



Aalto-yliopisto  
Perustieteiden  
korkeakoulu

Alex Mehdi Farsimadan

---

# Vaatimusmäärittelykäytännöt hajaute- tussa ja ketterässä ohjelmistokehityksessä

Diplomityö, joka on jätetty opinnäytteenä  
tarkastettavaksi diplomi-insinöörin tutkintoa varten.

Espoossa 20.04.2021  
Valvoja: Professori Risto Sarvas  
Ohjaaja: Pekka Korpi-Tassi

---

**Tekijä** Alex Mehdi Farsimadan

---

**Työn nimi** Vaatimusmäärittelykäytännöt hajautetussa ja ketterässä ohjelmistokehityksessä

---

**Maisteriohjelma** Informaatioverkostot

**Koodi** SCI3047

---

**Työn valvoja** Prof. Risto Sarvas

---

**Työn ohjaaja(t)** MSc Pekka Korpi-Tassi

---

**Päivämäärä** 20.04.2021

**Sivumäärä** 91

**Kieli** Suomi

---

### Tiivistelmä

Ohjelmistokehitysprojektien onnistuminen riippuu siitä, kuinka hyvin kehitetyn ohjelmiston toiminnallisuudet ja ominaisuudet vastaavat loppukäyttäjien tarpeita. Näin ollen määrämuotoinen ja laadukas vaatimusmäärittelyprosessi toimii onnistuneen ohjelmistoprojektin kulmakivenä. Tässä diplomityössä tutkitaan, millaisia haasteita ohjelmistoliiketoiminta alalla toimivat organisaatiot kohtaavat vaatimusmäärittelyssä tutkimalla alan kirjallisuutta. Aihetta lähestytään suorittamalla toimintatutkimus valitulle kohdeyritykselle, joka on erikoistunut tuotteistettuun ohjelmistoliiketoimintaan.

Tämän diplomityön tavoitteena on ensinnäkin luoda yhtä yritystä ja toimialaa laajempi ymmärrys ohjelmistoliiketoiminta-alalla käytetyistä vaatimusmäärittelykäytännöistä, sekä niihin liittyvistä haasteista. Toimintatutkimuksen tavoitteena on selvittää, minkälaisia haasteita kohdeyrityksen nykyisessä vaatimusmäärittelyprosessissa esiintyy vaatimusmäärittelijöiden näkökulmasta, sekä tunnistaa nykyisestä prosessista kipukohtia, joihin pyritään löytämään määrämuotoisia ratkaisukeinoja. Tutkimuksen pääasiallisena tiedonkeruumenetelmänä käytettiin puolistrukturoituja haastatteluja, joiden avulla nykyiset haasteet saatiin esille.

Tutkimustulosten perusteella alan kirjallisuudessa esiintyvät vaatimusmäärittelyhaasteet esiintyvät myös kohdeyrityksessä. Kohdeyrityksellä on haasteita soveltaa ketteriä menetelmiä hajautetussa ohjelmistokehitystiimissä. Ketterien menetelmien mukaisesti kohdeyrityksessä dokumentoidaan vaatimuksia minimaalisesti, joka on osoittautunut haasteelliseksi. Lisäksi hankkeiden aikataulu ja budjettirajoitteiden vuoksi kohdeyrityksen toimitusprojekteissa vaaditaan tietynlaista jäykkyyttä, jonka vuoksi kohdeyrityksessä tarvitaan määrämuotoisia ja yhtenäisiä menetelmiä asiakkaiden todellisten tarpeiden ymmärtämiseen.

Tulokset osoittavat, että ketterien vaatimusmäärittelymenetelmien soveltaminen itsessään ei riitä kohdeyrityksessä, sillä hajautetun luonteen vuoksi korkealaatuista ja intensiivistä viestintää ei voida toteuttaa. Vuorovaikutukseen perustuvaan tiedonjakoon ei voida nojautua. Sen sijaan dokumentaation määrää täytyy tasapainottaa niin, että se vastaa kohdeyrityksen tarpeita.

Tämä tutkimus tukee alan kirjallisuudessa havaittuja haasteita peilaamalla tuloksia olemassa oleviin tutkimuksiin. Lisäksi tämä tutkimus tarjoaa käytännönehdotuksia vaatimusmäärittelyprosessin tehostamiseen sekä kohdeyritykselle että muille vastaaville organisaatioille.

---

**Avainsanat** Vaatimusmäärittelymenetelmät, toimintatutkimus, hajautettu ketterä ohjelmistokehitys

---

---

**Author** Alex Mehdi Farsimadan

---

**Title of thesis** Requirements engineering practices in distributed and agile software development

---

**Master programme** Information Networks

**Code** SCI3047

---

**Thesis supervisor** Prof. Risto Sarvas

---

**Thesis advisor(s)** MSc Pekka Korpi-Tassi

---

**Date** 20.04.2021

**Number of pages** 91

**Language** Finnish

---

### Abstract

The success of software development projects depends on how well the functionality and features of the developed software meet the needs of the end users. Thus, a high-quality requirement engineering process serves as the cornerstone of a successful software project. This research examines the challenges that organizations operating in the software business field face in the context requirements engineering. The topic is approached by conducting an action research for a selected target company that specializes in software product business.

The aim of this research is, firstly, to create a broader understanding of the requirements engineering practices used in the software business, as well as the related challenges, by one company and one industry. The aim of the action research is to find out what kind of challenges exist in the target company's current requirements engineering process from the point of view of requirements specialists, and to identify the pitfalls in the current process for which specific solutions can be introduced. The main data collection method of the study was semi-structured interviews, which were used to highlight the current challenges.

Based on the research results, the requirements engineering challenges that appear in the literature in the field also occur in the target company. The target company faces challenges in applying agile methods in a distributed software development team. In accordance with agile methods, the requirements are documented minimally in the target company, which has proven to be challenging. In addition, due to project timelines and budgetary constraints, a certain degree of rigidity is required in the delivery projects of the target company, which requires specific and uniform methods in the target company to understand the real needs of the customers.

The results indicate that applying agile requirements engineering practices are not sufficient, as high quality and intensive communication cannot be implemented due to its distributed nature. Thus, Interaction-based information sharing cannot be relied upon. Instead, the amount of documentation must be balanced to meet the needs of the target company.

This study supports the challenges identified in the literature by mirroring the results to existing studies. In addition, this study provides practical suggestions for streamlining the requirements engineering process for both the target company and other similar organizations.

---

**Keywords** Requirements engineering, distributed agile development, action research

---

## ESIPUHE

Pakko kyllä sanoa, että oli kyllä aikamoinen työ saada maisteritutkintoa suoritettua täysipäiväisen konsultointityön ja yrittäjyyden ohella. Insinööriopintojen alussa tiesin, että haluan jatkaa opintoja ja valmistua Aalto Yliopistosta diplomi-insinööriksi. Noh, nyt se on tehty. VIHDOIN! Itse työn aihe muodostui omien havaintojen perusteella. Olin jo työskennellyt tuossa yrityksessä melkein kolme vuotta, kun aloin diplomityötä kirjoittaa. Keskustelin diplomityön aiheesta esimiehieni kanssa ja sain hyväksynnän työn aiheelle. Sain myös tukea ja aikaa diplomityön kirjoittamiselle, kiitokset tästä erityisesti Ollille.

Isoimmat kiitokset kuitenkin kuuluvat Risto Sarvakselle. En olisi millään työtä tehtyä näin menestykkäästi ilman niitä monia sparrailusessioita ja neuvoja. Kiitokset myös Marjo Kauppiselle. Sytytit minussa inspiraation vaatimusmäärittelyjä kohtaan ja tietoisuus aiheen tärkeydestä kasvoi entistä vahvemmaksi. Kiitokset myös koko perheelle ja erityisesti äidille Soonialle tuesta ja motivaatiosta.

Nyt juhlitaan. Over and out.

Espoossa 20.04.2021

Alex Mehdi Farsimadan

## Sisällysluettelo

Sisällysluettelo .....	1
1 Johdanto .....	3
1.1 Tutkimuksen tausta ja motivaatio .....	3
1.2 Tutkimuksen tavoitteet ja tutkimuskysymykset .....	5
1.3 Tutkielman rakenne .....	7
1.4 Tutkimuksen laajuus .....	8
2 Tausta ja kirjallisuus .....	9
2.1 Ketterä ohjelmistokehitys .....	9
2.1.1 Ketterän ohjelmistokehityksen periaatteet.....	11
2.1.2 SCRUM .....	14
2.2 Hajautettu ketterä ohjelmistokehitys.....	17
2.2.1 Hajautetun ketterän ohjelmistokehityksen hyödyt ja haasteet.....	18
2.3 Vaatimusmäärittelyt .....	20
2.3.1 Vaatimusmäärittelytyypit .....	21
2.4 Perinteinen vaatimusmäärittelyprosessi .....	22
2.5 Ketterä vaatimusmäärittelyprosessi .....	28
2.5.1 Ketterät vaatimusmäärittelymenetelmät ja niihin liittyvät haasteet.....	28
2.6 Ketterien ja perinteisten vaatimusmäärittelymenetelmien vertailu.....	31
2.7 Yhteenveto .....	33
3 Metodologia.....	37
3.1 Tutkimuksen lähestymistapa .....	38
3.2 Case-yrittäjä .....	40
3.2.1 Kohdeyrityksen kuvaus ja tuotteistettu ohjelmisto .....	40
3.2.2 Henkilöstön rakenne.....	42
3.3 Tiedonkeruumenetelmät .....	43
3.3.1 Diagnosointivaiheen tiedonkeruumenetelmät .....	44
3.3.2 Toimintavaiheen tiedonkeruumenetelmät .....	47
3.4 Analysointimenetelmät.....	48
3.4.1 Diagnosointivaiheen tiedon analysointi .....	49
3.4.2 Toiminnan tiedon analysointi .....	51
3.5 Tutkimusetiikka .....	51
4 Tulokset.....	52
4.1 Diagnosointivaiheen tulokset .....	53
4.1.1 Nykyinen vaatimusmäärittelyprosessi.....	53
4.1.2 Nykyiset haasteet (TK1) .....	57
4.2 Toiminnan toteutuksen tulokset.....	65
4.2.1 Iteratiivisen työpajan käytännöt ja menetelmät .....	65
4.2.2 Haastatteluissa esiin tulleiden haasteiden esittäminen ja niistä keskusteleminen .....	66
4.2.3 Vaihtoehtoisten ratkaisujen havainnointi ja kipukohtien löytäminen (TK2). 71	
5 Johtopäätökset .....	74
5.1 TK1: Haasteet nykyisessä vaatimusmäärittelyprosessissa .....	74

5.2	TK2: Sovellettavat vaatimusmäärittelykäytännöt ja kipukohtien havainnointi ...	77
5.3	Tutkimuksen vaikutus kohdeyrityksessä ja toimenpidesuositukset.....	79
5.4	Tutkimuksen kontribuutio .....	83
5.5	Tutkimuksen rajoitteet .....	84
6	Lähteet.....	87

## 1 Johdanto

Tässä diplomityössä tutkitaan alan kirjallisuuden sekä case-tutkimuksen kautta millaisia vaatimusmäärittelyihin liittyviä haasteita organisaatiot kohtaavat. Diplomityön tarkoituksena on löytää case-yrityksen vaatimusmäärittelyprosessin haasteet työntekijöiden näkökulmasta sekä löytää kipukohtia, joihin voidaan soveltaa vaihtoehtoisia menetelmiä.

