



**LAUREA**  
AMMATTIKORKEAKOULU  
*Yhdessä enemmän*

Ketterät menetelmät:  
Muutos vesiputousmallista  
ketteriin menetelmiin siirryttäessä  
kansainvälisissä  
ohjelmistokehitysorganisaatioissa

Törnqvist, Pauliina

Laurea-ammattikorkeakoulu  
Otaniemi

Ketterät menetelmät: Muutos vesiputousmallista  
ketteriin menetelmiin siirryttäessä kansainvälisissä  
ohjelmistokehitysorganisaatioissa

Pauliina Törnqvist  
Liiketalous  
Opinnäytetyö  
Huhtikuu, 2015

Pauliina Törnqvist

**Ketterät menetelmät: Muutos vesiputousmallista ketteriin menetelmiin siirryttäessä kansainvälisissä ohjelmistokehitysorganisaatioissa**

2015

Sivumäärä 45

---

Opinnäytetyön lähtökohtana oli ohjelmistokehitysalalla puhaltavat uudet tuulet. 70-luvun alkupuolella esitelty vaiheistettu ohjelmistokehitysmenetelmä vesiputousmalli oli pitkään ohjelmistokehityksen vallitseva projektihallintamuoto. 90-luvun alkupuolella sen rinnalle syntyi uudenlaisia kevyitä kehitysmenetelmiä, jotka pyrkivät ratkaisemaan vesiputousmallin mukaisen vaiheittaisen kehitystavan tuottamat ongelmat. Nämä ketteriksi menetelmiksi nimittetyt projektinhallintamenetelmät korostivat vaiheittaisen kehityksen sijasta iteratiivista kehitysmallia, ja mahdollistivat suunnitelmanmuutokset kehitysvaiheessa, mikä toi ohjelmistokehitykseen sen kaipaamaa joustoa.

Ketterät menetelmät kasvoivat nopeasti hyvin suosituiksi. Moni ohjelmistoja kehittävä organisaatio on käynyt 2000-luvulla läpi raskaan muutosprosessin kun vesiputousmallista on siirrytty ketterien menetelmien pariin. Muutos on sekä yksilön että organisaation toimintaa vahvasti muovaava prosessi, jonka avulla organisaatiot ovat pyrkineet nopeuttamaan ohjelmistojen kehityskaarta, saamaan kehitystyöhön joustoa, helpottamaan riskienhallintaa, tekemään paremmin asiakkaiden muuttuvia tarpeita vastaavia tuotteita ja mahdollistamaan suunnitelmanmuutoksiin reagoinnin.

Tämän opinnäytetyön tarkoituksena oli tarkastella tätä muutosprosessia kolmessa kansainvälisessä ohjelmistoja kehittävässä suurorganisaatiossa, sekä pohtia miten ketterien menetelmien arvolupaukset ovat toteutuneet muutoksen jälkeen, ja mitkä ovat onnistuneen muutoksen avaintekijöitä. Opinnäytetyön empiirinen tutkimustyö toteutettiin haastatteleamalla jokaisesta kolmesta organisaatiosta edustajaa, ja analysoimalla haastatteluista saatua materiaalia vertaamalla sitä teoreettiseen tietoon ketteristä menetelmistä.

Haastatteluiden perusteella ketteriin menetelmiin siirtymisessä oleellisinta on ymmärtää, että muutos koskettaa koko organisaatiota. Jotta muutoksella saavutetaan halutut tulokset, niin prosessien, ihmisten kuin työkalujenkin on muututtava. Haastateltujen organisaatioiden edustajien mukaan ketterien menetelmien arvolupaukset ovat toteutuneet muutoksen jälkeen hyvin, mutta ne ovat tuottaneet myös haasteita ja tuoneet mukanaan paljon uutta opeteltavaa. Kaikki kolme ovat kuitenkin yhtä mieltä siitä, että muutos on ollut organisaation toiminnan kannalta oikean suuntainen.

Haastatteluista saadun materiaalin, ja siitä tehtyjen johtopäätösten avulla opinnäytetyön toiminnallisena tuotoksena syntyi karkea opas vesiputousmallista ketteriin menetelmiin siirtymiseen. Oppaan ei ole tarkoitus olla täydellinen ohjekirja muutosohjelmaan, vaan se nostaa esiin asioita, jotka haastateltujen organisaatioiden edustajat kokivat oleellisiksi onnistuneen muutoksen takaamiseksi. Oppaassa on käytetty teoreettisena viitekehyksenä Dean Leffingwellin teosta Agile Software Requirements: Lean requirements practices for teams programs and the enterprise (2014).

Avainsanat: ketterät menetelmät, vesiputousmalli, ohjelmistokehitys, muutosprosessi

Pauliina Törnqvist

**Agile software development methods: The transition from waterfall model to agile methods in international software development companies**

2015

Pages

45

---

The background of this bachelor's thesis was the new era of software development. Presented in the early 70's, the phased waterfall model was for long the most common software development method. In the early 90's new lightweight development methods were presented, that were to solve the problems that waterfall's phased development cycle had started to cause many developers. These new methods - known as the Agile software development methods - emphasized iterative development over phased model, and allowed developers to react to changes of plans, which brought the long needed flexibility for the development.

Agile methods grew very popular fast, and many software development companies have gone through a heavy transition process in the 21<sup>st</sup> century when adopting agile methods and relinquishing the waterfall model. The transition is a process that affects the operations of both the organization and an individual. The aim is to achieve faster development life-cycle, make the development process more flexible, ease risk management, make products that answer the customers' actual needs, and allow developers to react to changes of plans.

The aim of this thesis was to study this transition process in three international large-scale software development companies, and reflect how the value proposition of agile methods have realized after the transition, and what the key factors in a successful agile transition are. The empirical research of this thesis was done by interviewing a representative that has been a part of the transition process, from each company, and analyzing the material gotten from the interviews by comparing it to the theoretical information about agile methods.

According to the interviewed companies, the most important thing to understand in the agile transformation is that the whole organization changes. In order to achieve the best outcome the processes, people, and tools need to change. The company representatives all agreed that the value proposition of agile methods has realized well after the transition, but the new methods have also brought many challenges and new things to learn. Overall they all think that the transition has been a step into the right direction.

The functional outcome of this thesis is a short guide for transitioning from waterfall to agile based on the materials and conclusions made from the interviews. The purpose of this guide is not be a thorough manual for the transition, but to point out the things and factors that the interviewed company representatives found to be the most crucial ones in order to execute the transition successfully. Dean Leffingwell's book *Agile Software Requirements: Lean requirements practices for teams programs and the enterprise* (2014), was used as theoretical framework for the guide.

**Keywords:** agile software development methods, waterfall model, software development, transition process

## Sisällys

1	Johdanto.....	6
2	Opinnäytetyön tavoitteet, tarkoitus ja tutkimuskysymykset.....	7
3	Ohjelmistokehitysmenetelmät.....	8
3.1	Vesiputousmalli.....	8
3.1.1	Toimintaperiaate.....	8
3.1.2	Vesiputousmallin ongelmat.....	10
3.2	Ketterät menetelmät.....	10
3.2.1	Ketterän ohjelmistokehityksen julistus.....	11
3.2.2	Menetelmät ja niiden yhteispiirteet.....	12
3.2.3	Ketterien menetelmien arvolupaukset.....	14
3.3	Vesiputousmallin ja ketterien menetelmien erot.....	14
4	Muutosprosessi.....	16
5	Opinnäytetyön toteutus ja kohdeorganisaatiot.....	17
5.1	Toteutus.....	17
5.2	Kohdeorganisaatiot.....	18
6	Tulokset.....	20
6.1	Muutoksen käynnistämisen syyt ja ajankohta.....	20
6.2	Muutosprosessin toteutustapa ja kesto.....	21
6.3	Muutos työympäristössä ja työyhteisössä.....	22
6.4	Ketterien menetelmien tuomat hyödyt yksilö- ja organisaatiotasolla.....	23
6.5	Ketterien menetelmien tuomat haitat yksilö- ja organisaatiotasolla.....	23
6.6	Ketterien menetelmien arvolupauksien toteutuminen.....	24
6.7	Tutkimustulosten luotettavuuden arviointi.....	26
7	Yhteenveto ja johtopäätökset.....	27
8	Toiminnallinen tuotos.....	30
9	Opinnäytetyön arviointi.....	31
	Liitteet.....	35

## 1 Johdanto

Ohjelmistokehityksen juuret ovat aina 1940-luvulla saakka, jolloin ensimmäiset ohjelmoitavat tietokoneet kehitettiin. Ohjelmistojen kehitys oli pitkään etukäteen suunniteltujen ohjelmistojen tuottamista, mutta teknologian kehittyessä ja ympäristön muuttuessa vaatimukset ohjelmistoille alkoivat muuttua nopeammin kuin projektit valmistuivat. Tämä hankaloitti ohjelmoijien ja ohjelmistotuottajien työskentelyä, ja ilmiön seurauksena syntyi uudenlaisia projektinhallintamenetelmiä ohjelmistokehitystä varten. (Webopas 2013.) Perinteisten metodien, joissa koko kehitystyö ja tuotteen ominaisuudet suunniteltiin etukäteen, rinnalle kehittyi niin kutsuttuja ketteriä menetelmiä, jotka mahdollistivat suunnitelmanmuutokset kehitysvaiheessa. Näiden uusien metodien perustana oli, että ohjelmistokehitys on jatkuvasti projektin aikana elävää ja olosuhteisiin sopeutuvaa, sekä kehitystiimi on monitoiminnallinen (crossfunctional) itsenäinen ryhmä.

Ketterät ohjelmistokehitysmenetelmät kasvoivat nopeasti hyvin suosituiksi, ja tänä päivänä lähes kaikki suurimmat ohjelmistoyritykset ovat siirtyneet käyttämään niitä. Tällä hetkellä menetelmiä on kymmenkunta, ja kaikkia näitä yhdistävät samat periaatteet ja arvot. Jokainen niistä on kuitenkin suunniteltu erityyppisiin projekteihin ja projektien eri vaiheisiin.

Perinteisistä kehitysmenetelmistä ketteriin menetelmin siirtyminen on organisaation sisällä raskas ja pitkä prosessi. Jotta tämä muutos käynnistetään, on yrityksen tiedostettava nopean ja evoluutionomaisen kehityksen tarve sekä punnittava muutoksen mahdollisia hyötyjä ja haittoja. Prosessin onnistuminen vaatii sekä organisaatiolta että yksilöiltä pitkäjänteisyyttä, avoimuutta ja muutosvastarinnan käsittelemistä. Uusi menetelmä tuo mukanaan uuden opettelua ja vanhoista kaavoista irtautumista, joka voi suuressa organisaatiossa viedä vuosia.

Tämä opinnäytetyö tarkastelee perinteisistä menetelmistä ketteriin menetelmiin siirtymistä kolmessa kansainvälisessä ohjelmistojä tuottavassa suurorganisaatiossa ja summaa uuden kehitysmenetelmän tuomia hyötyjä ja haittoja sekä pohtii ketterien menetelmien arvolupauksien toteutumista muutoksen jälkeen. Materiaalin avulla vielä perinteisiä menetelmiä käyttävät yritykset voivat punnita ketteriin menetelmiin siirtymisen kannattavuutta ja helpottaa siirtymäprosessiaan sekä sen suunnittelua. Tutkimustyö pohjautuu teoreettiseen tietoon ketteristä menetelmistä ja niiden arvolupauksista sekä muutosprosessijohtamiseen niiden käyttöönotossa. Teoreettista tietoa verrataan opinnäytetyötä varten tehdyistä haastatteluista saatuun materiaaliin ja haastateltujen henkilöiden kokemuksiin, ja pohditaan mitä yhtäläisyyksiä ja eriävyyksiä niissä on.

## 2 Opinnäytetyön tavoitteet, tarkoitus ja tutkimuskysymykset

Tämän opinnäytetyön tavoitteena on selvittää, minkälainen prosessi vanhoista kehitysmenetelmistä ketteriin menetelmiin siirtyminen on ollut kolmessa kansainvälisessä suuressa ohjelmistojen kehittävässä organisaatiossa. Tarkoituksena on myös avata yksilöhaastatteluiden kautta kokemuksia muutosprosessista ja ketterien menetelmien tuomia hyötyjä sekä haittoja, ja luoda tähän tietoon pohjautuen karkea opas ketteriin menetelmiin siirtymistä varten.

Oleellisimmat tutkimuskysymykset ovat: 1) miten muutosprosessi vesiputousmallista ketteriin menetelmiin on toteutettu ja viety läpi kansainvälisissä ohjelmistoalan organisaatioissa, ja 2) miten muutos toteutetaan onnistuneesti. Opinnäytetyö pyrkii vastaamaan näihin kysymyksiin kattavasti, jolloin se palvelee sekä haastateltavia yrityksiä, että vielä vesiputousmallia käyttäviä yrityksiä. Ketterien menetelmien arvolupauksien toteutumisen pohtiminen avaa muutoksen ja sen läpikäymisen kannattavuutta, jolloin ketterien menetelmien hyötyjä ja haittoja on helpompi arvioida. Muutosprosessin toteutus ja johtaminen antavat kuvan prosessin läpikäymisestä sekä siitä mitä tapoja sen toteuttamiseen on kansainvälisissä yrityksissä käytetty.

