ollen myös kuvatussa prosessimallissa toistuvat iteratiivisesti läpi koko kehittämisen tässä tutkielmassa käsi-

tellyt asiakkaan näkökulman kolme osaa eli arvon määrittäminen, sopimuksen tekeminen ja osallistuminen kehittämisen aikana.

TAULUKKO 15 Asiakkaan näkökulman osien väliset vuorovaikutussuhteet

Polku	Lähde	Kohde	Vaikutus
A->B	Arvon määrittäminen	Sopimuksen tekeminen	Sopimusmalli tulee valita tavoiteltavan arvon mukaisesti. Optimaalinen sopimusmalli on erilainen riippuen siitä onko tavoitteena esimerkiksi mahdollisimman laadukas ohjelmisto kustannuksista välittämättä, mahdollisimman halpa tarkoitukseen riittävä ohjelmisto vai tarkoitukseen riittävä ohjelmisto tietyllä aikataululla.
B->A	Sopimuksen tekeminen	Arvon määrittäminen	Valittu sopimusmalli vaikuttaa saavutettavaan arvoon. Epäoptimaalinen sopimusmalli vaikeuttaa kehittämistä ja voi esimerkiksi alentaa kehitettävän ohjelmiston tuottamaa arvoa sen alentuneena käyttökelpoisuutena tai kehitettävästä ohjelmistosta saatavaa nettoarvoa kohonneiden kehityskustannusten kautta.
B->C	Sopimuksen tekeminen	Osallistuminen kehittämisen aikana	Sopimusmalli vaikuttaa asiakkaan mahdollisuuksiin osallistua kehittämisen aikana. Esimerkiksi puhtaasti kiinteähintainen sopimusmalli rajoittaa ketterän kehittämisen mukaisia mahdollisuuksia asiakkaan osallistumiselle kehittämisen aikana, kun taas joustavammat sopimusmallit mahdollistavat asiakkaan tehokkaan osallistumisen.
C->B	Osallistuminen kehittämisen aikana	Sopimuksen tekeminen	Sopimusmallin valinnassa tulee ottaa huomioon toivottu osallistumisen määrä. Koska eri sopimusmallit vaikuttavat asiakkaan osallistumismahdollisuuksiin rajoittavasti tai edesauttavasti, tulee nämä vaikutukset ennakoita ja ottaa huomioon sopimusmallin valinnassa.
A->C	Arvon määrittäminen	Osallistuminen kehittämisen aikana	Osallistuminen kehittämisen aikana perustuu kehittämisellä tavoitellulle arvolle. Osallistuminen kehittämisen aikana on arvontuottamisprosessi ja osallistumisen toimenpiteet tulee valita niiden tuottaman arvon perusteella.
C->A	Osallistuminen kehittämisen aikana	Arvon määrittäminen	Osallistuminen kehittämisen aikana mahdollistaa saavutettavaan arvoon vaikuttamisen. Ilman osallistumista kehitettävän ohjelmiston tuottama arvo ei tule olemaan optimaalinen kun taas tehokkaasti osallistumalla asiakkaalla on mahdollista nostaa ohjelmiston tuottama arvoa.



KUVIO 22 Asiakkaan toiminta ketterässä kehittämisessä

Asiakkaan näkökulman osien keskinäisistä vuorovaikutussuhteista huolimatta voidaan yksi näkökulman osa nostaa selvästi keskeiseen asemaan päätöksentekoa ohjaavana kriteerinä. Arvon määrittämistä koskevassa luvussa tuodaan esille arvon tuottamisen keskeisyys ketterässä kehittämisessä ja todetaan koko ketterän kehittämisen päätavoitteena olevan arvon tuottaminen (Agile Manifesto, 2001; Abrahamsson ym., 2002; Boehm, 2002; Highsmith, 2002). Näin ollen voidaan havaita arvon ja sen määrittämisen olevan perusteltu lähtökohta asiakkaan päätöksentekoprosessille. Kehittämisprojektin tulee arvon määrittämisen perusteella tuottaa valmistuessaan toteutuneita kustannuksiaan enemmän arvoa ja vain tällaisten nettoarvoa tuottavien kehittämisprojektien käynnistäminen on oikeutettua (Mutschler ym., 2007). Samoin tulee myös läpi koko kehittämisen tarkastella sen jatkamisesta saatavaa arvoa suhteessa kehittämisen kustannuksiin, ja kehittämistä tulee jatkaa vain siihen saakka kuin se on tämän tarkastelun mukaan perusteltua, kuten esimerkiksi "rahaa tyhjästä ja muutokset ilmaiseksi" -sopimusmallin yhteydessä mainitaan (Sutherland, 2008).