Tässä ensimmäisessä kappaleessa käydään läpi tutkimuksen taustaa, tavoitteita ja motivaatiota tutkimuksen suorittamiselle. Näiden lisäksi tässä kappaleessa esitellään tutkimuksen rakenne, laajuus ja mahdolliset rajoitteet.

### 1.1 Tutkimuksen tausta ja motivaatio

Ohjelmistokehitysprojektien onnistuminen riippuu siitä, kuinka hyvin kehitetyn ohjelmiston toiminnallisuudet ja ominaisuudet vastaavat loppukäyttäjän tarpeita. Vaatimusmäärittelyillä viitataan vaiheeseen, jossa asiakkaiden sidosryhmien tavoitteet, tarpeet ja järjestelmävaatimukset kirjataan ennalta määrättyyn muotoon. Yksinkertaistettuna vaatimusmäärittelyllä varmistetaan, että toimitettu ratkaisu vastaa todellisesti asiakkaiden odotuksia. Sillitti & Succi (2005) mukaan määrämuotoinen ja laadukas vaatimusmäärittelyprosessi toimii onnistuneen ohjelmistoprojektin kulmakivenä. Sen sijaan kehnosti suoritettu vaatimusmäärittely on yksi isoimmista syistä miksi monet ohjelmistoprojektit epäonnistuvat. Epäonnistuneet järjestelmähankkeet ovat harmillisen yleisiä ohjelmistoliiketoiminnan alalla. IAG:n tuottaman tutkimuksen mukaan kehnosti suoritettu vaatimusmäärittely ja hallinta kasvattaa projektin kokonaisbudjettia ja aikaa yli kuudellakymmenellä prosentilla (IAG, 2008). Tämän vuoksi on ehdottoman tärkeää tutkia yritysten kohtaamia vaatimusmäärittelyhaasteita sekä etsiä mahdollisia ratkaisukeinoja, mikäli onnistuneiden ohjelmistoprojektien osuutta halutaan kasvattaa. Näiden toimenpiteiden avulla yrityksillä on parempi valmius ymmärtää mistä kyseiset haasteet johtuvat, sekä mahdollisesti havaita ja ottaa käyttöön uusia, hyväksi todettuja menetelmiä.

Vaatimusmäärittelyihin liittyviä haasteita on tutkittu niin kauan, kun ohjelmistoja on ylipäättänsä kehitetty. Erilaisia tutkimuksia sekä kirjallisuutta vaatimusmäärittelyistä ja niihin liittyvistä haasteista löytyy runsaasti. Noraini & Abdullah (2015) mukaan vaatimusmäärittelyihin liittyvät haasteet ovat kuitenkin rakenteeltaan erittäin monimutkaisia, sillä vaatimusmäärittelyprosessi usein vaatii laajaa koordinoitua sekä asiakkaiden että toimittajien sidosryhmien kanssa. Tämä myös tarkoittaa, että viestinnällä on suuri merkitys vaatimusmäärittelyprosessissa. Davis et.al (2006) mukaan vaatimusten kartoittaminen (elicitation) on vaatimusmäärittelyprosessin tärkein vaihe. Vaatimusten kartoittamisella pyritään ymmärtämään eri sidosryhmien tarpeet, jolloin onnistuneen viestinnän merkitys kasvaa tässä vaiheessa.

Sopivien vaatimusmäärittelymenetelmien valinta yrityksessä on osoittanut hankalaksi. Jossain yrityksessä saattaa toimia menetelmät, jotka taas eivät toisessa yrityksessä toimi ollenkaan. Yousuf & Asger (2015) mukaan ei ole olemassa täydellistä vaatimusmäärittelysapluunaa, joka toimisi kaikissa yrityksissä ja kaikissa tilanteissa. Menetelmien toimivuuteen vaikuttaa muun muassa organisaation rakenne, liiketoiminta-ala, toimittajan ja asiakkaiden välinen vuorovaikutussuhde tai esimerkiksi käytetyt viestintämenetelmät. Erilaiset projektinhallinta- ja vaatimusmäärittelymenetelmät vielä hankaloittavat asiaa entisestään. Sommerville (2004) mukaan perinteisen, eli vesiputousmallisen vaatimusmäärittelyprosessin periaatteena on luoda raskas dokumentaatio, jossa kaikki spesifikaatiot, toiminnalliset ja ei-toiminnallisuudet on kuvattu tarkalla tasolla ennen varsinaista kehitystyön alkua. Vesiputousmallisessa vaatimusmäärittelyprosessissa syntyy haasteita varsinkin niillä liiketoiminta-alueilla, joissa sidosryhmien tarpeet saattavat muuttua tai jopa vanhentua projektin aikana. Näiden nopeiden muuttuvien tarpeiden takia vesiputousmallista vaatimusmäärittelyä prosessia ei voida sellaisenaan käyttää tietyillä liiketoiminta-aloilla, jossa kaikki vaatimukset on kuvattava ennen kehityksen alkua. (Cao & Ramesh, 2009). Jokaisella menetelmällä on omat vahvuutensa ja haasteensa. Tietyissä ympäristöissä vaaditaan jäykkyyttä, jolloin vesiputousmalliset projektinhallinta- sekä vaatimusmäärittelymenetelmiä on suotuisaa soveltaa. Sen sijaan tietyissä ympäristöissä vaaditaan enemmän joustavuutta, jolloin sitten lukuisia ketteriä menetelmiä voidaan soveltaa. Näiden edellä mainittujen tekijöiden



vuoksi on ehdottoman tärkeää yrittää selvittää millaisia haasteita yritykset kohtaavat vaatimusmäärittelyissä sekä löytää mahdollisesti vaihtoehtoisia ratkaisukeinoja näihin haasteisiin.

Tämä tutkielma suoritetaan kohdeyritykselle, joka on erikoistunut tuotteistettuun ohjelmistoliiketoimintaan. Kohdeyritys on kehittänyt modernin toiminnanohjausjärjestelmän, jota asiakkaat käyttävät materiaalien, kunnossapidon, työntekijöiden ja palveluiden hallintaan. Kohdeyrityksen toiveiden mukaan yrityksen nimeä ei julkaista tässä diplomityössä. Kohdeyrityksen asiakkaat toimivat pääsääntöisesti kunnossapidon tai elintarvike- ja energiateollisuuden liiketoiminta-alalla. Tämän lisäksi kohdeyrityksen asiakkaita ovat eri aloilla toimivat yritykset, joilla on erilaisia tuotantolaitoksia. Asiakasyritysten heterogeenisten luonteen vuoksi kohdeyrityksellä on ollut haasteita vaatimusten hallinnassa. Tämän lisäksi vaatimusmäärittelyistä vastaavilla henkilöillä on toisistaan poikkeavat käytännöt vaatimusten hallintaan ja määrittelyihin liittyvissä tehtävissä. Kohdeyrityksessä toivotaan vaatimusmäärittelyihin vankempaa otetta ja määrämuotoisuutta, sillä nykyisessä prosessissa on haasteita. Tämän diplomityön laatija on työskennellyt kohdeyrityksessä yli kolme vuotta ja ollut lukuisissa asiakasprojekteissa mukana erilaisissa rooleissa. Tästä syystä diplomityön aihe syntyi tutkijan havaintojen perusteella. Havaintojen perusteella käytiin iteratiivista keskustelua tutkijan ja ylemmän johdon välillä, josta sitten yhteispäätöksellä sovittiin tarkka aihe diplomityölle.

## **1.2 Tutkimuksen tavoitteet ja tutkimuskysymykset**

Tämän diplomityön tavoitteena on ensinnäkin luoda yhtä yritystä ja toimialaa laajempi ymmärrys ohjelmistoliiketoiminta-alalla käytetyistä vaatimusmäärittelykäytännöistä selvittämällä alalla sovellettuja käytäntöjä, sekä selvittää, minkälaisia haasteita organisaatiot kohtaavat vaatimusmäärittelyihin liittyen. Aihetta lähestytään suorittamalla toimintatutkimus valitulle kohdeyritykselle, joka on erikoistunut tuotteistettuun ohjelmistoliiketoimintaan. Toimintatutkimuksen tavoitteena on selvittää, minkälaisia haasteita kohdeyrityksen nykyisessä vaatimusmäärittelyprosessissa esiintyy haastateltavien näkökulmasta, sekä löytää jo-

kin tietty kipukohta iteratiivisen prosessin kautta, johon pyritään hakemaan ratkaisua. Tutkimuksen tavoitteisiin päästään vastaamalla tutkimuksessa seuraaviin tutkimuskysymyksiin:

**TK1:** Minkälaisia haasteita esiintyy kohdeyrityksen nykyisessä vaatimusmäärittelyprosesseissa vaatimusmäärittelijöiden näkökulmasta?

**TK2:** Millaisilla käytännöillä voitaisiin parantaa kohdeyrityksen tuoteliiketoiminnan hajautettujen ohjelmistokehityshankkeiden vaatimusmäärittelyprosessia?

Ensimmäisen tutkimuskysymyksen vastaamiseen tässä tutkielmassa hyödynnetään laadullisia menetelmiä keräämällä empiiristä aineistoa puolistrukturoitujen haastattelujen avulla. Haastateltaviksi valitaan ne työntekijät, jotka vastaavat tosiasiallisesti kohdeyrityksen vaatimusmäärittelyprosessin hallinnasta. Haastatteluista kerätyn aineiston avulla luodaan syvälinen ymmärrys kohdeyrityksen nykyisen vaatimusmäärittelyprosessin haasteista haastateltavien näkökulmasta. Toisen tutkimuskysymyksen vastaamiseen hyödynnetään sekä haastatteluista kerättyä empiiristä aineistoa sekä luvussa kaksi havainnoitua erilaisia vaatimusmäärittelymenetelmiä. Luvussa kaksi havainnoidaan ohjelmistoliiketoiminta-alalla käytettyjä projektinhallinta- ja vaatimusmäärittelymenetelmiä sekä niihin liittyviä haasteita, jotka pohjautuvat olemassa oleviin tutkimuksiin. Tämän diplomityön tuloksia hyödyntäen kohdeyrityksellä on parempi ymmärrys haasteista, jotka liittyvät heidän tuoteliiketoimintansa vaatimusmäärittelyprosessiin. Näiden haasteiden tiedostamisen avulla kohdeyrityksessä pystytään luomaan yhtenäisiä käytäntöjä, joiden avulla voidaan kasvattaa toimitusprojektin laatua. Näin ollen tutkielma on hyvin käytännönläheinen, sillä tutkimuksessa haastatellaan henkilöitä, jotka ovat tosiasiallisesti tehneet vaatimusmäärittelyjä ohjelmistoliiketoiminnan alalla vuosia. Tutkimuksen viimeisessä osiossa myös havainnoidaan tutkimuksen yleistettävyyttä peilaamalla tutkimuksen empiirisen osan tuloksia olemassa oleviin tutkimuksiin.