### 3 Ohjelmistokehitysmenetelmät

Ohjelmistokehitys on terminä hyvin laaja, ja sillä voidaan yleisesti viitata lähes mihin tahansa toimintoon jonka tavoitteena on tuottaa tietokoneelle koodattu ohjelma tai useista ohjelmista koostuva ohjelmisto. Kehitys tapahtuu kirjoittamalla sarjan toisiinsa liittyviä ohjelmointikoodeja ja -käskyjä jotka tietokoneen suoritin toteuttaa haluttujen toimenpiteiden aikaansaamiseksi. (Techopedia 2015.)

Ohjelmistojen kehityksen yhteydessä käytetään lähes aina projektinhallintamenetelmiä. Varsinkin jos kehitystiimi koostuu useammasta kuin yhdestä henkilöstä, on työskentelylle ideaalista antaa viitekehys. Nämä ohjelmistokehitysmenetelmiksi nimitettävät projektinhallintatavat määrittelevät usein projektin eri vaiheet ja tiimin toimintatavat, budjetin, tavoiteltavan laadun sekä aikataulun. Ohjelmistokehitysmenetelmiä on lukuisia erilaisia, ja yleisesti tunnetuista metodeista on syntynyt yhtä laaja kirjo variaatioita, kuin on ammattilais- ja vapaa-ajan ohjelmointitiimejäkin.

Tunnetuimpia näistä ohjelmistokehitysmetodeista ovat vesiputousmalli sekä ketterät menetelmät. Vesiputousmalli, joka juontaa juurensa 1970-luvulle, oli pitkään lähes kaikkien ohjelmistokehittäjien yleinen toimintaohje, ja sitä sekä sen variaatioita nimitetäänkin usein perinteisiksi menetelmiksi (Waterfall Model 2015). Vesiputousmallin rinnalle alkoi kuitenkin 2000-luvun vaihteessa syntyä uusia kehitysmenetelmiä, sillä teknologian kehittyessä yhä nopeammin, ja ohjelmistojen vaatimuksien muuttuessa nopeammin kuin projekteja saatiin valmiiksi, vesiputousmallille ominainen vaiheistettu kehitystapa ei enää palvellut suurta osaa ohjelmistokehittäjistä optimaalisesi (Agile Methodology 2014).

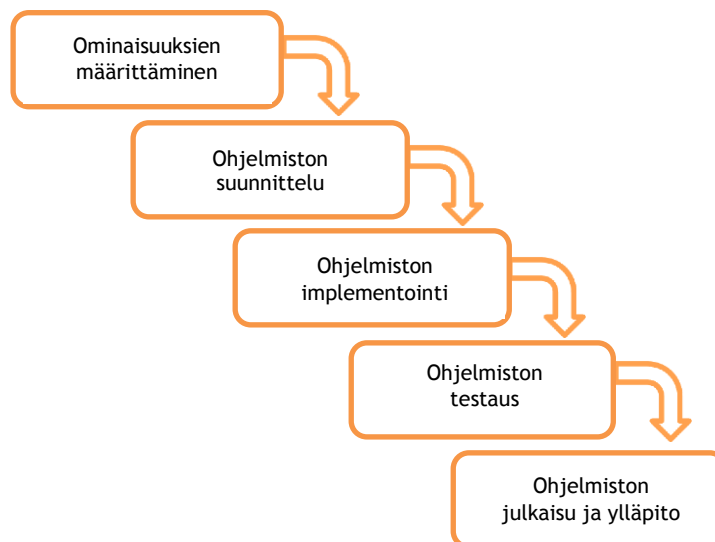
#### 3.1 Vesiputousmalli

##### 3.1.1 Toimintaperiaate

Vesiputousmalli on Winston Roycen vuonna 1970 esittämä ohjelmistokehitysmenetelmä. Vesiputousmallille ominaista on lineaarinen vaiheittainen kehitys, joka etenee vesiputouksen tavalla vaiheesta seuraavaan kunnes lopputuotteena syntyy toimiva ohjelma tai ohjelmisto. Jokaisella kehitysvaiheella on tarkat tavoitteet, mutta itse vaiheet ovat määriteltä hyvin geneerisesti, joten Vesiputousmalli voidaan adaptoida hyvinkin monenlaisiin organisaatioihin ja projekteihin. (Waterfall Model 2015.)



Ennen vesiputousmallia ohjelmistoprojektit kärsivät usein sekavuudesta ja rakenteen puutteesta, ja siitä syystä vesiputousmallilla pyrittiinkin tuomaan selkeyttä ohjelmistojen tekoon (Waterfall Model 2015). Vesiputous-nimi tulee mallin vaiheista:



Kaava 1: Vesiputousmalli

Vesiputousmalli jakaa ohjelmistokehityksen vaiheisiin. Vaiheet luovat projektille strukturoidun kulun, jossa keskitytään vuorollaan kaikkiin ohjelmistojen rakentamiseen tarvittaviin toimiin. Oleellista on huomata, että kaaviossa edetään alaspäin, mutta ei taaksepäin. Projekti etenee mallin mukaisesti vaiheesta toiseen, ja kun yhdestä vaiheesta siirrytään seuraavaan, ei edellisiin enää palata myöhemmin.

Vesiputousmallissa on viisi vaihetta:

- Ominaisuuksien määrittäminen: Projektia aloitettaessa tehdään taustatutkimusta aiheesta, määritellään tarkasti minkälainen ohjelmisto/ohjelma halutaan tehdä ja arvioidaan budjettia. Tämän jälkeen projektille ja ohjelmistolle tehdään kattava suunnitelma.
- Suunnittelu: Kun projektille on tehty kattava suunnitelma, on seuraava askel luoda ohjelmistolle perussuunnittelu. Kun perustat on saatu tehtyä ja hyväksytty, tehdään ohjelmistolle yksityiskohtaisempi ja teknisesti kattavampi suunnittelu. Tämä kattaa muun muassa osien funktioiden määrittämisen.

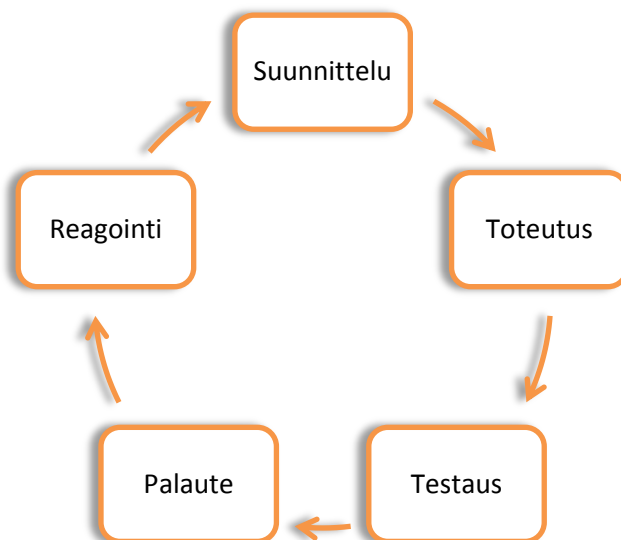
- Implementointi: Kun ohjelmiston suunnittelutyöt ovat valmiit, siirrytään itse ohjelmiston rakennukseen. Rakennusvaiheessa luodaan lähdekoodi, josta ohjelmisto koostuu.
- Testaus: Kun ohjelmisto on rakennettu, sen funktionaalinen toimivuus testataan, ja varmistetaan että ohjelmisto vastaa vaatimuksia. Mikäli ohjelmakoodi tai suunnittelu ovat puutteellisia tai virheellisiä, se ilmenee projektin tässä vaiheessa.
- Julkaisu ja ylläpito: Mikäli ohjelmisto läpäisee testauksen, se voidaan ottaa käyttöön. Projekti siirtyy ylläpitovaiheeseen, jolloin ohjelmistoa ei enää aktiivisesti rakenneta, vaan sitä ylläpidetään varmistamalla, että se toimii yhä halutulla tavalla.

### 3.1.2 Vesiputousmallin ongelmat

Vesiputousmallin suurin hyöty ohjelmistokehittäjälle on, että se tekee projektista systemaattisen ja strukturoidun. Kun kehityksen eri toimet jaetaan omiin vaiheisiinsa, pysyy projektin rakenne selkeänä, ja kehittäjän on helppo edetä. Vesiputousmallin ongelmana on kuitenkin ennalta tehty suunnitelma, josta poikkeamista, esimerkiksi paluuta suunnitteluun, on vaikeaa hallita projektissa. Vesiputousmalli olettaa, että tuotteen kaikki vaatimukset ovat etukäteen tiedossa, mutta suuret ohjelmistoprojektit saattavat kestää vuosia. Teknologian kehittyessä sekä asiakkaiden muuttuvien tarpeiden vuoksi, vesiputousmallia käyttävät kehittäjät huomasivat, että ohjelmiston ominaisuudet ja vaatimukset saattoivat muuttua nopeammin kuin projektit valmistuivat. Tämä saattoi johtaa siihen, että suuri osa tehdystä työstä oli turhaa. Muutoksiin reagoiminen ei ollut käytännössä mahdollista suunnitelmien lukkoon lyömisen jälkeen, jolloin monet pitkät projektit tuottivat lopulta todella paljon turhia ominaisuuksia ja turhaa koodia. Tämä ongelma aiheutti monille yrityksille päänvaivaa ja konkreettista taloudellista tappiota, ja monet ohjelmistokehittäjät kritisoivat perinteisiä ohjelmistokehitysmenetelmiä liian hitaiksi ja byrokraattisiksi. (Leffingwell 2014, 6.)

### 3.2 Ketterät menetelmät

2000-luvun alkupuolella perinteisten menetelmien rinnalle nousi joukko uudenlaisia ohjelmistokehitysmenetelmiä. Sen sijaan, että olisi Vesiputousmallin tavoin korostettu etukäteen suunnittelua ja vaadittujen ominaisuuksien yksityiskohtaista listaamista, nämä uudet menetelmät keskittyivät painottamaan jatkuvaa suunnittelua, monitoiminnallisia tiimejä, asiakasosallistumista, jatkuvaa testaamista, sekä tärkeimpänä lyhyissä nopeissa iteraatioissa tapahtuvaa kehitystä. (Waters 2007.)



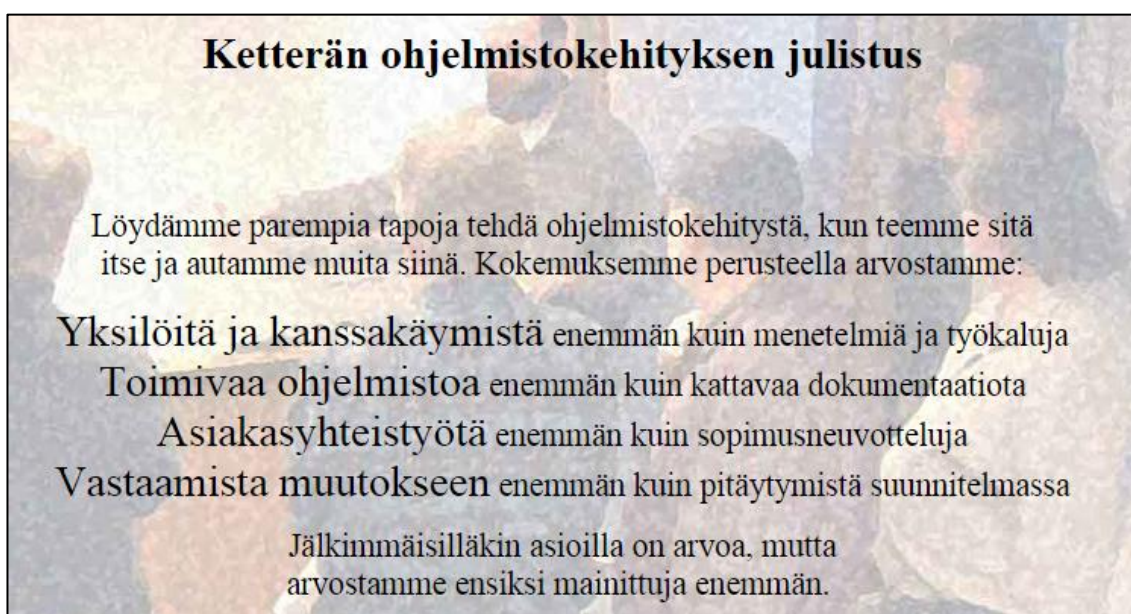
Kaava 2: Ketterä kehitysmalli

Ketterä kehitys sisältää samat toiminnot kuin vesiputousmallikin, mutta erottavana tekijänä menetelmien välillä on kehitystapa. Siinä missä vesiputousmalli jakaa kehityksen vaiheisiin, ketterässä kehityksessä toistetaan lyhyissä (yleensä noin 2-4 viikon) iteraatioissa kaikki toiminnot, jonka jälkeen tuotos arvioidaan, ja sen ominaisuuksia joko poistetaan, lisätään tai priorisoidaan uudelleen seuraavaa iteraatiota varten. Ohjelmisto rakentuu siis lyhyissä kierroksissa joissa lisätään aina ominaisuuksia ja pohditaan asiakkaan kanssa ovatko tarpeet muuttuneet. Jokaisen iteraation päätteeksi tiimillä tulisi olla valmiina toimiva ohjelmisto, ja iteraatioita toistetaan niin monta kertaa, että kaikki halutut ominaisuudet on saatu valmiiksi ja implementoitua.