Arvon määrittämisen ja mahdollisesti positiivisen aloitus- tai jatkamispäätöksen jälkeen jatketaan sopimuksen tekemiseen tai tarkentamiseen. Sopimuksen tekemiseen vaikuttavat tavoiteltava arvo sekä asiakkaan toivoma osallistumisen määrä. Sopimuksesta on mahdollista neuvotella kiinteäsisältöinen, jolloin asiakkaan kehittämisen aikainen osallistuminen rajautuu lähes täysin pois (Van Cauwanberghe, 2003a; Stevens, 2009b). Sopimuksesta voidaan myös neuvotella esimerkiksi aika- ja materiaalit - muotoinen tai yhteistyösopimuksen mukainen, jolloin asiakas on oikeutettu ja velvollinen osallistumaan voimakkaasti koko kehittämisen ajan (Stevens, 2009b; Thorup & Jensen, 2009). Myös sopimuksen tekeminen ja tarkentaminen tapahtuu iteratiivisesti koko kehittämisen ajan, sillä vaikka perinteisesti sopimuksen tekemisen käsitetään liittyvän vain kehittämisprojektin alkuun, liittyy lähes kaikkiin käsiteltyihin sopimusmalleihin ehtoja

sopimuksen tarkentamisesta tai uudelleen neuvottelemisesta kehittämisen kuluessa (Book ym., 2012; Poppendieck & Poppendieck, 2003; Stevens, 2009b; Sutherland, 2008; Thorup & Jensen, 2009).

Sopimuksen tekemisen tai tarkentamisen jälkeen asiakkaan tulee toimia kehittämisprojektin hyväksi sopimuksen mahdollistamissa rajoissa edistämällä kehittämistä ja sen tuottamaa arvoa. Aiemmin mainitun mukaisesti osa sopimusmalleista rajoittaa asiakkaan osallistumismahdollisuuksia, kun taas toiset sopimusmallit kannustavat ja suorastaan velvoittavat asiakkaan osallistumaan. Asiakkaan mahdollisuutena toimia tässä ovat kehittämisen aikaisen osallistumisen aktiviteetit, joilla hän voi ohjata kehittämistä haluamaansa suuntaan (Markus & Mao, 2004). Osallistumisen aktiviteettien valinnassa asiakkaan tulee ottaa huomioon kunkin aktiviteetin ominaisuudet tyyppin, rikkauden ja muodon osalta sekä eri aktiviteettien resurssien tarve samoin kuin kokemukset niiden soveltamisesta käytäntöön. Osallistumisen yhteydessä asiakkaan tulee edelleen suorittaa arvon määrittämistä tarkkailemalla kehittämisen edetessä saavutetun arvon muuttumista (Rawsthorne 2004, 2006; Alleman 2003; Cabri & Griffiths, 2006; Sulaiman ym., 2006), jolloin kuviossa 22 esitetty prosessi alkaa iteratiivisesti uudelleen alusta.

Kuvattua prosessimallia seuraamalla asiakkaalla on mahdollisuus tehdä perusteltuja valintoja kehittämisprojektiin liittyvissä päätöksentekotilanteissa ja siten edesauttaa kehittämisen onnistumista ja sen välityksellä saavutettavaa arvoa.

7 YHTEENVETO

Tämän tutkielman tavoitteena oli selvittää, miten asiakas voi toimia menestyksekkäästi ketterässä ohjelmistokehityksessä. Tässä tarkoituksessa tutkielmassa rakennettiin laajaan kirjallisuuteen perustuen jäsennyksiä, joilla voidaan valaista asiakkaan näkökulmaa sopimuksen tekemiseen, arvon määrittämiseen ja osallistumiseen kehittämisen aikana sekä arvioida ja vertailla kirjallisuudessa esitettyjä sopimusmalleja, arvon määrittämisen menetelmiä ja osallistumisen aktiviteetteja asiakkaan näkökulmasta.

Tutkielmassa käsiteltiin aluksi ketterää ohjelmistokehitystä yleisesti. Esitetyn määritelmän mukaan ketterä kehittäminen merkitsee keveiden, mukautuvien ja yksinkertaisten kehitysmenetelmien käyttöä iteratiivisesti ja inkrementaalisesti suoritettavassa ohjelmistokehityksessä, jossa korostetaan ihmiskeskeisyyttä ja yhteistyötä, tavoitteena tuottaa asiakkaalle arvoa nopeasti ja jatkuvasti sekä hyväksyen muutokset. Ketterän kehittämisen menetelmistä esiteltiin kaksi eniten käytettyä eli Scrum ja Extreme Programming (XP) sekä lisäksi Lean-ajatteluun perustuva lähestymistapa ohjelmistokehitykseen. Ketterän kehittämisen käsittelyn yhteydessä käsiteltiin myös asiakkaan merkittävää ja vaativaa roolia ketterässä kehittämisessä ja määriteltiin asiakas tahoksi, jolla on hallussaan kehittämiseen käytettävät varat ja valtuudet päättää hankinnoista. Luvun merkittävin tarkoitus koko tutkielman kannalta oli johtaa asiakkaan näkökulma ketterään kehittämiseen. Ketterää lähestymistapaa sekä asiakkaan roolia koskevaa teoreettista ja empiiristä tutkimusta koskevaa kirjallisuutta käyttäen määriteltiin asiakkaan näkökulma koostuvaksi kolmesta osasta. Osat ovat sopimuksen tekeminen, kehitettävän ohjelmiston arvon määrittäminen ja osallistuminen kehittämisen aikana. Tämän jälkeen tutkielmassa keskityttiin yksitellen kuhunkin näistä kolmesta piirteestä.