### 1.3 Tutkielman rakenne

Tässä osiossa esitetään kokonaisuudet, joista tämä tutkielma muodostuu. Tutkielma on jaettu viiteen eri kokonaisuuteen, joita havainnoidaan alla olevassa listassa:

#### **Kappale 1**

Ensimmäisessä kappaleessa havainnoidaan tutkimuksen taustaa sekä motivaatiota tutkimuksen tekemiselle. Tämän lisäksi tässä kappaleessa esitellään tutkimuksen tavoitteet, tutkimuskysymykset, tutkielman rakenne sekä tutkimuksen laajuus ja rajoitteet. Tämän kappaleen tarkoitus on antaa lukijalle kattava käsitys tutkielmasta kokonaisuudessaan.

#### **Kappale 2**

Toisessa kappaleessa havainnoidaan olemassa olevia tutkimuksia ja kirjallisuutta, jotka ovat olennaisia tälle diplomityölle. Tämän kappaleen alussa esitellään yleisesti sekä ketterän että hajautetun ketterän ohjelmistokehitykseen liittyviä kokonaisuuksia, jonka jälkeen havainnoidaan erilaisia vaatimusmäärittelymenetelmiä ja verrataan näitä menetelmiä toisiinsa.

#### **Kappale 3**

Kolmannessa kappaleessa, eli tutkimuksen empiirisessä osassa havainnoidaan tarkemmalla tasolla mitä metodologiaa ja metodeja tutkimustyössä käytetään, sekä argumentoidaan metodologiavalintojen puolesta. Tutkimuksen empiirisessä osassa selvitetään kohdeyrityksen nykytila keräämällä laadullista dataa puolistrukturoitujen haastatteluiden ja havaintojen avulla. Tämän jälkeen toteutetaan yhteinen työpaja haastateltavien kesken, jonka tarkoitus on keskustella yhdessä löydetyistä haasteista ja mahdollisista ratkaisukeinoista.

#### **Kappale 4**

Neljännessä kappaleessa esitellään tutkimuksen empiirisen osakokonaisuuden tulokset. Kappaleessa 4.1 havainnoidaan ensimmäisen tutkimuskysymyksen tuloksia, jotka pohjautuivat kohdeyrityksen nykyisten haasteiden selvittämiseen puolistrukturoitujen haastatteluiden avulla. Kappaleessa 4.2 sen sijaan havainnoidaan toisen tutkimuskysymyksen tuloksia. Kappaleen 4.2 tulokset muodostuivat toiminnan toteutuksen aikana.

## **Kappale 5**

Viidennessä kappaleessa havainnoidaan tutkimuksen johtopäätöksiä ja vaikutuksia kohdeyrityksessä. Tämän lisäksi viimeisessä kappaleessa havainnoidaan tutkimuksen rajoitteita.

### **1.4 Tutkimuksen laajuus**

Tässä diplomityössä keskitytään vaatimusmäärittelyihin liiketoiminnan ja projektin johdon näkökannalta. Toisin sanoen, tässä diplomityössä keskitytään löytämään yksittäisiä ja yhteisiä haasteita haastateltavien näkökulmasta, jotka liittyvät toimitusprojektin vaatimusmäärittelyprosessiin. Tässä diplomityössä ei oteta huomioon haasteita vaatimusmäärittelyissä, jotka liittyvät ohjelmistokehityksen teknisiin haasteisiin. Sen sijaan tässä diplomityössä keskitytään ensisijaisesti löytämään haasteita kohdeyrityksen nykyisestä vaatimusmäärittelyprosessista, sekä löytämään jokin tietty kipukohta, johon haetaan ratkaisua yhteisvoimin. Näin ollen tutkimuksen tavoitteena on luoda interventioiden kautta määräämuotoisia menetelmiä, joiden avulla ohjelmistoprojektien toimittamisen laatua voidaan mahdollisesti parantaa.

## 2 Tausta ja kirjallisuus

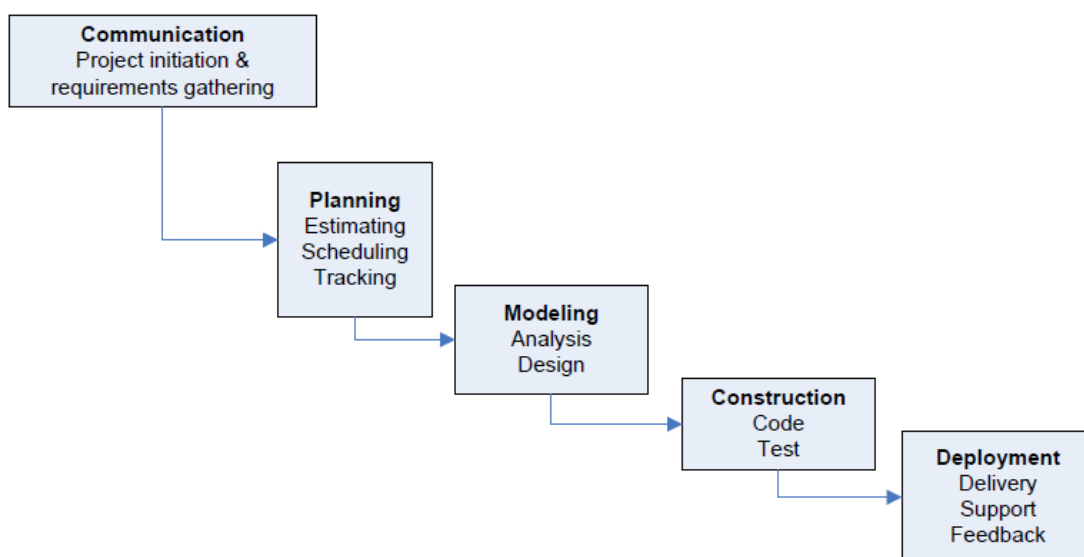
Tässä kappaleessa havainnoidaan tutkimuksen kannalta olennaisia aikaisempia tutkimuksia. Tässä luvussa esitetään tutkimusta tukevia sekä kirjallisuudessa esitettyjä vaatimusmäärittelykäytäntöjä ja käytäntöihin liittyviä prosesseja sekä haasteita. Näin ollen tutkielman taustoituksessa ja kirjallisuudessa keskitytään tunnistamaan menetelmiä, joita organisaatiot käyttävät vaatimusten läpiviennissä ja hallinnassa.

Jotta voidaan ymmärtää vaatimusmäärittelyihin liittyviä haasteita ja menetelmiä, on ensinnäkin ymmärrettävä menetelmien käyttöön sovellettavia projektinhallintamenetelmiä sekä niiden hyötyjä ja haasteita. Organisaatioiden jatkuvien muuttuvien liiketoimintatarpeiden vuoksi monet yritykset suosivat ketteriä menetelmiä perinteisten sijaan, jonka vuoksi luvussa 2.1 havainnoidaan ketterien ohjelmistokehityksen periaatteita. Hajautettujen ohjelmistokehitystiimien määrä on viime vuosina ollut kovassa kasvussa organisaatioiden tavoittelemien kustannussäästöjen vuoksi. Monet organisaatio kuitenkin pyrkivät käyttämään ketteriä menetelmiä myös hajautetuissa tiimeissä. Ketterien menetelmien soveltamisessa hajautetuissa tiimeissä syntyy myös haasteita. Hajautettujen ketterien menetelmiin liittyvät hyödyt ja haasteet avataan tarkemmalla tasolla kappaleessa 2.2. Vaatimusmäärittelyt voidaan karkeasti jaotella perinteisiin ja ketteriin vaatimusmäärittelyihin. Luvuissa 2.3–2.5 havainnoidaan ketterien ja perinteisten vaatimusmäärittelymenetelmien vaiheita ja niihin liittyviä, yleisesti tunnistettuja haasteita. Tämän luvun viimeisessä osiossa vertaillaan ketteriä ja perinteisiä vaatimusmäärittelymenetelmiä toisiinsa ja esitellään menetelmien hyvät puolet sekä mahdolliset haasteet.

### 2.1 Ketterä ohjelmistokehitys

Vesiputousmallinen ohjelmistokehitys perustuu lineaariseen prosessiin, jossa kaikki sidosryhmien toiminnalliset ja ei-toiminnalliset vaatimukset määritellään varhaisessa vaiheessa ennen varsinaista ohjelmistokehityksen aloittamista. Vesiputousmallisessa ohjelmistokehi-

tyksen lähestymistavassa oletetaan, että joka ikinen mahdollinen ongelma ja haaste on tiedossa ennen ohjelmistokehityksen aloittamista. Vesiputousmallinen ohjelmistokehitys perustuu raskaan dokumentaation luomiseen ja ylläpitämiseen sekä ennalta määriteltyjen vaatimusten täydelliseen havainnoimiseen (Mahapatra & Mangalaraj, 2005; Balaji & Murugaiyan, 2012, Petersen, Wohlin & Baca, 2009). Alla olevassa kuvassa havainnoidaan perinteisen ohjelmistokehitysprosessin lineaarista kulkua:



Kuva 1. Vesiputousmallinen ohjelmistokehitysprosessi (Pressman, 2005)

Schwaber (1997) mukaan vesiputousmallisen ohjelmistokehityksen lineaarinen jatkumo on prosessin suurin ongelma, sillä prosessissa ei ole käytäntöjä, jolla pystyttäisiin vastaamaan ennalta määritetyn määritysten ulkopuolisiin, arvaamattomiin vaatimusten muutoksiin. Vesiputousmallinen ohjelmistokehitys ei siis sovellu sellaisenaan liiketoiminta-alueisiin, jossa kaikkia ohjelmiston nyansseja, sidosryhmien tarpeita ei voida määritellä etukäteen. Vaikka sidosryhmien sen hetkiset tarpeet voitaisiinkin määritellä tarkasti etukäteen, on mahdotonta ennustaa, minkälaisia muutostarpeita tulevaisuus tuo tullessaan (Cao & Ramesh).