### 3.2.1 Ketterän ohjelmistokehityksen julistus

Helmikuussa 2001 17 ohjelmistokehittäjää kokoontuivat Utahissa keskustelemaan ohjelmistokehitysmenetelmistä ja niiden ongelmista. Nämä ansioituneet ohjelmistokehittäjät tiedostivat perinteisen kehityksen ongelmat, ja pyrkivät luomaan pohjaa uudellisille menetelmille jotka ratkaisisivat vesiputousmallimaisen kehityksen puutteet. Tuloksena syntyi Ketterän ohjelmistokehityksen julistus; ensimmäinen dokumentoitu yhteenveto ketteristä kehitystavoista. Julistus listasi ketterien menetelmien perusarvot, ja toimi sateenvarjomaisena yläkäsitteenä uusille kevyille ohjelmistokehitysmenetelmille.

Ketterän ohjelmistokehityksen julistuksen sisältö on seuraavanlainen:

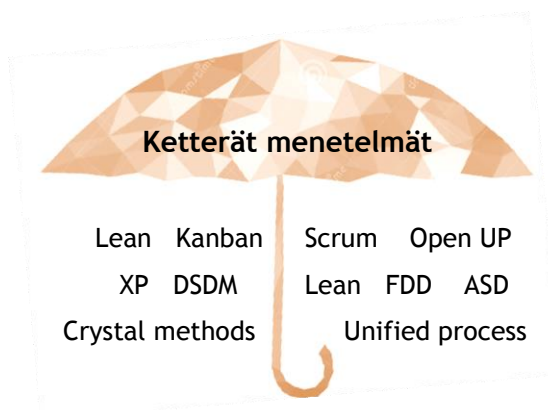


Kuva 1: Ketterän ohjelmistokehityksen julistus

Ketterän ohjelmistokehityksen julistuksen tarkoituksena oli luoda pohja ketterille menetelmille ja ketterälle kehitykselle, sekä näin edistää ketterän ajattelun leviämistä. Menetelmiä on julistuksen julkaisun jälkeen syntynyt useita erilaisia, mutta kaikilla niillä ovat yhteiset arvot ja periaatteet. Ketterän kehittämisen edistämistä ajaa Suomessa Agile Finland Ry, ja kansainvälisesti Agile Alliance (Agile Finland Ry 2014).

### 3.2.2 Menetelmät ja niiden yhteispiirteet

Ketteriä menetelmiä on 2000-luvun alkupuolen jälkeen syntynyt useita erilaisia, ja niillä on kaikilla omat erityispiirteensä. Osa menetelmistä keskittyy koko projektin elinkaareen, kun taas jotkut ainoastaan kehitysvaiheeseen. Tunnetuimpia menetelmiä ovat Scrum, Extreme Programming, Dynamic Systems Development Method ja Kanban. Vaikka kaikki ketterät menetelmät ovat keskenään erilaisia ja keskittyvät eri asioihin, on tärkeää ymmärtää, että niillä kaikilla on kuitenkin samat perusarvot ja periaatteet.



Kuva 2: Ketterien menetelmien sateenvarjokuva

Ketterä kehitys on yläkäsite, joka määrittää kaikkien menetelmien arvot ja periaatteet. Arvojen tavoin myös periaatteet listattiin ketterien menetelmien manifestissa, ja ne ovat seuraavanlaiset:

1. Tärkein prioriteetti on tyydyttää asiakas toimittamalla hyödyllinen ja toimiva ohjelmisto usein ja toistuvasti
2. Vaatimusten muutokset myöhäisessäkin kehitysvaiheessa ovat sallittuja ja jopa toivottuja
3. Toimita toimiva ohjelmisto usein, ideaalisesti muutaman viikon välein
4. Kaupallisten ja teknisten ihmisten tulee työskennellä yhdessä koko projektin ajan
5. Rakenna projektit motivoituneiden ihmisten ympärille, tue heitä siinä mitä he tekevät, ja luota, että työ tulee tehtyä
6. Paras tapa välittää tietoa kehitystiimille on keskustella kasvoistusten
7. Toimiva ohjelmisto on ensisijainen edistymisen mittari
8. Ketterät prosessit vaativat kestävästä kehitystä
9. Jatkuva teknisen laadun huomiointi parantaa ketteryyttä
10. Yksinkertaisuus on elintärkeää
11. Parhaat designit, ominaisuudet, suunnitelmat ja ideat nousevat monitoiminnallisista tiimeistä
12. Tasaisin väliajoin tiimi kokoontuu retrospektiiviseen tapaamiseen, jossa pohditaan kuinka asiat voisi tehdä paremmin, tehokkaammin tai nopeammin. (Agile Manifesto 2001.)

Nämä periaatteet ja arvot toistuvat kaikissa ketterissä menetelmissä. Vaikka kaikki ketterät menetelmät keskittyvät pääasiassa ratkaisemaan saman ongelman, on oikean menetelmän valinta haastavaa ja tärkeää. Jotta menetelmä palvelee parhaiten tiimiä ja projektia, on tunnettava menetelmien erot ja osattava valita juuri kyseiseen projektiin parhaiten sopiva

lähestymistapa. Mikäli tiimi valitsee väärän menetelmän, saattaa se pahimmassa tapauksessa vain hidastaa ja hankaloittaa työskentelyä.

### 3.2.3 Ketterien menetelmien arvolupaukset

Ketterät menetelmät nousivat suosioon nopeasti. Suuri syy tähän oli niiden takaamat hyödyt eli arvolupaukset, jotka ilmaisivat ketterän kehityksen poistavan monia vesiputousmallin ongelmia. Nämä arvolupaukset ovat seuraavanlaisia:

Nopeampi kehityskaari

Riskit pienempiä

Tekemisen läpinäkyvyys

Muutoksiin on helpompi reagoida

Tuote vastaa paremmin asiakkaan tarpeita. (Versionone 2015.)

### 3.3 Vesiputousmallin ja ketterien menetelmien erot

Jotta voidaan tarkastella vesiputousmallista ketteriin menetelmiin siirtymistä, on tärkeää ymmärtää menetelmien erot. Näin on helpompi myös lähestyä syitä siirtymisen taustalla, ja saada kokonaisvaltaisempi ymmärrys siitä, miksi muutos on nähty välttämättömäksi. Vesiputousmallin ja ketterien menetelmien vertailun helpottamiseksi niiden pääpiirteet on koottu seuraavaan taulukkoon:

Vesiputousmalli	Ketterät menetelmät
Vaiheittainen kehitystapa	Iteratiivinen kehitystapa
Omat tiimit kullekin kehitysvaiheelle	Monitoiminnalliset tiimit läpi koko projektin
Suuret tiimikoot	Pienet tiimikoot
Suunnitelmanmuutokset projektin alkamisen jälkeen vaikeita	Suunnitelmanmuutokset projektin alkamisen jälkeen tyypillisiä
Lopputuote vastaa hyvin alun perin suunniteltua, mutta ei välttämättä asiakkaan muuttuneita tarpeita	Lopputuote saattaa olla hyvin erilainen kuin alun perin on suunniteltu
Priorisoi laatua	Priorisoi nopeutta
Yksityiskohtainen kattava dokumentaatio läpi koko projektin, painottuen projektin alkuun	Vähäinen dokumentaatio kehitystyön aikana
Asiakkaan vähäinen osallistuminen kehitystyöhön	Asiakkaan vahva osallistuminen kehitystyöhön

Taulukko 1: Vesiputousmallin ja ketterien menetelmien vertailu

Vaikka vesiputousmalli saattaa vaikuttaa vanhanaikaiselta ohjelmistonkehitysmenetelmältä, ei se sitä ole. Ketterät menetelmät pyrkivät ratkaisemaan ongelman, joka syntyi kun

rakennettavien ohjelmistojen vaatimukset ja halutut ominaisuudet muuttuivat nopeammin kuin projektit valmistuivat. Tällaiseen tilanteeseen joudutaan usein, kun rakennetaan asiakkaalle laajaa ohjelmistoa ja projekti on kestoaltaan pitkä. Vesiputousmalli on kuitenkin toimiva projektinhallinnan viitekehys kun tiedetään tarkkaan minkälainen lopputuotteesta halutaan, ja ollaan varmoja, etteivät halutut ominaisuudet muutu projektin edetessä. Mikäli tällaiseen projektiin valittaisiin jokin ketteristä menetelmistä, saattaisivat ne sekoittaa ja hidastaa työskentelyä turhalla aikalaatikoimisella ja palavereilla. Ketterät menetelmät tuovat useihin ohjelmistoprojekteihin tarvittavaa joustoa, mutta on täysin projektista ja projektityypistä riippuvaista kumpi menetelmä soveltuu käyttöön paremmin.

#### 4 Muutosprosessi

Muutosprosessi on termi joilla viitataan yleisesti siihen kun jokin asia muutetaan systemaattisesti toiseen. Yritys- ja organisaatiokontekstissa tällä tarkoitetaan usein organisaation sisällä tapahtuvaa muutosta, ja tämä opinnäytetyö viittaa sillä ohjelmistokehitysmenetelmän vaihtumiseen perinteisestä vesiputousmallista ketteriin menetelmiin.

Muutoksen lähtökohtana on yleensä organisaation oma toiminta ja siinä havaitut muutostarpeet. Muutoksen suunnittelussa on otettava huomioon sekä koko organisaation tavoittelema hyöty, että yksilötason edut. Ennen muutoksen käynnistämistä on tärkeää varmistaa muutoksen tarpeellisuus. (Luomala 2008, 4 - 6.) On tärkeää tiedostaa selkeästi mikä ongelma muutoksella pyritään ratkaisemaan, minkälainen toivottu lopputulos on, ja mitä toimenpiteitä muutoksen onnistuminen vaatii. Muutos sisältää riskejä ja haasteita, mutta samaan aikaan se tarjoaa paljon kiinnostavia mahdollisuuksia jotka saattavat helpottaa niin yksilön kuin organisaationkin jokapäiväistä toimintaa.

Useat suuret ohjelmistokehitysalan yritykset ovat viime vuosikymmenen aikana käyneet läpi muutosprosessin, jonka myötä vanhoista projektinhallintamenetelmistä on siirrytty ketteriin menetelmiin. Muutoksen haastavuus ja prosessin pituus on pitkälti riippuvainen organisaation koosta, sillä vanhojen kaavojen muuttaminen ja uusien menetelmien läpiajo vie huomattavasti kauemmin aikaa suuremmissa tiimeissä ja organisaatioissa. Pienemmät yritykset pystyvät kokeilemaan uusia menetelmiä yksittäisissä projekteissa helposti, mutta suuressa organisaatiossa tämä vaatisi henkilöstön uudelleen ryhmittämistä (tyypillisesti esimerkiksi ohjelmoijat toimivat omassa tiimissään, graafikot omassa tiimissään ja testajat omassaan). Ketterissä menetelmissä kuitenkin korostetaan, että työtiimit ovat monitoiminnallisia ja muista tiimeistä riippumattomia. Näin ollen suuren organisaation ryhtyessä muutokseen on kaikki tiimit rakennettava uudelleen. Pienessä yrityksessä koko kehitystiimi saattaa toimia muutenkin yhdessä. (Versionone 2012.)

Ohjelmistokehitysmenetelmän muutoksessa organisaation on tiedostettava kehitysmenetelmän vaihtamisen tarve, ja visioitava selkeästi mihin toimintoihin halutaan parannusta. Useimpien ohjelmistokehittäjien kohdalla ongelmana oli vesiputousmallin vaiheittainen kehityssykli, jossa suunnitelmanmuutokset eivät onnistuneet enää projektin alkamisen jälkeen. Tämän tuloksena monet tekivät paljon turhaa työtä, ja projektit kestivät niin kauan, että kun ohjelmisto lopulta valmistui, oli se jo valmiiksi vanhentunut tai se ei vastannut asiakkaan muuttuneita tarpeita. (Leffingwell 2014, 6.) Ketterien menetelmien myötä tähän ongelmaan löytyi kuitenkin ratkaisu ja ne ovat käynnistäneet lukuisissa organisaatioissa - niin pienissä kuin suurissakin - muutosprosessin jonka tuloksena koko



organisaation toiminta on uudelleenmuotoiltu taidoista, rooleista, asenteista, työrytmistä ja prosesseista aina työkaluihin asti.

## 5 Opinnäytetyön toteutus ja kohdeorganisaatiot

### 5.1 Toteutus

Opinnäytetyön empiirinen tutkimustyö toteutettiin haastattelemalla kolmen suuren kansainvälisen ohjelmistoja tuottavan organisaation edustajia yksilöllisesti yrityksiensä omissa tiloissa. Haastattelut nauhoitettiin ja ne litteroitiin jälkikäteen. Saatu aineisto analysoitiin ja läpikäytiin, ja siitä tehtiin summaava yhteenveto. Lopullinen työ toimitettiin kaikille haastatteluihin osallistuville henkilöille.

Syy haastattelumenetelmän valintaan esimerkiksi kyselylomakkeen sijasta oli, että haastattelemalla henkilöt pystyvät kertomaan laajemmin näkemyksensä, sekä haastattelijalla on mahdollisuus esittää tarkentavia kysymyksiä. (Hirsijärvi & Hurme 2000, 35.)

Haastattelu oli muodoltaan teemahaastattelu, ja haastattelukysymykset muotoiltiin avoimiksi, jolloin niihin saatiin herkemmin tunteisiin ja kokemuksiin perustuvia vastauksia. Vaikka opinnäytetyö pyrkii antamaan laajan kuvan muutosprosessista organisaatiotasolla, haluttiin tutkimustyö tehdä kvalitatiivisena, jolloin haastatteluista saatu aineisto oli kokemuksiin perustuvaa ja aiheeseen voidaan pureutua syvemmin. (Tuomi & Sarajärvi 2009.) Näin työhön saatiin yksilötason ajatuksia ja tunteita ohjelmistokehitysmenetelmän vaihtumisesta. Tämä opinnäytetyö ei pyri yleistykseen, vaan tarkastelemaan muutosprosessia kohdeyrityksissä, ja luomaan haastatteluista saadun tiedon perusteella yhteenvetoon ketterisiin menetelmiin siirtymiseen liittyvistä asioista.