Ensimmäisenä asiakkaan näkökulman osana tutkielmassa käsiteltiin sopimuksen tekemistä. Tämän luvun tarkoituksena oli kartoittaa ketterässä kehittämisessä yleisesti käytettyjä sopimusmalleja ja löytää niistä asiakkaan näkökulmasta suositeltavimmat. Sopimusmallien käsittelyssä sopimusmallit jaoteltiin kiinteähintaisiin sopimusmalleihin, aika-ja-materiaalit -sopimusmalleihin sekä hybridisopimusmalleihin. Kirjallisuudesta johdetut sopimusmallien arvi-

oinnin kriteerit pitivät sisällään sopimuksen rakenteen eli tuotosten ja laskutuksen hallinnoinnin, riskin jakautumisen osapuolten kesken, muutostenhallinnan sekä yhteistyömallin. Sopimusmallien arviointi edellä mainittujen kriteereiden perusteella antoi lopputuloksena käsityksen asiakkaan näkökulmasta parhaiten ketterään kehittämiseen sopivista sopimusmalleista.

Toisena asiakkaan näkökulman osana tutkielmassa käsiteltiin arvon määrittämistä. Tämän luvun tarkoituksena oli selvittää, millaisia keinoja on olemassa kehitettävän ohjelmiston arvon määrittämiseen ja mitkä niistä ovat tähän tehtävään asiakkaan kannalta parhaiten soveltuvia. Arvon määrittämisen menetelmien todettiin jakaantuvan niiden yhteydessä käytetyn arvoon viittaavan termin mukaan kolmeen ryhmään eli ansaitun arvon määrittämiseen, taloudellisen arvon määrittämiseen sekä liiketoiminta-arvon ja ansaitun liiketoiminta-arvon määrittämiseen. Näiden lisäksi käsiteltiin kirjallisuudesta löytynyttä negatiivisen arvon käsitettä. Arvon määrittämisen menetelmien arvioinnin kriteerit pitivät sisällään kaikki löydettyjen menetelmien käsittelemät ulottuvuudet eli ohjelmiston arvon määrittäminen, arvon siirto nykyiseen nettoarvoonsa, arvon jakaminen osiin ja ansaitun arvon määrittäminen sekä negatiivisen arvon huomioonottaminen ja lisäksi yhteensopivuuden ketterän kehittämisen kanssa. Arvon määrittämisen menetelmien arviointi asiakkaan näkökulmasta toi ilmi kaikkien käsiteltyjen menetelmien puutteellisuuden sekä käsityksen puutteellisuudestaan huolimatta käyttökelpoisimmista menetelmistä. Kirjallisuudesta löydettyjen menetelmien arvioinnin lisäksi esiteltiin käsiteltyjen lähteiden perusteella muodostettua arvon määrittämisen kokonaiskuvaa ja kaikki sen ulottuvuudet huomioon ottavan menetelmän hahmotelma.

Kolmantena asiakkaan näkökulman osana tutkielmassa käsiteltiin osallistumista kehittämisen aikana. Tämän luvun tarkoituksena oli selvittää, millaisia osallistumismuotoja asiakkaalla on valittavanaan ketterässä kehittämisessä ja mitkä niistä ovat asiakkaan kannalta suositeltavimpia. Kirjallisuudesta löytyneet asiakkaan osallistumisen aktiviteetit jakautuvat neljään kategoriaan, joita ovat projektinhallinta, sisällöstä päättäminen, testaus ja hyväksyntä sekä yhteydenpito ja konsultointi. Osallistumisen aktiviteettien arvioinnin kriteerit pitävät sisällään tyyppin, rikkauden, muodon sekä resurssien tarpeen ja näiden lisäksi arvioinnissa huomioitiin empiiristen tutkimusten esittämät käytännön kokemukset. Osallistumisen aktiviteettien arviointi edellä mainittujen kriteerien perusteella antoi lopputuloksena käsityksen asiakkaan suoritettavaksi suositeltavimmista osallistumisen aktiviteeteista.

Lopuksi suoritettiin tulosten pohdinta kertaamalla kolmea asiakkaan näkökulman osaa koskevien lukujen tulokset ja tekemällä johtopäätöksiä niiden laadusta ja luotettavuudesta. Tässä yhteydessä kerrottiin myös jatkotutkimustarpeista. Tämän jälkeen käsiteltiin näkökulman osien välisiä suhteita ja esitettiin niiden muodostama kokonaiskuva. Lisäksi muodostettiin kokonaiskuvan perusteella prosessimalli asiakkaan toiminnasta ketterässä kehittämisessä helpottamaan tutkielman tulosten hyödyntämistä käytännössä.