Näiden edellä mainittujen haasteiden vuoksi seitsemäntoista ohjelmistokehitysalan ammattilaista kehittivät prosessin, jota hyödyntämällä organisaatiot kykenevät vastaamaan sidosryhmien tarpeiden sekä järjestelmävaatimusten nopeisiin muutoksiin (Larman, 2007). Dybå & Dingsøyr (2008) mukaan yksinkertaisuudessaan ketterässä ohjelmistokehityksessä on kyse ohjelmiston sopeutumis- ja muutoskyvystä nykyajan nopeasti muuttuvassa liiketoimintaympäristössä.

Ketterässä ohjelmistokehityksessä kaikkia vaatimuksia ei tarvitse määritellä ennen kehitystyön aloittamista. Cao & Ramesh (2010) mukaan raskaan vaatimusmäärittelyprosessin sijaan ohjelmistoprojekti on jaettu pienempiin osakokonaisuuksiin, jotka toteutetaan lyhyissä, aikarajoitetuissa iteraatioissa toimittajan ja sidosryhmien välisellä yhteistyöllä. Ohjelmistoa tuotetaan inkrementaalisesti, jossa asiakkaan sidosryhmät ovat mukana kehitysprojektin jokaisessa vaiheessa (Robertson & Robertson, 2006). Bjarnason et.al (2011) mukaan iteraatioprosessin tarkoitus on minimoida riskejä, joita syntyy vaatimusmäärittelyiden aikana. Ketterän ohjelmistokehityksen prosessit ja periaatteet avataan kappaleessa 2.1.1. Ketterän ohjelmistokehityksen periaatteita sovelletaan monissa erilaisissa projektinhallinnan viitekehityksissä. Nämä viitekehitykset eroavat toisistaan, mutta kaikilla viitekehityksillä pääperiaatteina toimii joustavuus ja kyky vastata muutoksiin nopeasti. Scrum-metodi on ohjelmistokehitysalalla yksi vakiintuneimmista viitekehityksistä. Scrum-metodin sisältö avataan tarkemmalla tasolla kappaleessa 2.1.2. Mitä periaatteita ketterä ohjelmistokehitys pitää sisällään? Tarkastellaan kysymyksen vastauksia seuraavassa kappaleessa.

### **2.1.1 Ketterän ohjelmistokehityksen periaatteet**

Ketterät menetelmät kehitettiin vastaamaan haasteisiin, joita esiintyi perinteisen ohjelmistokehityksen lineaarisuuden luonteen vuoksi. Perinteinen ohjelmistokehitys perustuu raskaaseen dokumentointiin ja sen ylläpitoon, jossa kaikki sidosryhmien vaatimukset on kirjattu ennen varsinaista ohjelmistokehityksen alkua. Ohjelmistoliiketoiminnan alalla pian huomattiin, että perinteisiä menetelmiä käyttämällä ei saavutettu haluttuja tuloksia. Näin ollen ketterät menetelmät kehitettiin, jossa pyritään maksimoimaan ohjelmiston arvontuotto pilkkomalla kehityskomponentit pienempiin osakokonaisuuksiin. Ketterät menetelmät koostuvat periaatteista, jotka on havainnoitu kahdessa alla olevassa listassa:

Ketterän ohjelmistokehityksen arvot	Perinteisen ohjelmistokehityksen arvot
Yksilöt ja kanssakäyminen	Menetelmä ja työkalut
Toimiva ohjelmisto	Kattava dokumentaatio
Asiakasyhteistyö	Sopimusneuvottelut
Vastaaminen muutoksiin	Pitäytyminen suunnitelmassa

Kuva 2. Ketterien ja perinteisten ohjelmistokehityksen periaatteet (Conboy et.al, 2010)

Periaate 1	Tärkein tavoitteemme on tyydyttää asiakas toimittamalla tämän tarpeet täyttäviä versioita ohjelmistosta aikaisessa vaiheessa ja säännöllisesti.
Periaate 2	Otamme vastaan muuttuvat vaatimukset myös kehityksen myöhäisessä vaiheessa. Ketterät menetelmät hyödyntävät muutosta asiakkaan kilpailukyvyyn edistämiseksi.
Periaate 3	Toimitamme versioita toimivasta ohjelmistosta säännöllisesti, parin viikon tai kuukauden välein, ja suosimme lyhyempää aikaväliä.
Periaate 4	Liiketoiminnan edustajien ja ohjelmistokehittäjien tulee työskennellä yhdessä päivittäin koko projektin ajan.
Periaate 5	Rakennamme projektit motivoituneiden yksilöiden ympärille. Annamme heille puitteet ja tuen, jonka he tarvitsevat ja luotamme siihen, että he saavat työn tehtyä.
Periaate 6	Tehokkain ja toimivin tapa tiedon välittämiseksi kehitystiimille ja tiimin jäsenten kesken on kasvokkain käytävä keskustelu.
Periaate 7	Toimiva ohjelmisto on edistymisen ensisijainen mittari.
Periaate 8	Ketterät menetelmät kannustavat kestävään toimintatapaan. Hankkeen omistajien, kehittäjien ja ohjelmiston käyttäjien tulisi pystyä ylläpitämään työtahtinsa hamaan tulevaisuuteen.
Periaate 9	Teknisen laadun ja ohjelmiston hyvän rakenteen jatkuva huomiointi edesauttaa ketteryyttä.
Periaate 10	Yksinkertaisuus - tekemättä jätettävän työn maksimointi - on oleellista.
Periaate 11	Parhaat arkkitehtuurit, vaatimukset ja suunnitelmat syntyvät itseorganisoituvissa tiimeissä.
Periaate 12	Tiimi tarkastelee säännöllisesti, kuinka parantaa tehokkuuttaan, ja mukauttaa toimintaansa sen mukaisesti.

Kuva 3. Ketterien menetelmien periaatteet (Agile Foundation: Principles, practices and frameworks, 2015).



Ketterässä ohjelmistokehityksessä korostetaan yksilöitä ja yksilöiden välistä vuorovaikutusta prosessien ja käytettyjen työkalujen sijaan. Sen sijaan että yritettäisiin sovittaa ihmisiä mukautumaan ennalta määritettyyn prosessiin, ketterissä menetelmissä pyritään muokattamaan prosesseja niin, jotta ne palvelisivat yksilöjä. Yksilöiden välisellä vuorovaikutuksella pyritään korostamaan yhteisöllisyyttä ja parantamaan ihmissuhteita, jotka toisaalta parantavat kommunikaatiota yksilöiden välillä (Cao & Ramesh, 2010). Kommunikointi, joka on elinehto onnistuneille ohjelmistoprojekteille, on hyvä esimerkki havainnoida prosessimaisen ja yksilömallisen kommunikaation eroja (Abrahamsson et.al, 2002). Prosessimaisessa tavassa kommunikointi on yleisesti tarkasti ajoitettua ja vaatii erityistä sisältöä, kun taas yksilömallisessa tavassa kommunikointi tapahtuu tarvittaessa ja on näin ollen sujuvampaa. Ketterän ohjelmistotiimin jäsenet ovat tyypillisesti myös itseohjautuvia. (Abrahamsson et.al, 2002; Cockburn & Highsmith, 2001).

Perinteinen ohjelmistokehitys pohjautuu laajan ja raskaan dokumentaation luontiin ja ylläpitoon, jotka voivat vaikuttaa negatiivisesti ohjelmiston laatuun ja aikatauluihin. Laajan dokumentaation ylläpito on hankalaa ja haasteellista. Paulk (2002) argumentoi, että ainoa ajan tasalla oleva dokumentaatio löytyy ohjelmistokoodista. Abrahamsson et.al (2002) mainitsee, että kehittäjien vastuulla on pitää ohjelmistokoodi helposti ymmärrettävänä ja luettavana, jotta dokumentaation määrää voitaisiin pienentää. Näin ollen ketterissä menetelmissä dokumentaation määrää pyritään minimoimaan. Käyttäjätarinat ovat yleinen dokumentaation muoto projekteissa, joissa sovelletaan ketteriä menetelmiä. Raskaan dokumentaation sijaan ketterässä ohjelmistokehityksessä keskitytään toimivan ja oikeanlaisen ohjelmiston kehitykseen. Ketterissä menetelmissä tarvitaan myös dokumentaatiota, mutta dokumentaation laajuus ja kattavuus on tapauskohtaista. Ketterissä menetelmissä ohjelmiston kehitys on jaoteltu pienempiin iteraatioihin, joiden aikana tuotetaan ohjelmistoinkrementtejä arvioitavaksi sidosryhmille. Ketterissä menetelmissä pyritään maksimoimaan liiketoiminnan arvoa kehittämällä ohjelmistoa, sillä asiakkaat mittaavat onnistumista ja arvoa tuotettujen ohjelmistokomponenttien perusteella (Beck et al., 2001; Larman, 2004). Ohjelmistoinkrementtien jatkuva toimitus ja arviointi toimii ketterien menetelmien kulmakivenä (Abrahamsson et.al, 2002; Beck et.al, 2001).

Sopimusneuvottelujen sijaan ketterissä menetelmissä korostetaan asiakasyhteistyötä. Asiakasyhteistyöllä pyritään parantamaan arvontuottoa, sillä asiakasarvon tuotto toimii mittarina ketterän ohjelmistoprojektin onnistumiselle. (Highsmith & Cockburn, 2001). Abrahamsson et.al (2002) mukaan ketterissä menetelmissä pyritään korostamaan kehittäjien ja asiakkaiden välistä vuorovaikutusta sopimusneuvottelujen sijaan. Asiakasyhteistyöllä varmistetaan, että asiakas on jokaisessa kehitysvaiheissa mukana, jolloin asiakas pystyy seuraamaan ja arvioimaan julkaistuja inkrementtejä. Näin ollen asiakas pystyy vaikuttamaan kehitettyihin ohjelmistoihin aikaisessa vaiheessa. Asiakasyhteistyötä hyödyntämällä projektissa kuluu vähemmän aikaa erilaisiin tehtäviin, jotka liittyvät sidosryhmien raportointiin ja dokumentointiin (Cockburn, 2002).

Perinteisessä kehityksessä pyritään pidättäytymään ennalta määritetyssä suunnitelmassa. Pienikin muutos saattaa käynnistää laajemman prosessin, jonka vuoksi perinteisessä kehityksessä muutosten teko on hankalaa ja täten kallista. Asiakkailla tarpeet saattavat kuitenkin muuttua ohjelmistoprojektin aikana, jonka vuoksi ketterissä menetelmissä pyritään vastaamaan mahdollisiin muutoksiin. Ohjelmistokokonaisuuden jaottelulla pienempiin osakokonaisuuksiin ketteriä menetelmiä hyödyntämällä voidaan vastata mahdollisiin muutoksiin nopeammin ja kustannustehokkaammin. Asiakasyhteistyötä hyödyntämällä varmistetaan jatkuvan palautteen saaminen, jolloin voidaan jo varhaisessa vaiheessa havaita mahdolliset muutostarpeet. Varhaisten havainnon vuoksi muutosten aiheuttamia vaikutuksia voidaan hallita tehokkaammin. (Abrahamsson et al., 2002, Highsmith, 2002).