Litteroitu haastattelumateriaali analysoitiin sisällön teoriaohjaavaa analyysiä käyttäen. Analyysiin sisältyy teoreettista tietoa ketteristä menetelmistä ja niiden arvolupauksista, mutta pääasiassa analyysissä lähdettiin liikkeelle hyvin aineistopohjaisesti haastatteluista saatuihin materiaaleihin nojautuen. Kustakin haastattelusta saatua materiaalia ryhmiteltiin ja luokiteltiin teorian ohjaamana, ja niistä pyrittiin rakentamaan loogisia asiakokonaisuuksia. Näin tutkimusilmiötä tarkasteltiin aineistolähtöisesti, mutta lopputulos rakentui teoreettiseksi kokonaisuudeksi (Tuomi & Sarajärvi 2009, 95-98).

Lähestymistapoina toimivat sekä konstruktiivinen tutkimus että toimintatutkimus. Tämä opinnäytetyö tarkastelee yksittäisten yritysten toimintatapoja ja lähtee liikkeelle kartoittamalla muutosprosessin käynnistymisen taustaa ja aiheen teoreettista pohjaa, mutta lopputuloksena syntyy tiivis yhteenveto muutosprosessin läpikäynnistä ja sen tuomista

hyödyistä ja haitoista näissä yksittäisissä yrityksissä. Alussa keskityttiin siis enemmän lähestymään aihetta toimintatutkimuksen kantilta, kun taas lopputuotos on enemmän konstrukttiivinen kooste ketterien menetelmien arvolupauksien toteutumisesta ja menetelmien tuomista hyödyistä ja haitoista suurorganisaatioiden toiminnassa.

Yksittäisistä yrityksistä ei tehty erikseen tapauskertomuksia, jotta työ ei paisunut liian suureksi, ja jotta työhön osallistuvat yritykset säilyttävät mahdollisimman hyvin yksityisyytensä. Haastatteluista saatua materiaalia oli myös helpompi jäsentää ja luokitella, kun tietoja ei yhdistetty tiettyyn kertomukseen, vaan ne pidettiin yleisellä tasolla.

## 5.2 Kohdeorganisaatiot

Opinnäytetyön tarkastelun kohteena toimi kolme suurta kansainvälistä ohjelmistojä tuottavaa yritystä. Yrityksiä valittiin kolme, sillä opinnäytetyö pyrki tarkastelemaan ketterien menetelmien käyttöönottoa ja niiden hyötyjä ja haittoja yleisellä tasolla suurissa yrityksissä kokemuksiin nojautuen, sen sijaan että keskityttäisiin yksittäiseen yritykseen. Kolmea saman tason yritystä ja heidän menettelytapojaan vertailemalla saatiin kattavampi kuva prosessista ja sen vaatimista toimenpiteistä. Haastatteluista saatu materiaali oli yritysten ollessa samaa kokoluokkaa vertailukelpoisempaa. Kohteeksi valittiin suuret, alan johtavat yritykset, sillä niiden muutosprosessijohtaminen ja prosessin läpiviennin metodit ovat alan parhaita ja tehokkaimpia. Keskittyessä suuriin yrityksiin prosessista saatiin myös kaikista kattavin kuva, sillä mitä suurempi organisaatio viedään läpi muutosprosessista, sitä todennäköisempää on, että muutos ei ole ollut yksinkertainen toteuttaa, ja se on vaatinut asian punnitsemista useammalta kannalta.

Opinnäytetyöhön haastatellut yritykset ja niiden edustajat jätettiin heidän toiveestaan työhön anonyymeiksi. Tulosten purkamisen selventämiseksi yrityksiä nimitetään työssä seuraavasti:

Yritys 1

Yritys 2

Yritys 3.

Haastatellut henkilöt ovat olleet oleellisessa roolissa kehitysmenetelmän muutosprosessissa. Jotta saatiin kattava kuva muutokseen johtaneista syistä, muutoksen suunnittelusta ja toteutuksesta sekä sen tuoneista hyödyistä ja haitoista, oli haastateltavien henkilöiden tiedettävä aiheesta laajasti.

Yrityksen 1 edustaja toimi muutosohjelman aikaan organisaatiossa tiimipäällikkönä, ja hän osallistui muutosohjelmassa organisaation rakenteen ja toimintamallin suunnitteluun ja toteutukseen. Hän on työskennellyt ohjelmistokehitysalalla 16 vuotta, ja yrityksessä 16 vuotta.

Yrityksen 2 edustaja on työskennellyt teollisten ohjelmistojen parissa 22 vuotta, ja yrityksessä 22 vuotta. Hän toimi johtajana yksikössä jossa muutosohjelma toteutettiin, ja on tällä hetkellä yrityksen 2 yhden liiketoimintadivisioonan globaalien palveluliiketoiminnan teknologiajohtaja.

Yrityksen 3 edustaja toimi organisaation muutosohjelman vetäjänä vastaten yhden tuotekehitysyksikön kokonaisvaltaisesta uudistamisesta ketterillä prosesseilla ja menetelmillä. Kyseinen muutoshanke oli aikanaan yksi laajimmista koskaan tehdyistä tuoteorganisaatioiden ketteröittämissankkeista ja sen piirissä oli lähes 2000 ihmistä ohjelmistoinsinööreistä johtajiin. Henkilö on toiminut yli 20 vuotta ohjelmistotalalla suunnittelu-, laadunvarmistus- ja operatiivisen toiminnankehityksen tehtävissä sekä asiantuntijana että johtotehtävissä. Hän on myös osallistunut Scaled Agile Frameworkin kehittämiseen erityisesti vaatimustenhallinnan osalta.

## 6 Tulokset

### 6.1 Muutoksen käynnistämisen syyt ja ajankohta

Yritys 1 käynnisti muutosprosessin vuonna 2007. Syitä vesiputousmallista luopumiselle ja ketteristä menetelmistä kiinnostumiselle oli monia, mutta niistä päällimmäisinä nousivat haastattelussa esiin johdon uusi osaaminen ja tuotekehityksestä nousseet ongelmat. Organisaation tuotekehitysyksikön program-taso oli jatkuvasti ongelmissa, sillä tuotteiden vaatimukset muuttuivat jatkuvasti, ja tehdyt projektisuunnitelmat pitivät vain muutaman viikon. Projektipäälliköt kokivat hyvin työlääksi ja turhauttavaksi tehdä samoille projekteille jatkuvasti uusia projektisuunnitelmia kun ominaisuuksien vaatimukset vaihtelivat jopa viikoittain. Ketterät menetelmät nähtiin tähän ongelmaan ideaalisena ratkaisuna. Organisaation johto oli saanut samoihin aikoihin perehdytystä ketteristä menetelmistä, ja niihin siirtyminen nähtiin loogisena askeleena tuotekehityksen työskentelyn ja tuottavuuden parantamiseksi.

Yritys 2 aloitti ketterien menetelmien parissa vuonna 2010. Lähtöasetelmana tähän oli, että organisaatiossa havaittiin suuren määrän asiakaslisäarvosta ja sen tekemisestä kärsivän siitä, että perusasioita jouduttiin tekemään toistuvasti uudestaan. Yrityksen tuoteyksiköt olivat melko pieniä, ja nähtiin, että jos nämä tuoteyksiköt joutuvat rakentamaan omassa käytössään olevat ohjelmistot ja työkalut itse, ei aikaa jää juurikaan muulle työskentelylle. Tämän ongelman ratkaisemiseksi organisaatiolle perustettiin oma platform-yksikkö joka tekisi sisäisten tuoteyksiköiden käyttöön tarvittavia ohjelmistoja ja työkaluja. Platform-yksikkö toimisi ”avoin lähdekoodi”-periaatteella. Sen avulla pyrittiin saamaan ketteryttä asiakaslisäarvon tuottamiseen kohdentamalla tekeminen ja resurssit oikein.

Yritys 3 aloitti ketteriin menetelmiin siirtymisen vuonna 2009. Syitä siirtymisen taustalla olivat halu keskittyä vahvemmin asiakkaan käyttökokemukseen, sekä parantaa työnteon laatua ja nopeutta. Organisaatiossa haluttiin siirtää suurin fokus teknologiasta käyttökokemukseen, ja uskottiin, että keskittymällä oikeisiin asioihin saataisiin ne tehtyä nopeammin ja samalla parempia käyttäjäkokemuksia. Yrityksen tuotekehitys oli ylikuormittunut ilmeisesti vääränlaisen fokuksen vuoksi ja ketterien menetelmien sekä oikeanlaisen muutosohjelman avulla fokus saataisiin oikeisiin asioihin ja tekemiseen saataisiin tarvittua joustavuutta ja nopeutta. Toivottiin, että samalla myös laatu paranisi.

## 6.2 Muutosprosessin toteutustapa ja kesto

Yritys 1 toteutti muutoksen siirtämällä noin 1,5 vuoden aikana vaiheittain kaikki tuotekehitys-yksiköt käyttämään ketteriä menetelmiä. Jokaisen yksikön muutos aikataulutettiin yksilöllisesti. Tästä seurasi pieniä ongelmia, koska saman tuotteen parissa työskentelevät eri yksiköt siirtyivät pois vesiputousmallista eri aikatauluissa, jolloin jotkut saattoivat tehdä kehitystyötä jo iteratiivisesti, kun taas toiset vielä vaiheittain. Noin puolessatoista vuodessa kaikki tuotekehitys-yksiköt olivat siirtyneet käyttämään ketteriä menetelmiä. Muutoksen jälkeen kesti noin vuosi, että menetelmä oli jalkautettu ja kehityksen voitiin sanoa olevan kunnolla ketterää. Tästä huolimatta koetaan, että prosessi ei todellisuudessa ole vielä ohi organisaatiotasolla. Tuotekehitys-yksiköt siirrettiin ketterien menetelmien käyttöön, mutta muu organisaatio on pääasiassa edelleen perinteisten menetelmien parissa, jolloin voidaan sanoa ettei täydellinen muutos ole tähänkään päivään mennessä valmis.

Yritys 2 otti ketterät menetelmät käyttöön platform-yksikkönsä perustamisen yhteydessä. Uuden yksikön ja sen toiminnan mahdollistamiseksi organisaatio keräsi yhteen oikeanlaiset teknologiat ja loi kehityksen tueksi hallinnointimallit teknologia- ja tuotekehityksen etenemissuunnitelman päättämiseksi. Organisaatioon luotiin uuden yksikön myötä myös teknologia-lautakunta joka määrittää investointiallokaation; kuinka paljon tehdään perusohjelmistoja ja -työkaluja, kuinka paljon tehdään inkrementaalista kehitystä ja mikä on tuotevalikoima. Teknologia-lautakunta määrittää myös kuinka paljon resursseja käytetään esimerkiksi ylläpitoon, ja kyseenalaistaa kerran puolessa vuodessa onko suunta kehityksen kannalta strategisesti. Tämän avulla organisaatio priorisoi työkokonaisuuksien listaa ja kehitystyötä oikeanlaiseksi säilyttääkseen mahdollisimman hyvän kytköksen oikeisiin kaupallisiin tarpeisiin. Käyttöön otettiin aikajaksoitettu kehitysmalli, jossa julkaisu tapahtui kaksi kertaa vuodessa. Näiden aikajaksojen sisällä työskentely tapahtui jatkuvan julkaisun kehitysmallilla. Ketterien menetelmien omaksumiseen ja jalkauttamiseen kului aikaa noin vuosi, mutta yrityksen 1 tapaan myös yritys 2 painottaa, että käytännössä prosessi ei ole vielä täysin ohi.

Yritys 3 käytti muutosohjelmassaan Dean Leffingwellin Scaled Agile Framework-mallia. Organisaatiossa nähtiin, että ketterään kehitykseen siirtyessä tarvitaan enemmän kuin ohje siitä kuinka puolen tusinan hengen tiimi toimii, joten he toteuttivat muutosohjelmansa Leffingwellin kehittämän suuryrityksille tarkoitetun ketterään kehitykseen siirtymisen prosessinhallintaohjelman avulla. Dean Leffingwell toimi muutosohjelman pääkonsulttina ja koulutti organisaation muutosohjelmavastaavien kanssa esimerkiksi paljon organisaation johtoa. Yritys piti tärkeänä, että muutosohjelmassa olisi selkeä operatiivinen visio siitä missä ollaan ja mihin muutoksella pyritään, ja Leffingwellin SAFe-malli nähtiin parhaiten palvelevana ja loogisimpana ratkaisuna. Muutosohjelman aikana kaikki tiimit kävivät läpi

ketterän kehityksen koulutuksen ja muutoksen avainhenkilöt kokoontuivat viikoittain arvioimaan saavutuksia ja suunnittelemaan tulevaa viikkoa, sekä päätöksenteko siirrettiin henkilöille jotka olivat parhaiten perillä käsiteltävästä asiasta. Organisaatiolla ei ollut pilottiprojekteja jotka olisivat lähteneet harjoittamaan ketteriä menetelmiä, vaan kaikki projektit siirrettiin omalla aikataulullaan sujuvasti ketteriin menetelmiin. Muutosprosessia käsiteltiin kuten monimutkaista ketterää ohjelmistoprojektia ottaen siihen ketterän toteutustavan. Muutosohjelma oli hyvin strukturoitu ja tarkasti aikataulutettu; itse muutokseen käytettiin vuosi, ja muutoksen vahvistamiseen toinen vuosi.