Tutkielma sisältää monia tutkimuksen kannalta merkittäviä kontribuutioita. Muodostettu asiakkaan näkökulma on kattavampi kuin mitä kirjallisuudessa

on aiemmin esitetty (esim. Lohan ym., 2010). Tutkielman ansioksi voidaan lukea myös selkeän kokonaisjäsennyksen lisäksi kunkin asiakkaan näkökulman osan jäsentämisen tarkemmalla tasolla, kattavuudeltaan ja rakenteeltaan aiemmista esityksistä poikkeavasti. Näkökulman osien mukainen jäsenitys osoittautui toimivaksi kirjallisuudessa esitettyjen sopimusmallien, arvonmäärittämisen menetelmien sekä osallistumisen muotojen arviointikriteereiden määrittämisessä ja arvioinnin suorittamisessa. Kolmanneksi työssä hahmoteltiin uudenlaisen arvon määrittämisen menetelmän hahmotelma poimimalla ja yhdistämällä ketterän kehittämisen yhteyteen parhaiten sopivia osia aiemmin esitetyistä arvon määrittämisen menetelmistä. Neljäntenä kontribuutiona on asiakkaan näkökulmien osien yhdistäminen ja prosessimallin esittäminen asiakkaan päätöksenteon tueksi.

Tutkielman tuloksia voidaan hyödyntää käytännön työssä monella tavalla. Muodostettu asiakkaan näkökulma määrittelee sen mitä ketterä kehittäminen käytännössä merkitsee asiakkaan kannalta ja mitä asiakkaan rooliin ketterässä kehittämisessä kuuluu. Tämä selventää asiakkaan roolia ja tähän rooliin kuuluvia tehtäviä sekä siten myös selkeyttää asiakkaan ja kehittäjien välistä työnjakoa. Kunkin asiakkaan näkökulman osan käsittelyssä saavutetut tulokset puolestaan antavat asiakkaalle selkeän näkemyksen käytettävistä olevista keinoista rooliinsa kuuluvissa tehtävissä menestymiseksi. Käytännön tasolla tämä merkitsee asiakkaalle mahdollisuutta suorittaa tietoinen ja perusteltu valinta eri sopimusmallien, arvon määrittämisen menetelmien ja kehittämisen aikaisen osallistumisen aktiviteettien välillä. Lisäksi esitetty prosessimalli asiakkaan toiminnasta ketterässä kehittämisessä ohjaa asiakasta käytännön tasolla tunnistamaan rooliinsa kuuluvat tehtävät ja tarvittavat päätökset sekä tekemään tietoisia ja perusteltuja valintoja. Lopuksi tutkielman tuloksia voidaan käyttää asiakkaiden auttamisen lisäksi myös kehittäjien tietoisuuden lisäämiseksi asiakkaan roolista, tuon roolin haasteista sekä käytettävissä olevista keinoista siinä menestymiseksi ja näin auttaa kehittäjiä tukemaan asiakasta roolissaan molempien osapuolten sekä koko kehittämisprojektin hyötyä edistävällä tavalla.

LÄHTEET

- Abbas, N., Gravell, A. & Wills, G. (2008). Historical Roots of Agile Methods : Where did "Agile Thinking" Come from? Teoksessa P. Abrahamsson, R.Baskerville, K.Conboy, B.Fitzgerald & X.Wang (toim.), *Proceedings of the 9th International Conference on Agile Processes in Software Engineering and Extreme Programming, XP 2008, Limerick, Ireland, June 10-14* (s.94-103). Berlin: Springer.
- Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). *Agile software development methods, Review and Analysis*. VTT Publications 478.
- Abrahamsson, P., Warsta, J., Siponen, M. & Ronkainen, J. (2003). New Directions on Agile Methods: A Comparative Analysis. *Teoksessa Proceedings of the 25th International Conference on Software Engineering, ICSE'03, Portland, Oregon, Usa, May 3-10* (s. 244-254). Washington, DC, USA: IEEE Computer Society.
- Agile Manifesto (2001). *Agile Manifesto*. Haettu 20.10.2012 osoitteesta <http://agilemanifesto.org>
- Ahire, L. S., Golhar, Y. D. & Waller, A. M. (1996). Development and Validation of TQM Implementation Constructs. *Decision Sciences*, 27(1), 23-56.
- Alleman, G.B., Henderson, M. & Seggelke, R. (2003). Making Agile Development Work in a Government Contracting Environment Measuring Velocity with Earned Value. *Teoksessa Proceedings of the Agile Development Conference 2003, ADC 2003, Salt Lake City, UT, USA, June 25-28* (s.114-119). Washington, DC, USA: IEEE Computer Society.
- Anderson, D. (2010). *Kanban – Successful evolutionary change for your technology business*. USA: Blue Hole Press.
- Bannock , G., Baxter, R. E. & Davis, E. (1998). *The Penguin Dictionary of Economics, 6th edition*. New York, NY, USA: Penguin Books.
- Beaumont, S. (2008). *Agile & Contracts*. Haettu 19.11.2012 osoitteesta <http://www.coactivate.org/projects/agile-contracts/presentations/Agile-Contracts-at-Scrum-Gathering.pdf>
- Beck, K. (1999a). *Extreme Programming Explained: Embrace Change*. Boston, MA, USA: Addison-Wesley.
- Beck, K. (1999b). Embracing Change with Extreme Programming. *Computer* 32(10), 70-77.
- Beck, K. & Andres, C. (2004). *Extreme Programming Explained: Embrace Change, 2nd edition*. Boston, MA, USA: Addison-Wesley.
- Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, 21(5), 61-72.
- Boehm B., (2002). Get Ready for the Agile Methods, With Care. *Computer*, 35(1), 64-69.
- Book, M., Gruhn, V. & Streamer, R. (2012). adVantage : A Fair Pricing Model for Agile Software Development Contracting. *Teoksessa C.Wohlin (toim.),*