### **2.1.2 SCRUM**

Scrum-metodilla viitataan projektinhallinnan viitekehykseen, jossa sovelletaan ketterän ohjelmistokehityksen periaatteita. Scrum-metodi on ohjelmistoalalla vakiintunut ja yleisimmin käytetty viitekehys, jota käytetään laajasti erityyppisissä ohjelmistokehitysprojekteissa. Yksinkertaisuudessaan Scrum-metodilla viitataan viitekehykseen, joka mahdollistaa ketterän periaatteiden mukaisen hallinnan ohjelmisto- ja tuotekehitysprojekteissa (Cobb, 2015; Schwaber, 2007). Scrum-metodissa hyödynnetään iteratiivista kehitysmallia, joka

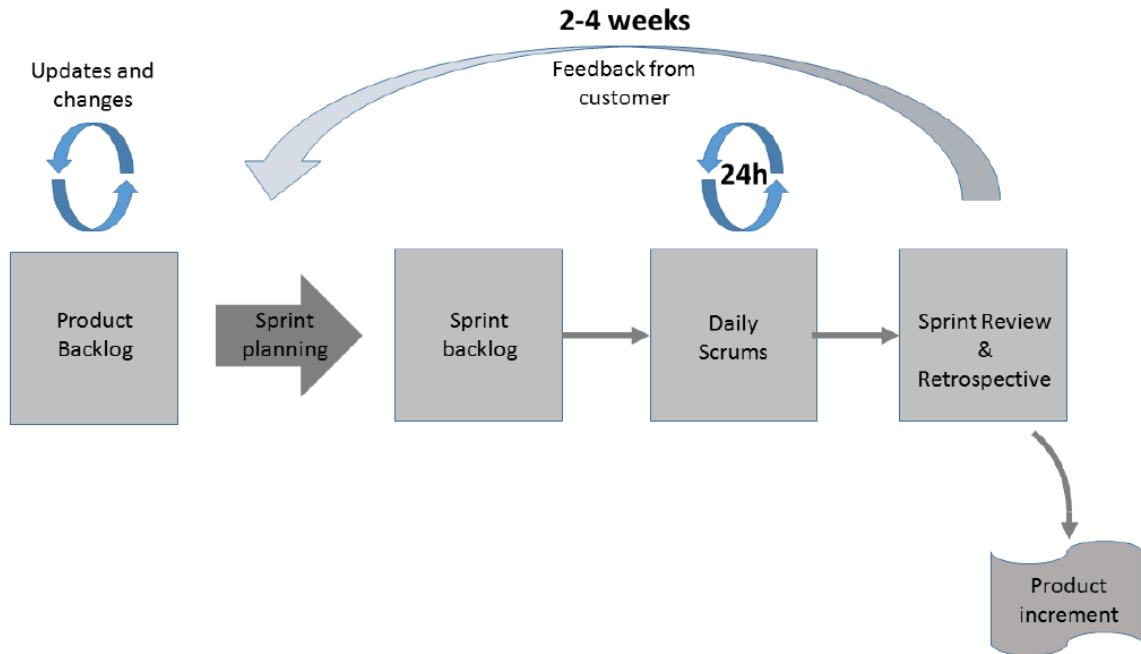
perustuu sidosryhmien väliseen jatkuvaan kommunikointiin, yhteistyöhön ja tuotosten arviointiin. Iteratiivisuuden lisäksi ohjelmistoa kehitetään inkrementaalisesti, jossa tietyn ajan välein julkaistaan sidosryhmille arvioitavaksi toimivia inkrementtejä ohjelmistosta (Abrahamsson et.al, 2002). Seuraavassa alikappaleessa havainnoidaan, mitä tekijöitä SCRUM-metodin prosessi sisältää.

### **2.1.2.1 SCRUM-metodin prosessi**

Kuten kappaleessa 2.1.2 mainittiin, Scrum-viitekehys perustuu iteratiiviseen ja inkrementaaliseen kehitykseen. SCRUM-tiimi muodostuu seuraavista rooleista, jotka on havainnoitu alla olevassa listassa:

- 1) Product Owner (Tuoteomistaja)
- 2) Scrum Master
- 3) Scrum-tiimi

Scrum menetelmässä tuoteomistajan tehtävä on ymmärtää sidosryhmien liiketoimintatarpeet, sekä muuntaa nämä tarpeet vaatimuksiksi tuotteen kehitysjonoon. Tuoteomistaja hallitsee yleisesti tuotteen kehitysjonoa, sekä vastaa vaatimusten priorisoinnista. Näin ollen, tuoteomistaja vastaa viime kädessä siitä, mitä ominaisuuksia tuotteeseen kehitetään missäkin järjestyksessä. Tuoteomistajan tärkein tehtävä on varmistaa, että kehitettävä ohjelmisto tuottaa mahdollisimman paljon arvoa yritykselle ja sen asiakkaille. (Kettunen, 2009; Measey et al., 2015). Scrum Masterin keskeinen tehtävä on eliminoida mahdolliset rajoitteet, joita saattaa kehitystyötä tekeville henkilöille syntyä. Näin ollen Scrum Master huolehtii kehitystiimin työympäristön optimaalisuudesta sekä kehitystöiden edistymisestä. Scrum Master järjestää ja ohjaa kokouksia ja päättää yhdessä tuoteomistajan kanssa mitä vaatimuksia tuotteen kehitysjonosta otetaan kehityksen alle seuraavassa sprintissä (Rising & Norman, 2000). Scrum-tiimillä viitataan joukkoon itseorganisoituvia henkilöitä, jotka kehittävät ja toimittavat inkrementtejä vaatimusten mukaisesti. Tiimi koostuu moniosaajista, joilla on valtuudet päättää työn organisoinnista ja toteutustavoista (Measey et al., 2015). Alla olevassa kuvassa havainnoidaan SCRUM-metodin prosessia:



Kuva 4. Scrum-metodin prosessikaavio (Schwaber, 2007).

SCRUM-metodissa tuotteen kehitysjonoon kerätään vaatimuksia, joita kehitettävään järjestelmään halutaan kehittää. Kehitystyöt on jaettu ennalta määritettyihin ajanjaksoihin, joita kutsutaan sprinteiksi. Kontekstin mukaan sprintin ajanjakso vaihtelee eri kehitystiimien välillä. Sprinteissä kuitenkin jokainen jakso kestää aina kahdesta neljään viikkoon (Cooper & Sommer, 2016). Sprinttien aikana SCRUM-tiimi sitoutuu kehittämään ja toimitamaan ennalta sovittuja vaatimuksia toimiviksi ohjelmistokomponenteiksi, jotka toimitetaan arvioitavaksi tarvittaville sidosryhmille. (Cobb, 2015; Schwaber, 2007; Abrahamsson et.al, 2002). Tuoteomistajan priorisoinnin perusteella tuotteen kehitysjonosta otetaan tiettyjä vaatimuksia kehitettäväksi, joiden kehityksestä ja aikatauluarvioista keskustellaan sprintin suunnittelun aikana scrum-tiimin kanssa. Sprintin suunnittelun aikana scrum-tiimi päättää kehitettävän työn määrästä arvioimalla vaatimusten kehitykseen vievää aikaa jonka jälkeen nämä vaatimukset viedään tuotteen kehitysjonosta sprintin kehitysjonoon. SCRUM-metodiin kuuluu päivittäiset tilannepalaverit, jonka aikana kehitystiimiltä odotetaan vastaukset seuraaviin kysymyksiin:

- 1) Mitä teit eilen?
- 2) Mitä suunnittelit tekeväsi tänään?
- 3) Mitkä ovat mahdollisia hidasteita?

Ennalta määritetyn ajanjakson jälkeen ajanjakson jälkeen pidetään istunto, jossa sprintin aikana tehtyjä kehitystöitä katselmoidaan ja arvioidaan. Tätä vaihetta kutsutaan sprintin katselmoinniksi. Katselmoinnin aikana SCRUM-tiimin jäsenet esittävät tuotoksiaan tarvittaville sidosryhmille. Katselmoinnin jälkeen pidetään yhteinen retrospektiivi, jossa pyritään tunnistamaan tapoja, joiden avulla kehitystyötä voidaan jatkossa optimoida. (Fitzgerald et al., 2006, Measey et al., 2017).

## 2.2 Hajautettu ketterä ohjelmistokehitys

Hajautettujen ohjelmistokehitystiimien määrä on kasvattanut suosioitaan viime vuosien aikana. Yksinkertaisuudessaan hajautettu ohjelmistokehitys on ohjelmistokomponenttien kehittämistä useammassa kuin yhdessä fyysisessä paikassa. Tuotantokustannusten säästöt ja globaali työvoima houkuttelevat yrityksiä laajentamaan ohjelmistotiimejä halvemman tuotannon maihin. (Vallon et.al, 2018) Työvoima kehittyvissä maissa on noin 35–40 % alhaisempi verrattuna kehittyneisiin maihin. (Beulen, 2010) Tuotantosäästöjen lisäksi yritykset pyrkivät hankkimaan osaavia kehittäjiä, jotka kykenevät työskentelemään ketterien menetelmien periaatteiden mukaisesti itseohjautuvasti. Osaavien kehittäjien löytäminen paikallisesti on haasteellista, jonka vuoksi yritykset pyrkivät hyödyntämään globaalia työvoimaa rekrytoinneissa. Hajautettuja ohjelmistokehitystiimejä syntyy myös ilman globalisointumista. Yrityksillä saattaa olla esimerkiksi monia toimipisteitä, tai jopa samojen toimipisteiden työntekijät saattavat tehdä töitä myös esimerkiksi etänä. Hajautetussa ketterässä ohjelmistokehityksessä on kuitenkin omat haasteensa. Kulttuurien väliset erot ja maantieteelliset etäisyydet tuovat haasteita, jotka johtavat huonolaatuihin viestintään, mikä vaikuttaa laajasti ohjelmiston laatuun ja kehitykseen. Hajautetun ketterän ohjelmistokehityksen haasteet avataan tarkemmalla tasolla kappaleessa 2.2.1.