### 6.3 Muutos työympäristössä ja työyhteisössä

Yrityksessä 1 muutoksen myötä työtiloja ja tiimejä jouduttiin muuttamaan ja järjestelemään uudelleen. Kun käytössä oli vielä vesiputousmalli, suurimmalla osalla ihmisistä oli omat työhuoneet ja monet työskentelivät hyvin itsenäisesti jopa hieman eristäytyneenä. Ketterien menetelmien ja toiminnan tehostuksen myötä siirryttiin tiimitiloihin ja avokonttoreihin. Tiimien koot pienenevät ketteriin menetelmiin siirryttäessä ja niihin syntyi paljon uusia rooleja. Tämän lisäksi käyttöön tuli paljon uusia työkaluja ja vanhoja poistettiin niiden tieltä. Joillekin organisaation työntekijöille muutos oli hyvin vaikea, sillä uudenlainen työskentelytapa pakotti kaikki työskentelemään yhdessä. Jotkut olivat nauttineet nimenomaan yksin omissa työhuoneissa työskentelystä. Mutta ulospäin suuntautuneille, tiimityöstä nauttiville henkilöille muutos oli positiivinen. Uuden rakenteen hyväksymisen helpottamiseksi työntekijät saivat valita itse oman tiiminsä ja tiimikumppaninsa kun niitä rakennettiin uudelleen, ja tämä oli yleisesti hyvin pidetty asia organisaation sisällä.

Yritys 2 rakensi integroituja tiimejä tiettyjen teknologiakomponenttien ympärille luomalla virtuaaliset kehitysympäristöt. Haasteena oli, että kehitystyö tapahtui käytännössä kokonaan Euroopassa, ja testaus Intiassa. Virtuaalisten kehitysympäristöjen avulla tiimit istuvat ikään kuin samassa huoneessa, vaikka todellisuudessa ovatkin fyysisesti eri puolilla maailmaa. Kulttuurillisesti tämä oli hyvin uutta, ja ajoittain esimerkiksi hierarkkinen maailmankatsomus oli hyvin vaihtelevaa pienenkin tiimin sisällä. Ennen muutosta toiminta oli ollut hyvin maakohtaista. Ketterien menetelmien myötä vastuu siirtyi esimiehiltä tiimeille ja työarkeen tuli paljon uusia muodollisia seremonioita. Nämä seremoniat ja tapaamiset herättivät alkuun paljon kyseenalaistusta, mutta kun ymmärrys ketteristä menetelmistä kasvoi, ymmärrettiin myös esimerkiksi päivittäisten tiimikokousten kuuluvan arkeen.

Myös yrityksessä 3 muokattiin tiimejä uudelleen. Ideana oli, että tiimeistä tulisi itsejärjestäytyneitä ja vastuu siirtyisi esimiehiltä tiimeille. Muutoksen myötä tiimeihin syntyi paljon uusia rooleja, sekä vanhat olemassa olevat roolit muuttuivat. Tämän lisäksi tuli uusia taitoja opeteltavaksi. Asenne haluttiin sellaiseksi, että halutaan oikeasti kerralla tehdä hyvää

jälkeä. Ihmisten asenteiden ja roolien muuttuessa myös hallintatavat ja prosessit muuttuivat. Gant-kaavioiden sijaan seinät peittyivätkin työkokonaisuuksien listoista ja priorisointilistoista, ja projektin etapit korvattiin iteraatioilla ja kehitysjaksoilla. Muutos vaikutti käytännössä kaikkiin ja kaikkeen organisaatiossa, ja se oli joillekin työntekijöille raskasta. Syklinen kehitys toi paremmin näkyviin jos joku ei pysynyt tahdissa. Kun vesiputousmallin aikaan kun tämä heikkous tuli esiin noin kerran vuodessa, tuli se nyt ketterien menetelmien myötä esiin parin kuukauden välein.

#### 6.4 Ketterien menetelmien tuomat hyödyt yksilö- ja organisaatiotasolla

Yrityksessä 1 ketterien menetelmien myötä yrityksen tuotekehityksen työskentelyyn on saatu kaivattua joustoa. Suunnitelmia ei tarvitse enää tehdä uudelleen uusien vaatimusten ilmaantuessa, vaan ne yksinkertaisesti lisätään työkokonaisuuksien listaan ja priorisoidaan jo olemassa olevien vaatimusten tapaan. Suurin hyöty ketterien menetelmien käyttöönotossa on ollut se, että pystytään suunnittelemaan tekemistä pienemmissä osissa. Työskentelystä tuli joustavampaa, ja kun tiimikoot pienentyivät muutoksen myötä, turha yleiskuormitus väheni. Tekemisen tehokkuus parani myös ketterien menetelmien myötä, ja resurssit saatiin hyödynnettyä paremmin.

Yrityksessä 2 ollaan ketterien menetelmien myötä saavutettu säästöjä organisaatiotasolla asti, sekä tehdyille tuotteille ja ohjelmistoille ollaan saatu parempi uudelleenkäyttöaste. Pällekkäistä tekemistä on vähemmän, ja työskentelyn läpinäkyvyys on parantunut merkittävästi. Tämä läpinäkyvyys on parantanut myös aikataulujen ennustettavuutta joka on ollut suureksi hyödyksi ulkoisessa toiminnassa asiakasrajapinnassa sekä sisäisessä kehityksessä. Yritys 2 kokee myös työskentelyn tehokkuuden, lopputuotteiden käyttäjäpalautteen ja virheiden havaitsemisasteen parantuneen ketterien menetelmien käyttöönoton myötä.

Yritys 3 näkee saavuttaneensa paremman fokuksen työskentelyyn ketterien menetelmien kautta. Uuden menetelmän myötä ymmärrettiin keskittyä siihen mikä on tärkeää ja mikä vähemmän tärkeää. Tämän lisäksi työnlaatu parani selvästi, ja vikamäärät laskivat. Suurena hyötynä nähdään myös parantunut työnopeus ja yleisen innostuneisuuden kasvu.

#### 6.5 Ketterien menetelmien tuomat haitat yksilö- ja organisaatiotasolla

Yritys 1 kokee ketterien menetelmien hyötyjen jäävän odotettua matalammiksi, sillä ainoastaan organisaation ohjelmiston suunnittelu, toteutus ja testaus on siirtynyt niiden käyttöön. Ketteriä menetelmiä on vaikeaa ja jopa turhauttavaa käyttää, kun esimerkiksi myynti tapahtuu kuitenkin olettaen, että kehitys on vesiputousmallimaista ja kaikki

ominaisuudet etukäteen suunniteltuja. Ajoittain ketterien menetelmien pakolliset seremoniat myös hankaloittivat työskentelyä kun muodollisia tapaamisia pidettiin vaikka niille ei välttämättä olisi tarvetta tai aikaa. Tämän lisäksi yksilötason haittoina nähtiin päivittäisen raportoinnin kasvu.

Yritys 2 on kokenut ketterät menetelmät lähes yksinomaan positiivisena asiana. Vaikka iteratiivinen kehitysmalli oli alkuun hankala sopeuttaa yksikköön, jonka perinteinen kehitysmalli on pitkään ollut vesiputoustyyppinen, nähtiin tämä ennenkin haasteena kuin negatiivisena asiana. Sekä yksilö- että organisaatiotason hyödyt ovat olleet korkeat, kun taas haittoja ei juurikaan ole havaittu.

Yritys 3 näkee ketterien menetelmien käyttöönoton haittapuolina muutosohjelman hintavuuden. Uusien kehitystapojen omaksuminen vaati paljon henkilöstön koulutusta ja organisaatiotason toimenpiteitä jotka olivat kalliita. Tämän lisäksi haittapuolena on koettu joidenkin yksilöiden kohdalla kasvanut paineen määrä. Muuten muutos ja uusi kehitysmenetelmä olivat positiivisia asioita ja ketteristä menetelmistä oltiin hyvin innostuneita.

## 6.6 Ketterien menetelmien arvolupauksien toteutuminen

Yrityksessä 1 ketterien menetelmien arvolupauksien koetaan toteutuneen osittain hyvin ja osittain heikosti. Tuotteiden ei koeta vastaavan asiakkaiden tarpeita aikaisempaa paremmin. Vesiputousmallin aikaan asiakkaalle tehtiin juuri se tuote joka suunniteltiin, mutta ketterän kehityksen myötä kehitys tapahtuu tekemällä usealle asiakkaalle tuotteita yhtä aikaa. Tällöin jokainen saa jotain, mutta kukaan ei saa täysin sitä mitä on tilattu. Tämän ongelman kääntöpuolena on kuitenkin se hyöty, että pystytään vastaamaan useamman asiakkaan tarpeisiin yhtä aikaa. Muutoksiin reagoinnin koetaan helpottuneen huomattavasti ketterien menetelmien myötä. Myös riskien minimointi on myös ketterän kehityksen tuoman läpinäkyvyyden ansiosta helpottunut. Ongelmana ketterän kehityksen kanssa on kuitenkin, että koko organisaatio ei ole ketterä ja siksi osa ketterien menetelmien arvolupauksista ei toteudu ja näin ollen kaikkea potentiaalista hyötyä ei saada irti.

Yritys 2 kokee ketterien menetelmien arvolupauksien toteutuneen erinomaisesti organisaatiossa. Kehityskaarta on teknologian osalta saatu nopeutettua, ja tekemisen ennustettavuus on parantunut merkittävästi. Tämän lisäksi tekemisen läpinäkyvyys on mahdollistanut nopean palautteen saamista, jolloin muutoksiin on ollut helppo reagoida ja lopputuotteet vastaavat paremmin asiakkaiden tarpeita. Ketterät menetelmät ovat vähentäneet turhan työn tekemistä. Tämän kautta uudelleenkäyttöaste on parantunut paljon ja prosessit ovat tehokkaampia.



Yritys 3 kokee myös ketterien menetelmien arvolupauksien toteutuneen hyvin. Yritys on saavuttanut kehitysmenetelmän muutoksen myötä nopeamman kehityskaaren, sekä riskienhallinta on helpottunut kehittämisen läpinäkyvyyden parannuttua. Merkittävimpinä hyötyinä pidetään kuitenkin sitä, että tuotteet vastaavat paremmin asiakkaiden tarpeita, ja että muutokseen on helpompi reagoida ja tekemisestä on tullut joustavampaa.

## 6.7 Tutkimustulosten luotettavuuden arviointi

Opinnäytetyöhön pyrittiin haastattelemaan henkilöitä joilla on mahdollisimman laaja tietämys ketteristä menetelmistä sekä teoreettisesti että käytännön tasolla. Jotta muutosprosessista ja sen tuoneista hyödyistä ja haitoista, sekä ketterien menetelmien arvolupauksien toteutumisesta saataisiin rehellinen ja laaja kuva, tuli henkilöiden olla perillä organisaation toiminnasta laajemminkin kuin tuotekehitystasolla. Tämä tavoite saavutettiin hyvin, sillä haastateltujen organisaatioiden edustajat olivat hyvin rehellisiä ja avoimia aiheen suhteen. Haastatelluilla henkilöillä oli sekä käytännön kokemusta että teoreettista tietoa ja osaamista ketteristä menetelmistä, ja he olivat kaikki muutoksessa mukana eri rooleissa.

Haastattelutulokset ovat kuitenkin vain organisaation yhden edustajan näkemys aiheesta, ja siihen saattavat vaikuttaa henkilökohtaiset mielipiteet, yksilölliset kokemukset sekä tiedon määrä. Tuloksiin on saattanut tämän lisäksi vaikuttaa haastattelukysymysten muotoilu sekä organisaation sisäisen tiedon salassapitovelvollisuus. Kaikki kolme haastateltua henkilöä ja yritykset joita he edustivat jätettiin työhön anonyymeiksi, jotta aiheesta voitiin keskustella vapaammin, mutta on yleisesti tiedossa, että osa asioista jotka saattaisivat vaikuttaa tutkimustuloksiin, ovat anonyymiteetistä huolimatta yrityksen sisäistä tietoa eikä niitä tuoda esiin.

## 7 Yhteenveto ja johtopäätökset

Kaikissa kolmessa haastattelussa organisaatiossa syyt muutoksen käynnistämiseksi olivat melko yhtenevät: haluttiin parantaa tehokkuutta, lisätä tekemisen joustavuutta ja nopeuttaa kehityskaarta. Organisaatiot olivat ymmärtäneet, että vesiputousmallimainen vaiheittainen kehityskaari ei enää palvele asiakkaiden ja tuotekehityksen tarpeita, joten muutos on välttämätön. Kaikissa yrityksissä muutos käynnistettiin vuosien 2007 ja 2010 välillä. Syitä ajankohdalle ei tullut esiin haastatteluissa, mutta tuohon aikaan ketterät menetelmät olivat jo vakuuttaneet lukuisat ohjelmistokehittäjät hyödyistään, ja yleinen tietämys niistä oli kasvanut paljon. Tämä saattaa olla eräänä taustasyynä juuri näille ajankohdille.