- Proceedings of the 13th International Conference on Agile Processes in Software Engineering and Extreme Programming, XP 2012, Malmö, Sweden, May 21-21* (s.193-200). Berlin: Springer-Verlag.
- Cabri, A. & Griffiths, M. (2006). Earned Value and Agile Reporting. Teoksessa J.Chao, M.Cohn, F.Maurer, H.Sharp & J.Shore (toim.), *Proceedings of the AGILE 2006 Conference, Minneapolis, Minnesota, USA, July 23-28* (s.17-22). Washington, DC, USA: IEEE Computer Society.
- Cockburn, A. (2002). *Agile software development*. Boston, MA, USA: Addison-Wesley.
- Cohen, D., Lindvall, M. & Costa P. (2004). An Introduction to Agile Methods. Teoksessa M.Zelkowitz (toim.), *Advances in Computers, volume 62: Advances in Software Engineering* (s.1-66). Amsterdam: Elsevier.
- Cohn, M. (2005). *Agile estimating and planning*. Upper Saddle River, NJ USA: Prentice-Hall.
- Conboy, K. (2009). Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development. *Information Systems Research*, 20(3), 329-354.
- Dybå, T. & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information Software Technology*, 50(9-10), 833-859.
- Eckfeldt, B., Madden, R. & Horowitz, J. (2005). Selling Agile : Target-Cost Contracts. Teoksessa *Proceedings of the Agile Development Conference 2005, AGILE 2005, Denver, CO, USA, July 24-29* (s.160-166). Washington, DC, USA: IEEE Computer Society.
- Erickson, J., Lyytinen, K. & Siau, K. (2005). Agile Modeling, Agile Software Development and Extreme Programming: The State of Research. *Journal of Database Management*, 16(4), 88-100.
- Favaro, J. M. (2005). That Elusive Business Value: Some Lessons from the Top. Teoksessa H.Baumeister, M.Marchesi & M.Holcombe (toim.), *Proceedings of the 6th International Conference on Extreme Programming and Agile Processes in Software Engineering, XP 2005, Sheffield, UK, June 18-23* (s.199). Berlin, Heidelberg: Springer.
- Fehr, E., Klein, A., & Schmidt, K. M. (2004). *Contracts, Fairness and Incentives*. Haettu 25.11.2011 osoitteesta http://epub.ub.unimuenchen.de/334/1/Contracts_Fairness_and_Incentives.pdf
- Franklin, T. (2008). Adventures in Agile contracting: evolving from time and materials to fixed price, fixed scope contracts. Teoksessa *Proceedings of the Agile Conference 2008, AGILE '08, Washington, DC, USA, August 4-8* (s.269-273). Washington, DC, USA: IEEE Computer Society.
- Fraser, S., Martin, A., Biddle, R., Hussman, D., Miller, G., Poppendieck, M., Rising, L. & Striebeck, M. (2004). The Role of the Customer in Software Development: the XP Customer – Fad or Fashion? Teoksessa *Proceedings of the Companion to the 19th annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications, OOPSLA 2004, Vancouver, BC, Canada, October 24-28* (s. 148-150). New York, NY, USA: ACM.