Kuten kappaleessa **2.1.1** mainittiin, ketterät ohjelmistokehityksen menetelmät toimivat parhaiten tiimeissä, jotka työskentelevät fyysisesti samassa tilassa (Ramesh et.al, 2010). Alan kirjallisuuden mukaan hajautetuissa ohjelmistokehitystiimeissä pyritään soveltamaan ketteriä projektinhallinta- ja vaatimusmäärittelymenetelmiä. Seuraavassa kappaleessa tarkastellaan riskejä ja haasteita, joita syntyy hajautetuissa ketterissä ohjelmistokehitystiimeissä.

### **2.2.1 Hajautetun ketterän ohjelmistokehityksen hyödyt ja haasteet**

Nykyään yrityksillä täytyy olla valmius tuottaa laadukkaita sovelluksia alhaisemmilla hinnoilla, jotta yritykset pystyvät kilpailemaan globaalissa tässä nykyisessä globaalissa markkinataloudessa. Hajautetut ohjelmistokehitystiimit sekä ketterien menetelmien käyttö ohjelmistoprojekteissa ovat olleet kovassa kasvussa viimeisten vuosien aikana. VersionOne suorittaman Agile Development Survey -tutkimuksen mukaan 57 % vastanneista yrityksistä vastasivat, että heidän ohjelmistokehitystiimensä on hajautettu fyysisesti eri paikkoihin. Tämän lisäksi 41 % vastanneista mainitsivat, että heidän ohjelmistokehitystiimensä joko käyttävät tai suunnittelevat käyttävänsä ketteriä menetelmiä hajautetuissa tiimeissä. Tuotantosäästöjen ja osaavien ammattilaisten saavuttamisen takia yritykset hyödyntävät globaalia työvoimaa, joiden takia ohjelmistokehitystyö hajautetaan fyysisesti eri paikkoihin. Melkein kaikissa ketterän ohjelmistokehityksen viitekehityksissä oletetaan, että ohjelmistoa kehittävät ihmiset sijaitsevat fyysisesti samassa huoneessa (Cao & Ramesh, 2010; Bjarnason et.al, 2011). Ketterien menetelmien käyttö hajautetuissa tiimeissä tuo sekä hyötyjä että haasteita, jotka avataan tarkemmalla tasolla seuraavissa alikappaleissa.

#### **2.2.1.1 Hyödyt**

Ketterien menetelmien iteratiivisuuden luonteen vuoksi mahdolliset ongelmat voidaan havaita varhaisessa vaiheessa, jolloin virheisiin voidaan reagoida nopeammalla aikataululla. Lyhyiden iteraatioiden vuoksi ketterien menetelmien periaatteiden käyttö parantaa kommunikaatiota hajautetuissa ohjelmistokehitystiimeissä, jolloin tiimien jäsenet kykenevät paremmin saavuttamaan yhteiset tavoitteet. (Paasivaara & Lassenius, 2006). Sprinttien kat-

selmoinnissa hajautettujen tiimien kehittäjät esittävät kehitettyjä ominaisuuksia, joka puolestaan edistää tiedonkulkua tarvittaville sidosryhmille. Yksilöiden välisellä vuorovaikutuksella pyritään korostamaan yhteisöllisyyttä ja parantamaan ihmissuhteita, jotka toisaalta parantavat kommunikaatiota ja luottamusta tiimin jäsenten välillä. Sujuva kommunikointi tiimin välillä myös kasvattaa kehitetyn ohjelmiston laatua. (Passivaara & Lassenius, 2006; Shrivastava & Date, 2010).

### **2.2.1.2 Haasteet**

Edellisessä kappaleessa mainittujen hyötyjen lisäksi ketterien menetelmien käytöstä hajautetuissa ohjelmistokehitystiimeissä syntyy myös haasteita. Ketterien menetelmien periaatteiden mukaisesti tehokkain ja toimivin tapa tiedon välittämiseksi on kasvokkain käytävät keskustelut. Hajautetuissa tiimeissä tämän edellä mainitun periaatteen toteuttaminen ei ole sellaisenaan mahdollista, kun tiimien jäsenet sijaitsevat fyysisesti eri toimipisteissä. Ketterässä ohjelmistokehityksessä dokumentaation määrä pyritään pitämään mahdollisimman vähäisenä. Jatkuva kommunikointi toimii ketterissä tiimeissä pääasiallisena tiedonvälitystapana raskaan dokumentaation sijaan. Perusteellisen dokumentaation puute on kuitenkin haasteellista hajautetuissa tiimeissä, sillä hajautetun luonteen vuoksi tiimien välinen kommunikointi kärsii. Shrivastava & Date (2010) mukaan monet offshore-kehitystiimit suosivat perinteistä lähestymistapa, jossa kaikki vaatimukset on dokumentoitu. Dokumentaation määrä ei saa kuitenkaan kasvaa liian suureksi ja yksityiskohtaiseksi, sillä tarvittavan tiedon löytäminen koituu hankalaksi (Shrivastava & Date, 2010; Alqhatani et.al, 2013). Tarvittavan dokumentaation määrän ja tiedon tarkkuuden tasapainottaminen on haasteellista. Jossakin tietyssä tiimissä jokin määritelty dokumentaation tarkkuus saattaa olla juuri sopiva, kun taas jossakin toisessa tiimissä vähempikin riittäisi.

Aika- ja maantieteellisistä eroista johtuvat tekijät heikentävät viestinnän laatua ja ylipääntänsä mahdollisuutta tehokkaaseen viestintään (Shrivastava & Date, 2010; Alzoubi & Gill, 2016; Hossain et.al, 2009; Kamaruddin & Arshad, 2010). Kamaruddin & Arshad (2010) argumentoivat, että vähäinen kasvoittain käytävä keskustelu johtaa luottamusongelmiin tiimin välillä, joka taas toisaalta heikentää tehokkuutta. Suoran kommunikoinnin puute asi-

akkaiden ja kehittäjien välille saattaa johtaa vaatimusten väärinymmärryksiin. Tämä tarkoittaa, että pahimmillaan kehittäjät saattavat joutua arvailemaan vaatimuksia (Kamaruddin & Arshad, 2010; Alqahtani, 2013). Joissain tapauksissa ulkoistetut toimijat eivät välttämättä uskalla sanoa rehellisesti mielipiteitään. Kajkon-Mattson et.al (2010) argumentoivat, että hajautetuissa tiimeissä on usein haasteita suoraan puhumisen tai rehellisyyden kanssa.

Shrivastava & Date (2010) argumentoivat, että offshore-kehitystiimeissä suositaan enemmän perinteisiä projektinhalinta- ja vaatimusmäärittelymenetelmiä. Jatkuvasti muuttuvien liiketoimintatarpeiden vuoksi täysin perinteisiin menetelmiin ei voida nojautua, sillä tietyissä hankkeissa vaaditaan jäykkyyden lisäksi myös ketteryyttä. Seuraavissa kappaleissa havainnoidaan ohjelmistoliiketoiminta-alalla käytettyjä vaatimusmäärittelymenetelmiä ja niihin liittyviä haasteita.

### **2.3 Vaatimusmäärittelyt**

Yksinkertaisuudessaan onnistuvalla vaatimusmäärittelyprosessilla viitataan asiakkaiden ja sidosryhmien todellisten tarpeiden ymmärtämiseen. Vaatimusmäärittelyllä viitataan mekanismiin, jossa ohjelmistoa tilaavan asiakkaan tarpeet saadaan mahdollisimman hyvin selville aikaisessa vaiheessa, jotta tuotettu ohjelmisto aidosti palvelisi aidosti asiakkaan tarpeita. Polini (2010) suorittaman tutkimuksen mukaan noin 37 % haastavien järjestelmien ohjelmistokehitysvaiheen ongelmista liittyy tavalla tai toisella vaatimusmäärittelyssä tehtyihin virheisiin. Jotta tarpeet voitaisiin mahdollisimman hyvin sisäistää ja ymmärtää, tulee vaatimusmäärittelyistä vastuussa olevilta henkilöiltä löytyä sen liiketoiminnan substanssi-osaamista, jossa kehitettävää ohjelmistotuotetta tullaan käyttämään. Vaatimusmäärittelyt voidaan jakaa seuraavaan kahteen kategoriaan:

1. Perinteinen / vesiputousmallinen vaatimusmäärittely
2. Ketterä vaatimusmäärittely

Perinteinen vaatimusmäärittelyprosessi sisältää sidosryhmien tarpeiden mallintamista, analysointia, vaatimukseen liittyvää neuvottelua sekä dokumentointiin liittyviä tehtäviä



(Sommerville, 2004; Glinz, 2007). Tämän lisäksi vaatimuksia täytyy hallita ja kontrolloida. Hallinta sisältää vaatimusten dokumentointiin ja validointiin liittyviä tehtäviä. Näillä edellä mainituilla tehtävillä voidaan varmistaa, että dokumentaatiossa vahvistetut vaatimukset vastaavat neuvoteltuja vaatimuksia (Cheng & Atlee, 2007). Ketterän vaatimusmäärittelyn ideana on se, että vaatimuksia täytyy pystyä lisäämään tai muuttamaan kehitystyön aikana ilman raskasta prosessia. Ketterässä vaatimusmäärittelyssä uskotaan siihen, ettei kaikkia mahdollisia sidosryhmien tarpeita voida tietää ennen kehitystyön alkua. Vesiputousmallisen ja ketterän vaatimusmäärittelyprosessin vaiheet avataan tarkemmalla tasolla kappaleissa **2.5 ja 2.6**.

Jotta voidaan ymmärtää, mitä vaatimusmäärittelyt tarkalleen tarkoittavat, on ensinnäkin ymmärrettävä, minkä tyyppisiä vaatimusmäärittelyjä on olemassa. Vaatimukset voidaan jakaa kahteen osa-alueeseen, joita ovat toiminnalliset ja ei-toiminnalliset vaatimukset (Glinz, 2007). Seuraavassa alikappaleessa avataan tarkemmalla tasolla, mitä näillä vaatimusmäärittelytyypeillä tarkoitetaan.