Ketteriin menetelmiin siirtymisen prosessi on voinut kaikissa haastatelluissa organisaatioissa paljon aikaa ja se on vaatinut investointeja, kouluttamista, tiimien uudelleen rakentamista, uusia asenteita, uusien työkalujen luomista, työtilojen muokkaamista, kehitysympäristöjen uudelleensuunnittelua, vastuunsiirtoa ja uusien taitojen opettelua. Haastateltujen yritysten muutosprosessin toteutus on ollut kaikissa yrityksissä erilainen, ja se antaakin hyvän kuvan siitä, miten monella eri tavalla muutosohjelman voi toteuttaa. Ei ole olemassa yhtä oikeaa tapaa siirtyä vesiputousmallista ketteriin menetelmiin - joskin toiset tavat saattavat lopputulokseltaan olla menestyksekkäämpiä kuin toiset. Kaikki riippuu siitä, minkälaiseen organisaatioon muutos ollaan toteuttamassa ja mihin muutoksella pyritään. Yritykset 1 ja 2 toteuttivat muutoksen itsenäisesti, kun taas yritys 3 käytti muutosohjelman konsulttina ketterien menetelmien erikoisasiantuntijaa. Kuitenkin, vaikka yritys 1 ja 2 toteuttivat melko yhteneväällä tavalla muutosohjelmansa, kokee yritys 2 saavuttaneensa muutoksen avulla huomattavasti enemmän kuin yritys 1.

Kaikki yritykset kokevat, että muutos on ollut oikean suuntainen, ja sen avulla on saavutettu haluttuja tuloksia. Yrityksen 1 tapauksessa ketteristä menetelmistä ei kuitenkaan koettu saatavan kaikkea niiden potentiaalista hyötyä irti. Haastattelun perusteella syy ongelmaan on ilmeinen. Kun tarkastellaan esimerkiksi yrityksen 3 muutosta, on siinä useaan kertaan korostettu, että kaiken ja kaikkien tulee muuttua, jotta onnistutaan. Puhutaan niin sanotusta ”isosta kuvasta”. Yrityksessä 1 ilmeinen ongelma on, että ainoastaan ohjelmistotuotanto on siirretty ketterään kehitykseen, mutta muu organisaatio toimii edelleen perinteisten menetelmien mukaisesti. Iteraatioittainen kehitys koetaan hankalaksi, kun esimerkiksi tuotemyynti tapahtuu edelleen vesiputousmallin tukemalla tavalla; koko tuote ja sen ominaisuudet suunnitellaan ja luvataan etukäteen, ja asiakkaiden osallistuminen kehitystyöhön on hyvin vähäistä jollei jopa olematonta. Tästä huolimatta muutos on tuonut mukanaan hyvääkin: muutoksiin reagointi on helpottunut ketteriin menetelmiin siirtymisen myötä paljon, ja tekemisen läpinäkyvyys on helpottanut riskien minimointia.

Yksinä suurimmista hyödyistä, joita ketterät menetelmät ovat tuoneet haastateltuihin yrityksiin, koettiin olevan kehityskaaren nopeutuminen ja turhan tekemisen väheneminen. On yleisesti tiedossa, että monet ohjelmistokehittäjät tekivät todella paljon turhaa työtä ohjelmistoprojekteissa. Dean Leffingwell toteaa kirjassaan *Agile Software Requirements: Lean Requirements Practices for Teams, Programs and the Enterprise* (2014, 6), että tutkimuksen mukaan 31 % vesiputousmallilla toteutetuista ohjelmistoprojekteista peruutetaan ennen kuin ne valmistuvat, ja loppuun viedyissä projekteissa vain 42 % ominaisuuksista vastasi toivottuja. Ohjelmistokehitys on siis kärsinyt turhan työn tekemisen ongelmasta vesiputousmallissa, mutta ketterät menetelmät ovat sekä yleisesti, että haastatellussa yrityksissä osittain ratkaisseet tämän ongelman.

Hyötyjen lisäksi ketterät menetelmät ovat myös haastateltujen yritysten mukaan tuoneet haasteita työskentelyyn. Vaikka tekemisen läpinäkyvyys on parantunut muutoksen myötä, koetaan esimerkiksi työskentelyn lyhytaikaisen ennustettavuuden hankaloituneen ketterien menetelmien vuoksi. Ketterät menetelmät korostavat suunnitelmien muutoksien tekemisen hyötyä, mutta juuri tämä kehityksen eläminen hankaloittaa joidenkin työskentelyä, sillä ei tiedetä, mitä esimerkiksi kahden viikon päästä tehdään. Osa yritysten työntekijöistä kokee tämän vaikeana ja jopa ahdistavana asiana. Ketterien menetelmien ja niihin siirtymisen haittoina nähdään myös työntekijöiden koulutus ja siihen kuluva aika ja investoinnit, menetelmien pakolliset seremoniat, joille ei aina olisi aikaa tai tarvetta ja joidenkin yksilöiden kohdalla lisääntynyt paineen määrä.

Ketterien menetelmien arvolupauksien koetaan toteutuneen hyvin yrityksissä. Vaikka menetelmien mukana on tullut haasteita, koetaan niiden ratkaiseen lupaamansa ongelmat. Haastatellut yritykset ovat hyvin yksimielisiä siitä, että muutoksiin reagointi on helpottunut, tekemisen läpinäkyvyys on parantunut jonka kautta riskien hallinta on helpottunut, kehityskaaret on saatu nopeammiksi ja turhan tekemisen määrä on vähentynyt huomattavasti. Yritys 1 kuitenkin kokee, ettei lopputuote vastaa muutoksen jälkeen paremmin asiakkaan tarpeita - toisin kuin yritykset 2 ja 3 joissa tämän arvolupauksen on koettu toteutuvan erinomaisesti. Syynä yrityksen 1 vastaukseen on yrityksen sisäiset päätökset resurssien kohdentamisesta.

Haitoista ja haasteista huolimatta voidaan haastattelujen perusteella tehdä johtopäätös, että ketteriin menetelmiin siirtyminen on kaikissa kohdeyrityksissä tuonut mukanaan enemmän hyvää kuin huonoa. On selvää, että kehitysmenetelmän muuttuessa niin ihmisten, prosessien kuin työkalujenkin on muututtava. Uudet roolit, taidot, asenteet ja työtahti on sopeutettava työyhteisöön, Gant-kaaviot on vaihdettava työkokonaisuuksien listoiksi, vaiheittainen kehitys on muutettava iteraatioiksi ja työkalujen on mahdollistettava jatkuva integraatio onnistuneen muutoksen takaamiseksi. Vesiputousmallin hylkääminen ja ketteriin menetelmiin siirtyminen

on raskas ja työläs prosessi, mutta yleisesti muutoksen koetaan haastatelluissa yrityksissä olleen oikean suuntainen.

## 8 Toiminnallinen tuotos

Opinnäytetyön toiminnallisena tuotoksena luotiin karkea opas vesiputousmallista ketteriin menetelmiin siirtymistä varten (liite 2). Opas perustuu haastatteluista saatuihin tietoihin, niistä tehtyihin johtopäätöksiin, sekä Dean Leffingwellin teokseen *Agile Software Requirements: Lean requirements practices for teams, programs and the enterprise (2014)*. Oppaan ei ole tarkoitus toimia täydellisenä ohjeena siirtymäprosessin toteuttamiseen, vaan nostaa esiin asioita jotka osoittautuivat oleelliseksi onnistumisen kannalta nimenomaan haastatteluista saadun materiaalin perusteella.

Vesiputousmallista luopuminen ja ketteriin menetelmiin siirtyminen on niin pienelle kuin suurellekin yritykselle tai tiimille ravisuttava muutos. Opinnäytetyön toiminnallisen tuotoksen tarkoituksena on toimia suuntaa antavana ohjenuorana tätä muutosta suunniteltaessa, sitä toteutettaessa ja muutoksen onnistumisen varmistamiseksi. Oppaassa on nostettu esiin asioita mitkä opinnäytetyötä varten haastatellut henkilöt kokivat oleellisimmiksi uuden menetelmän toimivuuden ja muutosohjelman onnistumiseksi, ja pyritty lähestymään aihetta mahdollisimman monelta kannalta.

Opas on pyritty kokoamaan niin, että se olisi sovellettavissa niin suuriin kuin pieniinkin organisaatioihin ja tiimeihin, mutta jotkut kohdat saattavat vaatia omaa tilanne- ja tulkintakykyä tarpeellisuutensa suhteen.

## 9 Opinnäytetyön arviointi

Opinnäytetyön tavoitteena oli pureutua aiheeseen, josta on tällä hetkellä varsin vähän saatavilla tietoa, mutta joka hyödyttäisi mahdollisimman monia osapuolia.

Ohjelmistokehitysalalla tätä opinnäytetyötä on kommentoitu seuraavasti: ” Aiheena oleva muutos on ohjelmistokehitysalalla erittäin kiinnostava koska siihen liittyy paljon arvolatauksia ja lupauksia tuottavuuden paranemisesta. Toisaalta samalla tiedetään, että muutos on suuri ja sen läpi vieminen vaatii aikaa, rahaa ja pitkäaikaista sitoutumista organisaation johdolta. Tuottavuuden ja tuotannon ylläpitäminen muutosprosessin aikana on haasteellista. Käsitellystä aihealueesta on kohtalaisen niukasti saatavilla kattavaa, todellisiin tapauksiin pohjautuvaa kokemusperäistä tietoa, koska ohjelmistokehityksen toimintatavat ovat keskeinen kilpailutekijä alan yrityksille; muutoksen onnistumisien yksityiskohtainen analysointi saattaa auttaa kilpailijoita omissa tehostamispyrkimyksissään. Toisaalta epäonnistumisista ei yritysmaailmassa juurikaan julkisuudessa kerrota.”

Työ onnistuu hyvin summaamaan muutoksen avaintoiminnot sekä nostamaan esiin samaan aikaan sen tuomia hyötyjä ja haasteita. Työn tulos-kappale palvelee haastateltuja yrityksiä tarjoamalla referenssitietoa muiden vastaavien yritysten muutosprosessista ja sen toteutuksesta, ja toiminnallinen tuotos auttaa vielä vesiputousmallia käyttäviä organisaatioita ja tiimejä muutoksen suunnittelussa ja toteutuksessa. Työn käytettävyys ja hyödyllisyys ulottuvat siis myös kohdeyritysten ulkopuolelle.

Opinnäytetyötä on ohjelmistokehitysalalla kommentoitu myös seuraavasti: ”Tätä opinnäytetyötä voisikin suositella lämpimästi lukupaketiksi erityisesti ylemmälle johdolle sellaisissa yrityksissä, jossa harkitaan ohjelmistokehityksen toimintatapojen muutosta. Opinnäyte esittelee sopivan tiiviissä paketissa muutoksen teoreettisella tasolla, konkretisoi asian yritys esimerkkien avulla ja vielä johtopäätöksillä sekä oppailla ja ohjeilla alleviivaa oleellisen.”

Opinnäytetyön toteuttamisen suurimpia haasteita olivat kokonaisuuden rajaaminen sekä uskottavan lähdemateriaalin löytäminen. Kuten aikaisemmin mainittiin, kyseisestä aiheesta on hyvin vähän saatavilla dokumentoituja aineistoja, ja lähes kaikki teoreettinen tieto ketteristä menetelmistä sekä vesiputousmallista on digitaalista, vieraskielistä sekä vaikeasti saatavilla olevaa. Työhön saatiin kuitenkin kerättyä riittävästi teoreettista pohjaa, jotta validiteetti ja reliabiliteetti eivät kärsineet. Työn rajaamisessa linja jouduttiin pitämään hyvin tiukkana, sillä organisaatiomuutokset ja muutosohjelmat käsittävät paljon muutakin kuin mihin tämä työ keskittyy. Työssä onnistuttiin kuitenkin rajaamaan epäoleellinen tieto pois, ja keskittymään ainoastaan oleelliseen.

Kohdeyrityksen 1 edustaja kommentoi valmista työtä seuraavasti: ” Erittäin hyvä, suoraviivainen ja jäsenelty työ jota on helppo lukea ja ymmärtää. Kirjoitustyyli on selkä ja luonteva. Tutkimuksessa on looginen jaottelu teoria osuuteen, tutkimuksen toteuttamiseen ja tuloksiin. Yritys 1:n osalta sekä muutosprosessin käynnistäneet syyt että prosessin tulokset on selkeästi ja helposti ymmärrettävissä ja lisäksi vertailu tutkimuksen muihin yrityksiin on helppoa. Tutkimus vahvistaa yritys 1:n olettamuksia syistä, jotka ovat olleet esteenä muutoksen täydelliselle onnistumiselle.”

Kokonaisuus onnistui aiheen haasteellisuudesta huolimatta, ja työ vastaa hyvin suunniteltua. Se avaa uusia näkökulmia muutoksen suunnitteluun, ja nostaa kokemusten perusteella havaittuja sudenkuoppia esiin, jotta muiden muutosta suunnittelevien organisaatioiden ja tiimien olisi helpompi väistää ne.



## Lähteet

Agile Finland Ry. 2014. <http://agile.fi/> Luettu 10.3.2015.

Agilemanifesto. 2001. Agile Manifesto. <http://agilemanifesto.org/> Luettu 10.3.2015.

Agilemanifesto. 2001. Principles behind the Agile Manifesto. <http://agilemanifesto.org/principles.html> Luettu 11.3.2015.

Agile Methodology. 2014. Agile Methodology. <http://agilemethodology.org/> Luettu 10.3.2015.

Agilesherpa. 2013. A brief history of Agile. [http://www.agilesherpa.org/intro\\_to\\_agile/a\\_brief\\_history\\_of\\_agile/](http://www.agilesherpa.org/intro_to_agile/a_brief_history_of_agile/) Luettu 26.1.2015.