- Gilb, T. (1988). *Principles of Software Engineering Management*. Boston, MA, USA: Addison-Wesley.
- Griffin, L. A. (2001). A Customer Experience: Implementing XP. Teoksessa D.Wells (toim.), *Proceedings of the XP Universe 2001, Raleigh, NC, USA, July 23-25* (s.195-200). Boston, MA, USA: Addison-Wesley.
- Grisham, P. S. & Perry, D. E. (2005). Customer relationships and Extreme Programming. *ACM SIGSOFT Software Engineering Notes*, 30(4), 1-6.
- Hanssen, G. K. & Fægri, T. E. (2006). Agile Customer Engagement - a Longitudinal Qualitative Case Study. *Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering, ISESE 2006* (s.164-173). New York, NY, USA: ACM.
- Harris, M., Herron, D. & Iwanicki, S. (2008). *The business value of IT: managing risks, optimizing performance, and measuring results*. Boca Raton, FL, USA: Auerbach Publications.
- Highsmith, J. (2000). *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*. New York, NY, USA: Dorset House.
- Highsmith, J. (2002). What Is Agile Software Development? *CrossTalk: The Journal of Defence Software Engineering*, 15(10), 4-9.
- Highsmith, J. & Cockburn, A. (2001). Agile Software Development: The Business of Innovation. *Computer*, 34(9), 120-122.
- Hoch, D., Roeding, C., Purkert, G., Lindner, S. & Miller, R. (1999). *Secrets of Software Success*. Boston, Harvard Business School Press.
- Hoda, R., Noble, J. & Marshall, S. (2009). Negotiating Contracts for Agile Projects: A Practical Perspective. Teoksessa P.Abrahamsson, M.Marchesi & F.Maurer (toim.), *Proceedings of the 10th International Conference on Agile Processes in Software Engineering and Extreme Programming, XP2009, Pula, Sardinia, Italy, May 25-29* (s. 186-191). Berlin: Springer.
- Hoda, R., Noble, J. & Marshall, S. (2010). Agile undercover: when customers don't collaborate. Teoksessa A.Sillitti, A.Martin, X.Wang, & E.Whitworth (toim.), *Proceedings of the Agile Processes in Software Engineering and Extreme Programming, 11th International Conference, XP 2010, Trondheim, Norway, June 1-4* (s.73-87). Berlin: Springer.
- Hunt, A. & Thomas, D. (1999). *The Pragmatic Programmer: From Journeyman to Master*. Boston, MA, USA: Addison-Wesley.
- Hyvärinen, H., Hulkko, P. & Ohvo, S. (2002). *Yksityisoikeuden perusteet*. Porvoo: WSOY.
- Ilieva, S., Ivanov, P. & Stefanova, E. (2004). Analyses of an agile methodology implementation. Teoksessa *Proceedings of the 30th Euromicro Conference* (s.326-333). Washington, DC, USA: IEEE Computer Society.
- Issac, G., Rajendran, C. & Anantharaman, R. N. (2004). A conceptual framework for Total Quality Management in software organizations. *Total Quality Management & Business Excellence*, 15(3), 307-344.
- Kautz, K. (2009). Customer and User Involvement in Agile Software Development. Teoksessa P.Abrahamsson, M.Marchesi & F.Maurer (toim.), *Proceedings of the Agile Processes in Software Engineering and Extreme Programming*,

- 10th International Conference, XP 2009, Pula, Sardinia, Italy, May 25-29 (s. 168-173). Berlin: Springer.
- Kniberg H. & Skarin M. (2009). *Kanban and Scrum – making the most of both*. USA: InfoQ.
- Korkala, M., Pikkarainen, M. & Conboy, K. (2009). Distributed Agile Development: A Case Study of Customer Communication Challenges. Teoksessa P.Abrahamsson, M.Marchesi & F.Maurer (toim.), *Proceedings of the 10th International Conference on Agile Processes in Software Engineering and Extreme Programming, XP2009, Pula, Sardinia, Italy, May 25-29* (s. 161-167). Berlin: Springer.
- Koskela, J. & Abrahamsson, P. (2004). On-Site Customer in an XP Project: Empirical Results from a Case Study. Teoksessa T.Dingsøy (toim.), *Proceedings of the 11th European Conference on Software Process Improvements, Euro-SPI 2004, Trondheim, Norway, November 10-12* (s.1-11). Berlin: Springer.
- Lantz, K. (1986). *The Prototyping Methodology*. Upper Saddle River, NJ, USA: Prentice-Hall.
- Larman, C. & Vodde, B. (2009). *Scaling Lean & Agile Development – Thinking and Organizational Tools for Large-Scale Scrum*. Boston, MA, USA: Addison-Wesley.
- Lean Enterprise Institute (2009). *Principles of Lean*. Haettu 16.11.2012 osoitteesta <http://www.lean.org/whatslean/principles.cfm>
- Liker, J. K. (2010). *Toyotan tapaan*. Helsinki: Readme.fi.
- Lohan, G., Lang, M. & Conboy, K. (2010). Having a customer focus in agile software development. Teoksessa J.Pokorny, V.Repa, K.Richta, W.Wojtkowski, H.Linger, C.Barry & M.Lang (toim.), *Information Systems Development, Business Systems and Services: Modeling and Development* (s. 441-453). New York: Springer.
- Mann, C. & Maurer, F. (2005). A case study on the impact of scrum on overtime and customer satisfaction. Teoksessa *Proceedings of the Agile Development Conference 2005, AGILE 2005, Denver, CO, USA, July 24-29* (s.70-79). Washington, DC, USA: IEEE Computer Society.
- Markus, M. L. & Mao, J. Y. (2004). Participation in Development and Implementation - Updating An Old, Tired Concept for Today's IS Contexts. *Journal of the Association for Information Systems*, 5(11-12), pp. 514-544.
- Martin, A., Biddle, R. & Noble, J. (2004a). The XP customer role in practice: three studies. Teoksessa *Proceedings of the Agile Development Conference, AGILE 2004, Salt Lake City, UT, USA, June 22-26* (s.42-54). Washington, DC, USA: IEEE Computer Society.
- Martin, A., Biddle, R. & Noble, J. (2004b). When XP met outsourcing. Teoksessa J.Eckstein & H.Baumeister (toim.), *Proceedings of the 5th International Conference on Extreme Programming and Agile Processes in Software Engineering, XP 2004, Garmisch-Partenkirchen, Germany, June 6-10* (s.55-59). Berlin, Springer-Verlag.
- Melville, N., Kraemer, K. & Gurbaxani, V. (2004). Information Technology and Organizational Performance: An Integrative Model of IT Business Value. *MIS Quarterly*, 28(2), 283-322.