### **2.3.1 Vaatimusmäärittelytyypit**

Toiminnallisilla vaatimuksilla viitataan johonkin tiettyyn toimintaan tai prosessiin, jotka on pystyttävä ohjelmistolla suorittamaan. (Sommerville, 2004; Glinz, 2007). Toiminnallisilla vaatimuksilla viitataan prosessiin, jota havainnoidaan alla olevassa listassa:

- 1) Käyttäjä käskee järjestelmää tekemään tietyn toiminnon
- 2) Järjestelmä prosessoi pyyntöä määritetyllä tavalla
- 3) Järjestelmä tuottaa määritellyt tulokset

Toiminnallisilla vaatimuksilla viitataan funktioihin, joita käyttäjät suorittavat järjestelmässä. Toiminnallisilla vaatimuksilla kuvataan järjestelmän toimintaa tietyn käyttäytymisen näkökulmasta tiettyjen ehtojen täytyessä (Glinz, 2007). Alla olevassa listassa havainnoidaan esimerkinomaisesti toiminnallisia vaatimuksia:

- Järjestelmään tunnistaudutaan sormenjälkitunnistuksella

- Käyttäjän pitää pystyä muuttamaan ohjelmiston taustan väriä
- Vain tietyt käyttöoikeudet omaava henkilö pystyy lisäämään muita käyttäjiä järjestelmään

Toiminnallisten vaatimusten lisäksi vaatimusmäärittelyprosessi sisältää myös ei-toiminnallisia vaatimuksia. Ei-toiminnallisten vaatimukset asettavat toiminnallisille vaatimuksille reunaehdoja ja rajoituksia, joiden puitteissa toiminnalliset vaatimukset voidaan toteuttaa. Ei-toiminnallisissa vaatimuksissa asetetaan ehdot, jotka toiminnallisten vaatimusten on täytettävä, jotta toiminnallisuus voidaan toteuttaa (Sommerville, 2004; Glinz, 2007). Toiminnallisten vaatimusten tarkoitus on tarkentaa mitä järjestelmällä voidaan suorittaa, kun taas ei-toiminnallisilla vaatimuksilla tarkennetaan millä tavalla toiminnallisia vaatimuksia pystytään suorittamaan (Adams, 2015). Alla olevassa listassa havainnoidaan esimerkinomaisesti ei-toiminnallisia vaatimuksia:

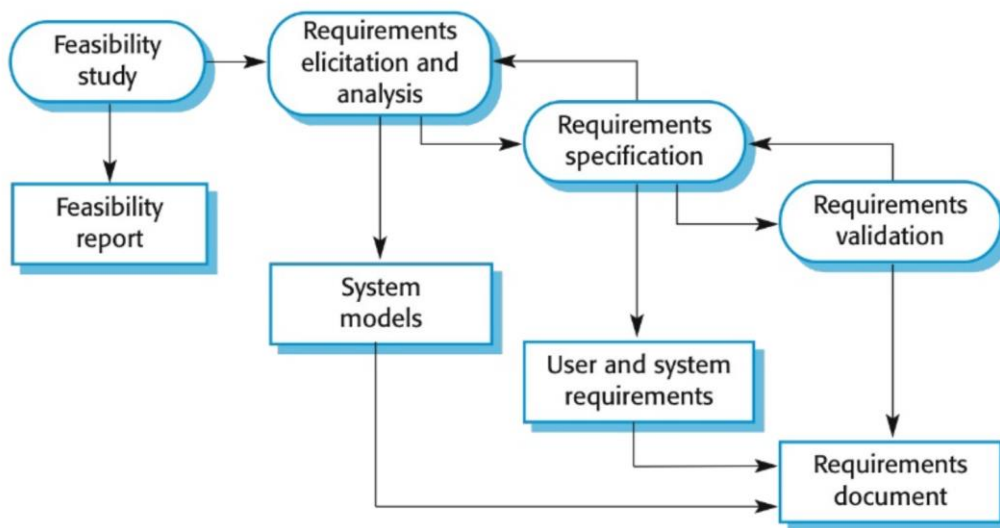
- Käyttäjä pakotetaan vaihtamaan salasanaa kolmen kuukauden välein
- Ohjelmistoa avattaessa käyttöliittymän latautumisaika ei saa ylittää kahta sekuntia
- Jokaisesta epäonnistuneesta kirjautumisyrityksestä täytyy jäädä merkintä lokitiedostoihin
- Järjestelmän täytyy kyetä skaalautumaan käyttäjämäärän kasvaessa

## 2.4 Perinteinen vaatimusmäärittelyprosessi

Kuten kappaleessa 2.4 mainittiin, perinteisellä vaatimusmäärittelyllä viitataan mekanismiin, jossa ohjelmistoa tilaavan asiakkaan tarpeet saadaan mahdollisimman hyvin selville aikaisessa vaiheessa ennen varsinaisen ohjelmistokehitystyön aloittamista (Kotonya & Sommerville, 2001; Nuseibeh & Easterbrook, 2000). Vaatimusmäärittelyprosessin päätaivoitteena on varmistaa, että suunniteltu ja määritelty järjestelmä vastaa asiakkaan sidosryhmien tarpeita sekä täyttää asiakkaan odotukset. Perinteinen vaatimusmäärittelyprosessi koostuu asiakkaiden tarpeiden ymmärtämisestä, vaatimusten analysoinnista ja neuvottelusta, vaatimusten validoinnista sekä vaatimusten hallinnasta. (Pressman, 2000). Kotonya & Sommerville (2000) mukaan vaatimusmäärittelyprosessi voidaan jakaa toimintoihin, jotka havainnoidaan alla olevassa listassa:

- Toteutettavuustutkimus (Feasibility Study)
- Vaatimusten kartoittaminen ja analysointi (Requirements Elicitation and Analysis)
- Vaatimusten spesifiointi ja dokumentointi (Requirements Specification and Documenting)
- Vaatimusten validointi (Requirements Validation)
- Vaatimusten hallinta (Requirements Management)

Alla olevassa kuvassa havainnoidaan perinteistä vaatimusmäärittelyprosessia:



Kuva 5. Perinteinen vaatimusmäärittelyprosessi (Sommerville, 2001).

### Toteutettavuustutkimus

Uuden järjestelmän tai ominaisuuden kehitysprosessi alkaa toteutettavuustutkimuksella, jonka tarkoituksena on selvittää vaatimuksen toteuttamiskelpoisuutta. Toteutettavuustutkimuksen aikana analysoidaan laajasti vaatimukseen liittyviä tekijöitä. Toteutettavuustutkimuksessa pyritään havaitsemaan vaatimuksesta syntyvät edut ja mahdolliset haitat ennen varsinaisen vaatimusmäärittelyprosessin aloittamista. Toteutettavuustutkimuksen ai-

kana ei siis pureuduta vaatimukseen yksityiskohtaisesti. Sen sijaan tämän vaiheen tarkoituksena on saada kokonaisvaltainen ymmärrys vaatimuksista, sekä ottaa huomioon asiakkaan liiketoimintatarpeet, taloudelliset, tekniset sekä aikatauluihin liittyvät tekijät.

Toteutettavuustutkimuksen lopussa laaditaan raportti, joka sisältää järjestelmän kehityksen kannattavuuteen liittyviä tekijöitä. (Kotonya & Sommerville, 2001; Paetsch et.al, 2003).

### **Vaatimusten kartoittaminen ja analysointi**

Vaatimusten kartoittaminen ja analysointi on varsinaisen määrittelyprosessin ensimmäinen vaihe, mikäli uusi järjestelmä tai ominaisuus päätetään toteuttaa. Vaatimusten kartoittamisen aikana pyritään ymmärtämään eri sidosryhmien tarpeet ja tavoitteet keräämällä tietoa erilaisilla menetelmillä. Yleisesti ottaen tämä vaihe sisältää mahdollisten ratkaisujen löytämistä tarkastelemalla ongelmia. Kartoittamisen aikana pyritään myös tunnistamaan rajoitteet, joiden puitteissa vaatimukset voidaan toteuttaa. (Paetch et.al, 2003; Nuseibeh & Easterbrook, 2000). Lauesen (2002) mukaan vaatimusten kartoittaminen voidaan osaluokkiin, jotka on havainnoitu alla olevassa listassa:

- 1) Kartoitetaan sidosryhmien tarpeet ja tavoitteet
- 2) Kartoitetaan nykyiset prosessit ja selvitetään nykyisten prosessien ongelmakohdat
- 3) Kerätään yksityiskohtaisesti tietoa erilaisilla menetelmillä
- 4) Tarkastellaan mahdollisia ratkaisuja
- 5) Muunnetaan mahdolliset ratkaisutavat vaatimuksiksi

Kerättävän tiedon laadun ja käyttötarkoituksen mukaan vaatimusten kartoittamiseen voidaan käyttää erilaisia menetelmiä. Näitä menetelmiä ovat esimerkiksi haastattelut, kyselyt, käyttötapausten, havainnot, kohderyhmät ja työpajat (Lauesen, 2002; Sommerville, 2004). Näiden menetelmien tavoitteena on luoda kokonaisvaltaista selkeyttä vaatimukseen sekä poistaa mahdollisia seikkoja, jotka aiheuttavat hämmennystä ja epäselvyyttä. (Chua et.al, 2010). Vaatimusten kartoittamisen onnistuminen riippuu monesta eri tekijästä. Vaatimusmäärittelyprosessin onnistumisen kannalta on tärkeää sitouttaa avainhenkilöt, joilta löytyy

tarvittava osaaminen kartoittaa vaatimuksia. Vaatimusmäärittelyprosessiin osallistuvilla henkilöillä on oltava osaamista kehitettävän ohjelmiston liiketoiminta-alueelta, organisaation liiketoimintaprosesseista sekä mahdollisista järjestelmän asettamista rajoitteista. (Kotonya & Sommerville, 1998). Yksinkertaisuudessaan kartoitusvaiheen pyritään tuomaan esille kaikki sidosryhmien vaatimukset.

Seuraavaksi analysoidaan kartoitusvaiheen aikana esille tuotuja vaatimuksia. Paetch et.al (2003) mukaan vaatimuspyynnöistä syntyy väistämättä konflikteja, jotka liittyvät kehitettävän järjestelmän asettamiin rajoitteisiin, kustannuksiin tai aikarajoitteisiin. Analyysin aikana neuvotellaan toteutettavista vaatimuksista ohjelmiston toimittajan ja asiakkaan välillä. Tämän vaiheen aikaan pyritään tunnistamaan mahdollisia puuttuvia, päällekkäisiä sekä epärealistisia vaatimuksia. (Kotonya & Sommerville, 1998; Ahmad, 2008). Näin ollen analyysivaiheen aikana tarkastellaan vaatimusten tarpeellisuus ja toteuttamiskelpoisuus liiketoiminnan näkökannalta. Esitettyjä vaatimuksia myös priorisoidaan kustannusten ja aikataulurajoitteiden takia. Priorisointi edesauttaa asiakkaita keskittymään järjestelmän kannalta olennaisiin vaatimuksiin, sekä jättämään pois vaatimuksia, joita ei kannata toteuttaa. Toisin sanoen, analyysivaiheen tavoitteena on löytää mahdolliset ongelmat ja riskit, selvittää vaatimusten toteutettavuuskelpoisuus sekä luoda kattava ymmärrys vaatimuksista yksityiskohtaisesti. (Paetch et.al, 2003; Pressmann, 2000). Vaatimusten kartoitus- ja analyysivaihe on luonteeltaan iteratiivista, sillä analyysin vaiheen aikana saattaa nousta uusia vaatimuksia, jolloin analyysi ja neuvotteluvaiheesta siirrytään takaisin vaatimusten kartoitusvaiheeseen.