Aspe sdlc Training. 2015. Transition from Waterfall: Becoming Agile. <http://www.aspe-sdlc.com/courses/transitioning-waterfall-agile/> Luettu 26.1.2015.

Codeproject. 2013. Agile software development methodologies and how to apply them. <http://www.codeproject.com/Articles/604417/Agile-software-development-methodologies-and-how-t> Luettu 26.1.2015.

Hirsijärvi, S. & Hurme, H. 2000. Tutkimushaastattelu: Teemahaastattelun teoria ja käytäntö. 4. Painos. Yliopistopaino.

Leffingwell, D. 2014. Agile Software Requirements: Lean requirements practices for teams, programs and the enterprise. 6. Painos. Pearson Education.

Luomala, A. 2008. Muutosjohtamisen ABC. <http://www.uta.fi/jkk/synergos/tyohyvinvointi/oppaat/muutoskirja.pdf> Luettu 23.3.2015.

Reinertsen, D. 2009. The principles of product development flow: Second generation lean product development. 1. Painos. Celeritas Publishing.

Techopedia. 2015. What does software development mean? <http://www.techopedia.com/definition/16431/software-development> Luettu 5.3.2015.

Tekes. 2012. Suomen ohjelmistoteollisuus edelleen kasvussa. <http://www.tekes.fi/nyt/uutiset-2012/suomen-ohjelmistoteollisuus-edelleen-kavussa/> Luettu 26.1.2015.

Tuomi, J. & Sarajärvi, A. 2009. Laadullinen tutkimus ja sisällön analyysi. 5. Painos. Kustannusosakeyhtiö Tammi.

Versionone. 2015. Agile software development benefits. <http://www.versionone.com/agile-101/agile-software-development-benefits/> Luettu 16.3.2015.

Versionone. 2012. 7th annual state of Agile Development survey. <http://www.versionone.com/pdf/7th-Annual-State-of-Agile-Development-Survey.pdf> Luettu 26.1.2015.

Waterfall Model. 2015. Waterfall Model. <http://www.waterfall-model.com/> Luettu 6.3.2015.

Waters, K. 2007. What is Agile? <http://www.allaboutagile.com/what-is-agile-10-key-principles/> Luettu 10.3.2015.

Webopas. 2013. Ohjelmistotuotannon historia. <http://www.webopas.net/otuotanto.html> Luettu 26.1.2015.

## Kuvat, kaavat ja taulukot

Kaava 1: Vesiputousmalli .....	9
Kaava 2: Ketterä kehitysmalli .....	11
Kuva 1: Ketterän ohjelmistokehityksen julistus .....	12
Kuva 2: Ketterien menetelmien sateenvarjokuvio .....	13
Taulukko 1: Vesiputousmallin ja ketterien menetelmien vertailu .....	14

## Liitteet

Opinnäytetyötä varten tehtyjen haastatteluiden kysymykset. ....	36
Ketterien menetelmien käyttöönotto-opas .....	37

Opinnäytetyötä varten tehtyjen haastatteluiden kysymykset.

### Ketteriin menetelmiin siirtyminen

1. Milloin organisaationne aloitti ketteriin menetelmiin siirtymisen prosessin?
2. Miksi prosessi käynnistettiin?
3. Kuinka kauan prosessi kesti?
4. Miten prosessi toteutettiin ja vietiin läpi?
5. Mitä toimenpiteitä organisaatiossa jouduttiin tekemään muutoksen myötä (toimistotilojen uudelleen järjestely, tiimien uudelleen rakennus..)?
6. Mitä vaikutuksia ketteriin menetelmiin siirtymisellä oli työyhteisöön?
7. Mitä hyötyjä ketterät menetelmät ovat tuoneet sekä yksilö- että organisaatiotasolla verrattuna vanhoihin menetelmiin (esim. Vesiputousmalli)?
8. Mitä mahdollisia haittoja ketterät menetelmät ovat tuoneet sekä yksilö- että organisaatiotasolla verrattuna vanhoihin menetelmiin?
9. Miten ketterien menetelmien arvolupaukset ovat toteutuneet organisaatiossanne muutoksen jälkeen (nopeampi kehityskaari, tuote vastaa paremmin asiakkaan tarpeita, muutokseen on helpompi reagoida, riskien minimointi..)?

## Ketterien menetelmien käyttöönotto-opas



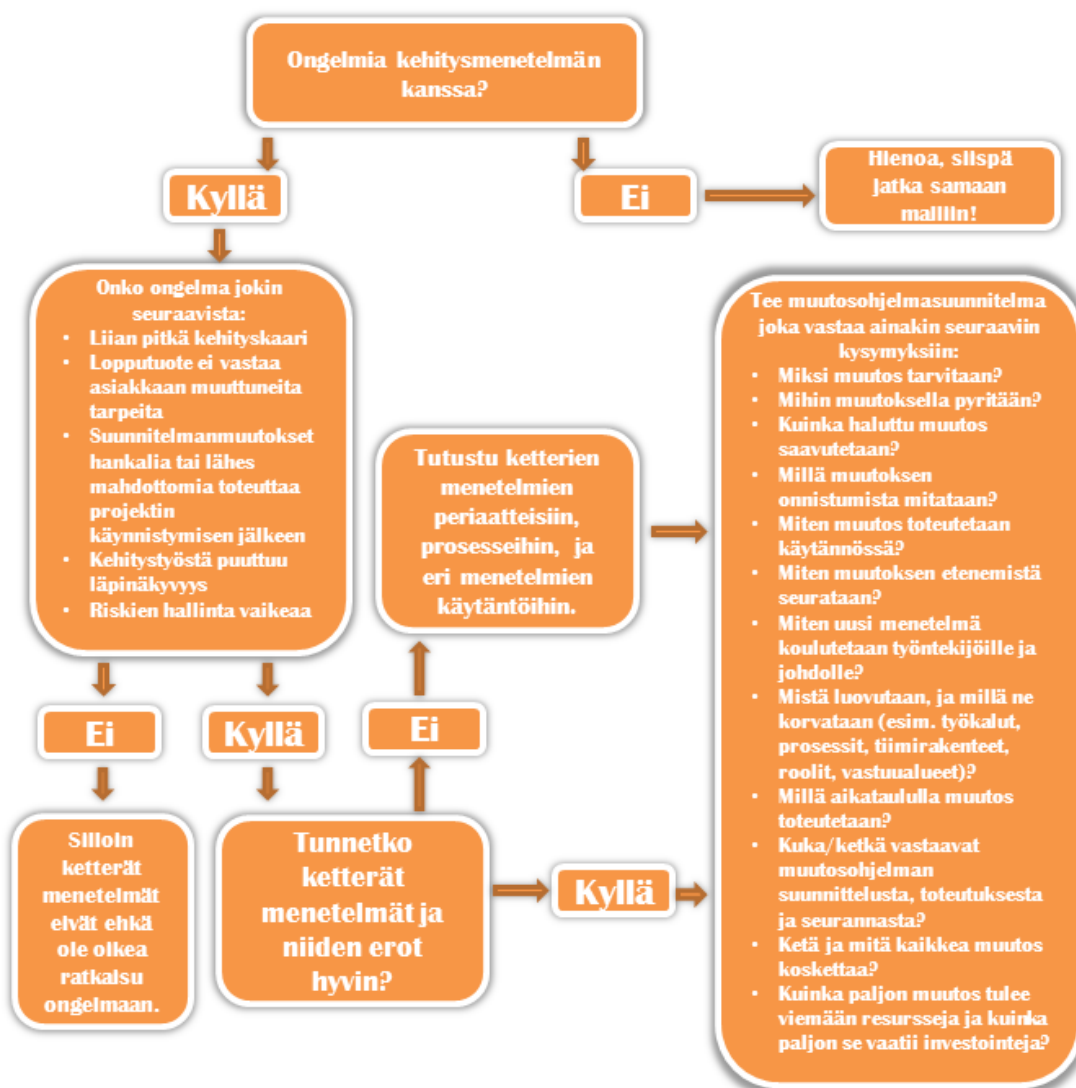
Tämä opas on koottu perustuen materiaaliin, joka on saatu haastattelemalla kolmen kansainvälisen ohjelmistojen kehittäjän suurorganisaation edustajia ketterien menetelmien käyttöönotosta sekä niiden hyödyntämisestä ohjelmistokehityksessä. Lisäksi ohjenuorana ja teoreettisena viitteenä on hyödynnetty Dean Leffingwellin teosta Agile Software Requirements: Lean requirements practices for teams, programs and the enterprise (2014). Oppaan on tarkoitus toimia karkeana ohjeena vesipustousmallista ketteriin menetelmiin siirtymisen prosessiin, sekä nostaa esiin asioita, jotka olisi hyvä ottaa huomioon ohjelmistokehitysmenetelmän muutosta suunniteltaessa, muutosohjelmaa toteuttaessa, ja uuden menetelmän jalkautumisen varmistamiseksi. Ohjeet ovat koottu niin, että ne olisivat mahdollisimman helposti sovellettavissa sekä suuriin että pieniin yrityksiin ja tiimeihin.

## Muutoksen suunnittelu

Ohjelmistokehitysmenetelmän muutoksen suunnittelu lähtee ongelman havaitsemisesta.

Mikäli perinteinen vesiputousmalli-tyyppinen kehitysmalli ei enää palvele parhaalla mahdollisella tavalla kehitystyötä, on pohdittava onko ongelma kehitysmenetelmässä tai sen rakenteessa. Uuteen menetelmään vaihtaminen ei ole ratkaisu kaikkiin mahdollisiin ongelmiin, joten on ensin kartoitettava hyvin ovatko ongelmat, joihin haetaan ratkaisua juuri sellaisia mihin ketterät menetelmät tuovat helpotusta.

Oheinen vuokaavio helpottaa päättämään nopeasti, onko ketteriin menetelmiin siirtymiselle mahdollisesti tarvetta ja miten muutosta kannattaa lähteä suunnittelemaan.



Ketterät menetelmät pyrkivät ratkaisemaan mm. seuraavat ongelmat:

- Liian pitkä kehityskaari
- Lopputuote ei vastaa asiakkaan muuttuneita tarpeita
- Suunnitelmanmuutokset hankalia tai lähes mahdottomia toteuttaa projektin käynnistymisen jälkeen
- Kehitystyöstä puuttuu läpinäkyvyys
- Riskien hallinta vaikeaa

Hyvin toteutettu muutos ei tarkoita hyppäämistä suin päin uuden menetelmän pariin, vaan on tärkeää määritellä tarkkaan muutoksen tavoitteet ja perusteltava tarve muutokselle. Mikäli ongelmat koostuvat edellä mainituista, ketterät menetelmät voivat hyvinkin olla ratkaisu organisaation tai tiimin ongelmiin. Muutostarpeen lisäksi muutosta suunniteltaessa on otettava huomioon kaikki mihin muutos vaikuttaa, ja pohtia miten muutos toteutetaan ja jalkautetaan niin tiimeissä, kuin prosesseissa ja työkaluissakin. Tämän lisäksi hyvin suunnitellussa muutosohjelmassa huomioidaan työolosuhteet, tiedonkulku, kannustaminen, johtaminen, työnsisällöt ja fyysiset olot.

Yksi tärkeimpiä päätöksiä kehitysmenetelmän muutoksessa on, mikä menetelmä otetaan käyttöön. On äärimmäisen tärkeää perehtyä eri ketteriin menetelmiin, siihen minkälaisiin projekteihin mikäkin menetelmä on ideaalisin, ja mikä niistä olisi sopivin omaan organisaatioon. Jotkut menetelmät keskittyvät enemmän kehitysvaiheeseen, kun taas jotkut koko projektin elinkaareen. Jotkut menetelmät ovat sopivia työkalujen kehittämiseen kun taas jotkut laajan asiakasohjelmiston luontiin. Oleellista on tuntea eri menetelmien erot, jotta niistä voidaan valita omaa organisaatiota parhaiten palveleva vaihtoehto. On myös mahdollista, että jokin ketteristä menetelmistä tuntuu muutosvaiheessa sopivalta, mutta myöhemmin huomataan, että se ei olekaan ideaalisin. Silloin on hyvä miettiä kannattaisiko menetelmää vaihtaa.

Menetelmän valinnan lisäksi ohjelmistokehitysiteraation pituus on tärkeää huomioida. Tätä on kuitenkin teoriassa hankala suunnitella täydellisesti etukäteen, sillä ideaalisin iteraation pituus selviää yleensä kokeilemalla. Iteraatio koostuu neljästä perusvaiheesta: suunnittelu, toteutus, arviointi ja retrospektiivi. Näiden vaiheiden kesto on jokaisessa kehitystiimissä hyvin yksilöllistä, ja siksi muutoksen vahvistamisvaiheessa olisi hyvä tarkkailla iteraation pituutta ja pohtia onko se ideaalinen resurssien käytön ja tehokkuuden kannalta.

Riippuen organisaation ja työyhteisön koosta, on myös otettava huomioon, ettei kehitysmenetelmän muutos ole ilmainen. Uuden menetelmän jalkauttaminen ja sekä johdon että työntekijöiden kouluttaminen vie aikaa, joka yritysmaailmassa on yhtä kuin rahaa. On

siis tärkeää tarkistaa myös budjetti, ja varmistaa että muutos voidaan toteuttaa parhaalla mahdollisella tavalla.