- Miller, G. (2001). The Characteristics of Agile Software Processes. Teoksessa *Proceedings of the 39th International Conference and Exhibition of Object-Oriented Languages and Systems (TOOLS 39)*, Santa Barbara, CA, USA, July 29 – August 3 (s.385-387). Los Alamitos, California, USA: IEEE Computer Society.
- Mohammadi, S., Nikkhahan, B. & Sohrabi, S. (2009). Challenges of user Involvement in Extreme Programming projects. *International Journal of Software Engineering and Its Applications*, 3(1), 19-32.
- Müller, M. M. & Padberg, F. (2003). On the Economic Evaluation of XP Projects. *Proceedings of the 9th European Software Engineering Conference held jointly with the 11th ACM SIGSOFT International Symposium on Foundations of Software Engineering, Helsinki, Finland, September 1-5* (s.168-177). New York, NY, USA: ACM.
- Mumford, E. (1983). *Designing Human Systems for New Technology - The ETHICS Method*. Manchester, UK: Manchester Business School.
- Mutschler, B., Zarvic, N. & Reichert, M. (2007). *A Survey on Economic-driven Evaluations of Information Technology (TR-CTIT-07)*. University of Twente.
- Patton, J. (2008). Ambiguous Business Value Harms Software Products. *IEEE Software*, 25(1), 50-51.
- Pinheiro, C., Maurer, F. & Sillito, J. (2008). Adopting Iterative Development: The Perceived Business Value. Teoksessa P. Abrahamsson, R.Baskerville, K.Conboy, B.Fitzgerald & X.Wang (toim.), *Proceedings of the 9th International Conference on Agile Processes in Software Engineering and Extreme Programming, XP 2008, Limerick, Ireland, June 10-14* (s.185-189). Berlin: Springer.
- PMI Standards Committee (1996). *A Guide to the Project Management Body of Knowledge*. Newton Square, PA, USA: Project Management Institute.
- Poppendieck, M. & Poppendieck T. (2003). *Lean software development: an agile toolkit*. Boston, MA, USA: Addison-Wesley.
- Porter, M. E. (1985). *Competitive Advantage: Creating and Sustaining Superior Performance*. New York, The Free Press.
- Racheva, Z., Daneva, M. & Sikkel, K. (2009). Value creation by agile projects: methodology or mystery? Teoksessa F.Bomarius, M.Oivo, P.Jaring & P.Abrahamsson (toim.), *Proceedings of the 10th International Conference on Product-Focused Software Process Improvement, PROFES 2009, Oulu, Finland, June 15-17* (s. 141-155). Berlin: Springer.
- Racheva, Z., Daneva, M., Sikkel, K., Herrmann, A. & Wieringa, R. (2010a). Do we Know Enough about Requirements Prioritization in Agile Projects: Insights from a Case Study. Teoksessa RE '10 *Proceedings of the 18th IEEE International Requirements Engineering Conference* (s. 147-156). Washington, DC, USA: IEEE Computer Society.
- Racheva, Z., Daneva, M., Sikkel, K. & Buglione, L. (2010b). Business Value Is Not Only Dollars: Results from Case Study Research on Agile Software Projects. Teoksessa *Proceedings of the 11th International Conference on Product-Focused Software Process Improvement, PROFES 2010, Limerick, Ireland, June 21-23* (s.131-145). Berlin: Springer-Verlag.

- Rawsthorne, D. (2004). *Managing the Work in an Agile Project*. Haettu 12.12.2012 osoitteesta
<http://www.netobjectives.com/files/resources/downloads/ManagingTheWork.pdf>
- Rawsthorne, D. (2006). *Calculating Earned Business Value For An Agile Project*. Haettu 13.12.2012 osoitteesta
http://www.danube.com/system/files/CollabNet_WP_Earned_Business_Value_041910.pdf
- Schwaber, K. (1995). Scrum Development Process. Teoksessa J.Sutherland, D.Patel, C.Casanave, G.Hollowell & J.Miller (toim.), *Business Object Design and Implementation: OOPSLA'95 Workshop Proceedings* (s.117-134). London, Springer-Verlag.
- Schwaber, K. (2004). *Agile Project Management with Scrum*. Redmond, Washington, USA: Microsoft Press.
- Schwaber, K. & Sutherland, J. (2011). *Scrum Guide*. Haettu 15.11.2012 osoitteesta
http://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum_Guide.pdf
- Stevens, P. (2009a). *Contracting for Agile Software Projects, Part 1*. Haettu 31.1.2013 osoitteesta
<http://agilesoftwaredevelopment.com/blog/peterstev/contracting-agile-software-projects>
- Stevens, P. (2009b). *10 Contracts for your next Agile Software Project*. Haettu 16.1.2013 osoitteesta
<http://agilesoftwaredevelopment.com/blog/peterstev/10-agile-contracts>
- Sulaiman, T., Barton, B. & Blackburn, T. (2006). AgileEVM – Earned Value Management in Scrum Projects. Teoksessa J.Chao, M.Cohn, F.Maurer, H.Sharp & J.Shore (toim.), *Proceedings of the AGILE 2006 Conference, Minneapolis, Minnesota, USA, 23-28 July* (s.7-16). Washington, DC, USA: IEEE Computer Society.
- Sutherland, J. (2008). *Agile Contracts: Money for Nothing and Your Change for Free*. Haettu 28.11.2012 osoitteesta
<http://jeffsutherland.com/Agile2008MoneyforNothing.pdf>
- Sutherland, J. (2010). *Scrum Handbook*. Haettu 10.12.2012 osoitteesta
<http://jeffsutherland.com/scrumhandbook.pdf>
- Svenbrant, H. (2008). *A process analysis of customer value in agile projects*. Haettu 3.2.2013 osoitteesta
http://www.pm.lth.se/fileadmin/pm/Exjobb/Exjobb_2008/Svenbrant_H._2008__A_process_analysis_of_customer_value_in_agile_projects_02.pdf
- Takeuchi, H. & Nonaka, J. (1986). The new new product development game. *Harvard Business Review*, January-February 1986, 137-146.
- Taloussanasto (1998). *Taloussanasto*. Helsinki: Taloustieto Oy.
- Thorup, L. & Jensen, B. (2009). Collaborative Agile Contracts. Teoksessa Y.Dubinsky, T.Dybå, S.Adolph & A.S.Sidky (toim.), *Proceedings of the Agile Development Conference 2009, AGILE 2009, Chigago, IL, USA, August 24-28* (s.195-200). Washington, DC, USA: IEEE Computer Society.

- Van Cauwanberghe, P. (2003a). *Agile Fixed Price Projects part 1: "The Price Is Right"*. Haettu 25.11.2012 osoitteesta
<http://www.nayima.be/html/fixedpriceprojects.pdf>
- Van Cauwanberghe, P. (2003b). *Agile Fixed Price Projects part 2: "Do you want agility with that?"*. Haettu 25.11.2012 osoitteesta
<http://www.nayima.be/html/agilefixedprice.pdf>
- Van Deursen, A. (2001). Customer involvement in extreme programming: XP2001 workshop report. *ACM SIGSOFT Software Engineering Notes*, 26(6), 70-73.
- VersionOne (2011). *State of Agile Development Survey 2011*. Haettu 1.12.2012 osoitteesta
http://www.versionone.com/pdf/2011_State_of_Agile_Development_Survey_Results.pdf
- Wang, X., Wu, Z. & Zhao, M. (2008). The Relationship Between Developers and Customers in Agile Methodology. Teoksessa *Proceedings of the 2008 International Conference on Computer Science and Information Technology, ICCSIT 2008* (s. 566-572). Washington, DC, USA: IEEE Computer Society.
- Warsta, J. (2001). *Contracting in Software Business*. Väitöskirja. Oulun Yliopisto.
- Wieggers, K. E. (1999). *First Things First: Prioritizing Requirements*. Haettu 8.1.2013 osoitteesta
<http://www.tarrani.net/linda/prioritizing.pdf>
- Womack, J., Jones, D. & Roos, D. (1990). *The Machine That Changed the World*. New York: Rawson Associates.
- Womack, J., Jones, D. & Roos, D. (1996). *Lean Thinking, Banish Waste and Create Wealth in your Corporation*. New York: Touchstone Books.