Muutosta suunnitellessa on välttämätöntä tehdä kirjallinen suunnitelma johon on summattu kaikki oleellinen muutosprosessiin liittyen. Kun muutosprosessi lopulta käynnistetään, on suunnitelmien oltava valmiita ja selkeitä.



## Muutoksen toteutus

Kun muutos on päätetty käynnistää ja suunnitelmat on lyöty lukkoon, alkaa varsinainen muutosohjelma. Tähän vaiheeseen sisältyy paljon henkilöstön koulutusta, ja on tärkeää, että muutos on perusteltu johdon lisäksi myös työntekijöille, jotta koulutukset ja uudet roolit, prosessit sekä työkalut otetaan vastaan avoimina. Kaikkia yksiköitä ei ole kannattavaa ja järkevää siirtää samanaikaisesti vanhan menetelmän parista uuteen, vaan jokaisen yksikön ja tiimin siirtyminen kannattaa aikatauluttaa erikseen - joskin tämä riippuu organisaation koosta. Mikäli organisaatio on hyvin pieni, voidaan muutos toteuttaa kerralla. Koulutuksen lisäksi työntekijöiden ja johdon olisi hyvä päästä muutoksen aikana halutessaan/tarvittaessa keskustelemaan käyttöön otettavan menetelmän asiantuntijoiden kanssa. Yksilöiden tulisi saada uuden menetelmän koulutuksen lisäksi muutoskoulutusta ja muutostukea. Muutokseen liittyvä kommunikaatio on tärkeää, ja siihen tulee panostaa muutosagenttien lisäksi myös digitaalisesti. Esimerkiksi blogi tai uutiskirje helpottaa johtoa ja muutosvastaavia muutokseen liittyvässä informoinnissa, ja tätä kautta uudesta menetelmästä ja muutosohjelmasta voidaan jakaa tietoa sekä päivityksiä.

Se, miten muutos käytännössä toteutetaan, riippuu täysin organisaation tahdosta. Muutoksessa on mahdollista hyödyntää Dean Leffingwellin SAFe:n kaltaisia muutosohjelmia, tai se voidaan toteuttaa täysin itsenäisesti omalla toteutuksella. Tärkeää on huomioida kaikki oleellinen muutoksen toteuttamisessa, jotta tuloksesta saadaan halutunlainen.

Kehitysmenetelmän muutos tarkoittaa käytännössä sitä, että lähes kaikki mihin ollaan totuttu, tulisi muuttua. Tiimit muotoillaan ja rakennetaan uudelleen, vanhojen roolien tilalle tulee uusia, vanhoja työkaluja korvataan uusilla, työrytmi muuttuu, opeteltavaksi tulee paljon uusia taitoja, ja prosessit muuttuvat vanhoista uutta kehitysmenetelmää tukeviin vaihtoehtoihin. On tärkeää kiinnittää huomiota muutoksessa siihen, että pelkän tuotekehitysyksikön muuttaminen ketteräksi ei palvele tarkoitustaan. Kaiken ja kaikkien on muututtava onnistumisen takaamiseksi.

Muutoksen onnistumisen ja etenemisen seuraamiseksi muutoksesta vastaavien henkilöiden olisi hyvä kokoontua koko prosessin ajan säännöllisin väliajoin keskustelemaan saavutuksista ja tulevista toimenpiteistä. Näin oikeat henkilöt pysyvät ajan tasalla siitä edetäkö muutoksessa oikeaan tahtiin ja oikeaan suuntaan, sekä voidaan suunnitella ratkaisuja, mikäli tulokset eivät vastaa haluttuja. Ideaalisessa tilanteessa muutosohjelman suunnitelmassa on määritelty miten muutoksen onnistumista ja etenemistä mitataan. Mikäli tarvitaan lisämittareita, erilaiset kyselyt auttavat seurannassa ja helpottavat muutosvastaavia paikallistamaan missä tarvitaan ja mitä tarvitaan resurssien kannalta.

Yksi tärkeimpiä muuttujia prosessissa on ihmisten asenteet. Muutos tulee toteuttaa niin, että uusi menetelmä otetaan avoimesti vastaan, ja siitä kiinnostutaan. Onnistuneessa muutosohjelmassa uusi asia nähdään positiivisena, koko työyhteisöä auttavana asiana, ja tämän takaamiseksi muutoksen perustelu ja muutossuunnitelmien pitää olla läpinäkyviä. Kun työntekijät tietävät miksi muutetaan, mihin suuntaan ollaan menossa ja miten muutos hyödyttää sekä organisaatiota että yksilöä, on muutos huomattavasti helpompi vastaanottaa. Tämä informointi ei kuitenkaan koske ainoastaan työyhteisöä, vaan myös asiakkaita. Ketterä kehitys korostaa vahvaa asiakasosallistumista kehitystyöhön, ja tämä merkitsee luonnollisesti sitä, että myös asiakkaan rooli kehitysprosessissa muuttuu. Vesiputousmallin aikaan asiakkaat osallistuivat ohjelmiston suunnitteluvaiheeseen, jolloin määriteltiin ominaisuudet joita tuotteeseen halutaan, mutta tämän jälkeen seuraava asiakaskohtaaminen saattoi olla vasta kun lopputuote oli valmis. Ketterien menetelmien hyödyn takaamiseksi on tärkeää, että vesiputousmallin kaava rikotaan, ja asiakkaat osallistuvat kehitystyöhön koko prosessin ajan antaen palautetta ja määrittelemällä uusia sekä priorisoimalla vanhoja ominaisuuksia. Tämän vuoksi on tärkeää informoida myös asiakkaita kehitysmenetelmän muutoksesta ja organisaation muutosohjelmasta.

## Muutoksen vahvistaminen

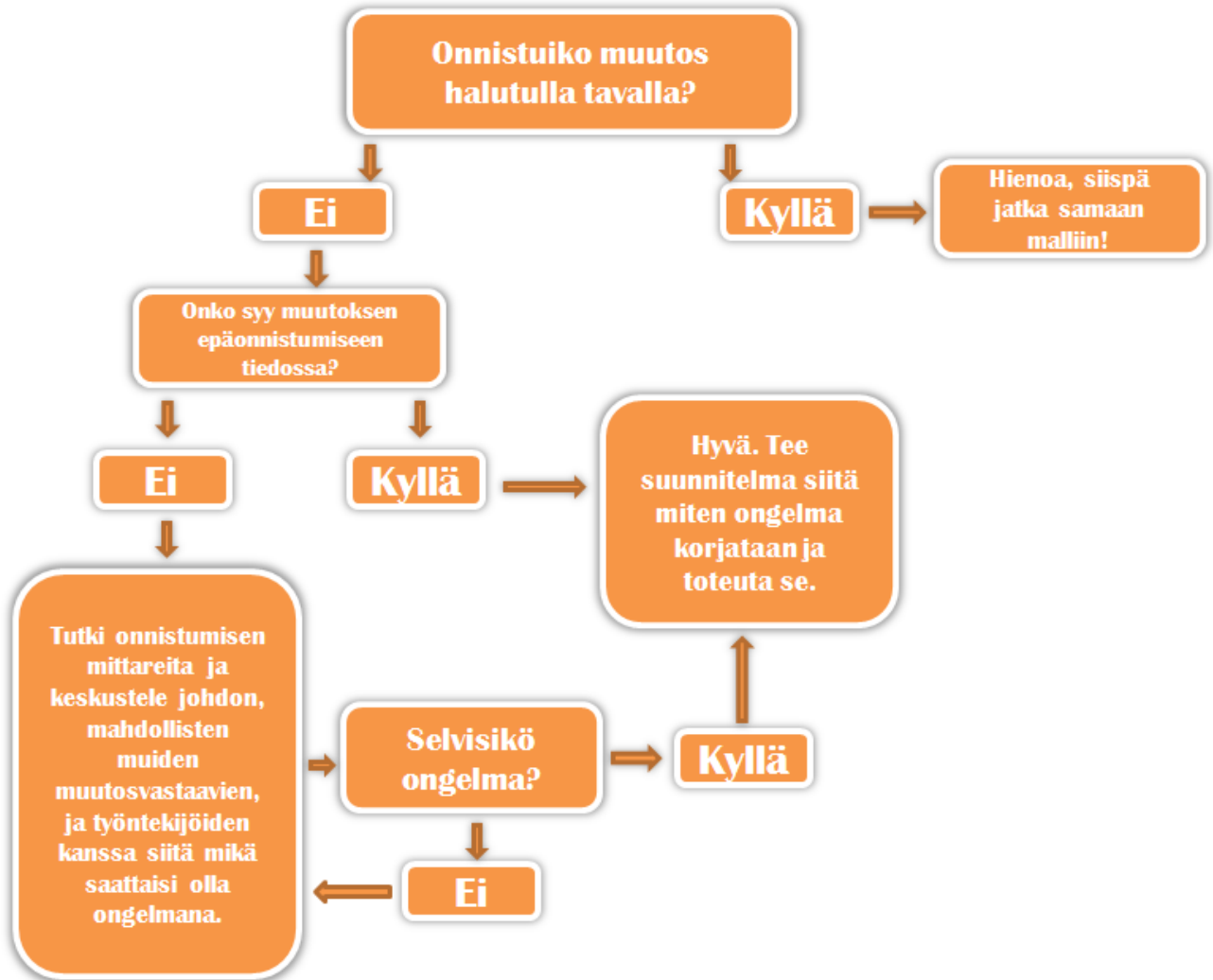
Muutosprosessi ei viittaa ainoastaan siihen ajanjaksoon jolloin konkreettiset muutokset tehdään, vaan siihen kuuluu oleellisena osana myös jalkauttamistoimenpiteet ja niihin kuluva aika. Kun esimerkiksi uusi tiimi ottaa käyttöön uuden menetelmän, lasketaan muutosprosessiin uuden menetelmän opetteluun kuluvan ajan lisäksi myös aika, joka menee uuden taidon lisäksi uuden työtavan ja työrytmin omaksumiseen. Vaikka joku taitaisi menetelmän teoreettisesti hyvin, ei muutos käytännössä ole valmis ennen kuin uusi menetelmä on perin pohjin tuttu ja osa tavallista arkea työpaikalla.

Jotta jalkauttamista ja voidaan seurata, pitää sitä varten olla mittareita. Mittareiden avulla muutosvastaavat voivat seurata kuinka uusi menetelmä on muutoksen jälkeen omaksuttu ja miten se on vaikuttanut työyhteisöön sekä työskentelyyn. Jalkauttamisen lisäksi muutoksen onnistumisesta kertoo melko suoralinjaisesti organisaation tulokset. Ollaanko muutoksella saavutettu haluttuja tuloksia? Ovatko ketterien menetelmien arvolupaukset toteutuneet yrityksen toiminnassa? Mikäli vastaus on kyllä, muutos on hyvällä mallilla. Mikäli vastaus on ei, on syytä pohtia mitkä tekijät ovat johtaneet siihen, ettei haluttuja tuloksia saavuteta, ja mitä voidaan tehdä asian korjaamiseksi. Mittareiden tulkintaa on hyvä tehdä esimerkiksi retrospektiivisissä tapaamisissa, joihin osallistuvat sekä muutosvastaavat ja tiimijohtajia että mahdollisesti yrityksen johtoa. Muutosvastaavat ovat parhaiten perillä siitä missä vaiheessa muutosta ollaan ja kuinka hyvin tavoitteet on saavutettu. Tiimijohtajat pystyvät antamaan parhaimman kuvan työntekijöiden näkökulmasta siihen mitä vaikutuksia muutoksella on ollut työyhteisöön ja työskentelyyn sekä miten uusi menetelmä on otettu vastaan. Yritysjohdo pystyy antamaan taloudellisesta näkökulmasta vankimman näkökulman siihen mitä vaikutuksia muutoksella on ollut.

Muutoksen konkreettisen osuuden jälkeen on myös hyvä pohtia ovatko kaikki käytössä olevat työkalut sellaisia, että ne tukevat uutta kehitystapaa parhaiten. Mikäli joku työkalu on täysin sopimaton uuden menetelmän käytön yhteydessä, tulee siitä varmasti palautetta nopeasti jo muutoksen konkreettisen osuuden aikana. Vahvistamisvaiheessa on hyvä selvittää palvelevatko työkalut parhaimmalla mahdollisimmalla tavalla, vai vain ihan hyvin. Paras palaute tästä saadaan epäilemättä työntekijöiltä; niiltä henkilöiltä jotka ko. työkalua käyttävät eniten. Tämän lisäksi vahvistamisvaiheessa olisi hyvä tarkistaa jo aikaisemmin mainittu iteraation pituus. Asia on tärkeä siksi, että uuden menetelmän jalkauttamiseksi olosuhteet uudelle työskentelylle tulisivat olla mahdollisimman hyvät.

Mitä suurempaa työyhteisöä/organisaatiota muutos koskettaa, sitä monimutkaisempaa sen toteuttaminen ja jalkauttaminen on. Aina muutos ei tuota haluttua lopputulosta, ja silloin on

hyvä pysähtyä pohtimaan missä on menty vikaan. Oheinen vuokaavio helpottaa menettelyä muutoksen onnistumisen arvioinnissa tai epäonnistumisen määrittelyssä.



Muutos vesiputousmallista ketteriin menetelmiin on raskas ja monimutkainen prosessi riippumatta organisaation koosta. Mikäli sillä haetaan ratkaisua oikeanlaisiin ongelmiin ja se toteutetaan oikein, se auttaa niin yksilöä kuin organisaatiotakin saavuttamaan parempia tuloksia ja saa aikaan tyytyväisempiä asiakkaita.