### **Vaatimusten spesifiointi ja dokumentointi**

Vaatimusten dokumentaation ja spesifikaation pääasiallinen tarkoitus on välittää tietoa vaatimuksista eri sidosryhmien välillä. Organisaatioissa määritellään laadultaan hyvinkin erilaisia vaatimuksia. Ohjelmistokehitysprojektin laajuus ja organisaatorakenne vaikuttaa myös dokumentaation määrään. Näin ollen vaatimusten dokumentoinnille ei ole olemassa

mitään standardia tapaa, sillä dokumentaation sisältö ja rakenne vaihtelee eri organisaatioissa (Paetch et.al, 2003). Toimittajan ja asiakkaiden välisessä viestinnässä suositetaan yleisesti korkean tason vaatimuksia, jossa keskitytään liiketoiminta- ja käyttäjätarpeisiin (Zhang et.l.al, 2010). Hyvä vaatimusmäärittely sisältää kuitenkin tiettyjä laatuksiteereitä, vaikka dokumentaation rakenteella ja sisällöllä on eroja organisaatioiden välillä. Lauesen (2002) mukaan vaatimusten pitäisi olla yksiselitteisiä, johdonmukaisia, täydellisesti kirjoitettuja, selkeitä, ymmärrettäviä, muokattavia sekä jäljitettäviä. Lauesen (2002) mukaan vaatimukset täytyy myös dokumentoida niin, jotta niihin ei jää minkäänlaista tulkinnan varaa. Vaatimusmäärittelyissä spesifikaatiodokumentaation pääasiallinen tarkoitus on välittää tietoja toimittajan ja asiakkaan välillä sekä toimittajan sidosryhmien kesken, jonka vuoksi tärkein laatuun vaikuttava asia on se, että dokumentaatio soveltuu käytettäväksi. Käytettävyydellä viitataan tekijään, jossa dokumentaation rakenne palvelee yleisöä jolle, dokumentaatio on tarkoitettu. (Wieggers & Beatty, 2013).

### **Validointi**

Vaatimusten validoinnin pääasiallinen tarkoitus on vahvistaa, että vaatimukset ovat johdonmukaisia, toteutettavissa ja realistisia. Validoinnin tarkoitus on myös välttyä virheiltä jo ennen ohjelmistokehityksen aloittamista. (Sommerville, 2004). Validointivaiheessa suoritetaan samantyyppisiä tehtäviä kuin analyysivaiheessa. Analyysivaiheessa vaatimukset ovat kuitenkin keskeneräisiä, kun taas validointivaiheen tarkoitus on hyväksyttää vaatimukset asiakkaalla ennen ohjelmistokehityksen aloittamista. Analyysivaiheessa pohditaan, että onko vaatimukset kirjattu oikealla tavalla, kun taas validointivaiheessa tarkastellaan, että vastaavatko dokumentoidut vaatimukset tosiasiallisesti asiakkaiden tarpeita. (Kotonya & Sommerville, 1998). Näin ollen validointivaiheessa tarkastellaan spesifikaatiodokumenttia siltä kannalta, että vastaako kirjatut vaatimukset tosiasiallisesti asiakkaan tarpeita järjestelmän rajoitteet huomioiden (Lauesen, 2002).

## Vaatimusten hallinta

Yritykset toimivat yhä enemmän liiketoimintaympäristöissä, jossa järjestelmätarpeet muuttuvat nopealla vauhdilla. Näin ollen yritysten tarpeet saattavat muuttua hyvinkin nopeasti, jolloin on oltava mahdollisuus muuttamaan vaatimuksia halitusti. Vaatimusten hallinnalla viitataan ennalta määrättyyn prosessiin, joiden avulla hallitaan vaatimuksiin tulleita mahdollisia muutoksia. Vaatimusten hallintaa harjoitetaan koko ohjelmistokehityksen elinkaaren aikana. Sommerville (2004) mukaan vaatimusten muutokset johtuvat yleensä organisaatioiden liiketoimintatarpeiden muutoksista. Lauesen (2002) mukaan projektin alussa määritellyt vaatimukset saattavat vanhentua, eikä niiden toteuttaminen ole välttämättä projektin lopussa enää tarpeellista. Sommerville (2004) mukaan mahdollisten vaatimusmuutoksia täytyy analysoida, sekä ottaa huomioon muutosten vaikutusta muihin projektin osa-alueisiin. Vaatimusten hallinnalla varmistetaan myös se, että mahdolliset muutokset kulkeutuvat ennalta määrätyn prosessin läpi, joka mahdollistaa vaatimusten jäljitettävyyden.

Perinteisellä vaatimusmäärittelyprosessilla tavoitellaan tietynlaista jämakkyttä. Perinteiset menetelmät sopivat parhaiten käytettäväksi hankkeissa ja ympäristöissä, joiden liiketoiminta-alat ovat kovasti säänneltyjä. Mahdolliset muutokset vaatimuksissa käynnistävät usein raskaan muutosprosessin, joka kasvattaa hankkeen kustannuksia. Näin ollen perinteisissä menetelmiä suositaan hankkeissa, missä loppukäyttäjien todelliset tarpeet voidaan tosiasiallisesti saada selville ennen varsinaisen kehitystyön alkua. Kaikissa hankkeissa kokonaisvaltaista etukäteismäärittelyä ei voida suorittaa, sillä monesti asiakkaatkaan eivät aina tiedosta omia tarpeitaan. Ketterissä vaatimusmäärittelyissä sen sijaan pyritään tunnistamaan kehityksen elinkaaren alussa hankkeen kannalta kaikista kriittisimmät ominaisuudet, jonka jälkeen tarkemmat vaatimukset luodaan iteraatioiden kautta. Seuraavassa kappaleessa havainnoidaan ketteriä vaatimusmäärittelymenetelmiä sekä niihin liittyviä haasteita.

## 2.5 Ketterä vaatimusmäärittelyprosessi

Viimeisten vuosien saatossa ketterät menetelmät ovat kasvattaneet suosiotaan ohjelmistokehitystä tekevissä yrityksissä. Cao & Ramesh (2010) argumentoivat, että ketteriä vaatimusmäärittelymenetelmiä soveltamalla ohjelmistoyritykset kykenevät vastaamaan sidosryhmien liiketoimintatarpeiden äkillisiin muutoksiin. Vesiputousmallisen vaatimusmäärittelyprosessi nojautuu dokumentaatioon ja sen validointiin, kun taas ketterä vaatimusmäärittely on iteratiivinen prosessi, jossa jatkuvasti iteratiivisesti kommunikoidaan toimittajan ja asiakkaan välillä. Ketterä vaatimusmäärittelyprosessi sisältää samankaltaisia vaiheita kuin perinteinenkin vaatimusmäärittelyprosessi. Perinteisessä ohjelmistokehityksessä vaatimusmäärittelyprosessi on niin ikään irrallinen toiminto itse ohjelmistokehityksestä. Sen sijaan Schön (2017) mukaan ketterässä ohjelmistokehityksessä vaatimusmäärittelyt ovat mukana kaikissa kehityssykleissä, joiden aikana vaatimukseen voi tulla tarvittaessa muutoksia.

Vaikka ketterien ja perinteisten vaatimusmäärittelymenetelmien tavat poikkeavat toisistaan, ovat tavoitteet kuitenkin yhtenäiset. Molemmissa menetelmissä tähdätään saamaan sellaisia vaatimuksia, jotka tosiasiallisesti vastaavat asiakkaiden ja muiden sidosryhmien todellisia tarpeita. Pää tavoitteita kuitenkin lähestytään näissä menetelmissä eri tavoin. Paetsch et.al (2003) mukaan vaatimusten dokumentoinnissa menetelmien välillä on isoja eroja. Seuraavissa alikappaleissa havainnoidaan ketteriä vaatimusmäärittelymenetelmiä sekä niihin liittyviä haasteita.

### 2.5.1 Ketterät vaatimusmäärittelymenetelmät ja niihin liittyvät haasteet

#### 2.5.1.1 Vuorovaikutus kehitystiimin ja asiakkaan välillä

Ketterien vaatimusmäärittelymenetelmien kulmakivenä toimii jatkuva vuorovaikutus kehitystiimin ja asiakkaan välillä. Kasvotusten käytävä keskustelu on ketterien menetelmien periaatteiden mukaisesti tehokkain tapa välittää tietoa eri sidosryhmien välillä (Agile Founda-



tion, 2015). Paetsch et.al (2003) mukaan kehitystiimin ja asiakkaan välillä ei saisi olla erillistä viestintäkerrosta. Sillitti & Succi (2016) argumentoivat, että kehitystiimin ja asiakkaan suoran vuorovaikutuksen avulla voidaan välttyä mahdollisilta väärinymmärryksiltä, joita vaatimusmäärittelyissä väistämättä esiintyy. Raskaan dokumentoinnin sijaan ketterissä vaatimusmäärittelymenetelmissä pyritään tilanteeseen, jossa vuorovaikutusten tuloksena saadaan asiakkaan tarpeiden mukaisia vaatimuksia kehitettäväksi kehitystiimille. Cohn (2004) mukaan ketterissä menetelmissä dokumentaation muotona toimii lyhyet ja informatiiviset käyttäjätarinat.

Jatkuvien muuttuvien tarpeiden vuoksi ketterissä menetelmissä nähdään raskaan dokumentaation luonti ja ylläpito turhana prosessina tapauksissa, missä voidaan saavuttaa tarpeeksi korkea suora vuorovaikutus asiakkaan ja kehitystiimin välillä (Cao & Ramesh, 2010; Sillitti & Succi, 2016). Cao & Ramesh (2010) mukaan jatkuvan vuorovaikutuksen kautta asiakkaat ovat mukana koko ohjelmistokehityksen elinkaaren aikana, jolloin mahdollisiin väärinymmärryksiin pystytään reagoimaan. Väärinymmärrysten lisäksi asiakkaiden tarpeet saattavat muuttua kehityksen elinkaaren aikana.

Hyötyjen lisäksi menetelmän soveltamisessa on myös haasteita. Cao ja Ramesh (2010) mukaan jatkuvan viestinnän tehokkuus riippuu täysin asiakkaiden ja kehittäjien välisestä vuorovaikutuksesta. Tämä lähestymistapa kuitenkin saattaa aiheuttaa riskejä kehitysprojekteissa, missä tarpeeksi intensiivistä vuorovaikutusta ei voida saavuttaa. Cao & Ramesh (2010) mukaan riskit ovat mittaavia, sillä pahimmassa tapauksessa saatetaan kehittää väärinymmärrettyjä tai puutteellisia vaatimuksia. Tämän lisäksi asiakkailla saattaa olla sidosryhmiä, joilla on projektin onnistumisen kannalta tärkeää informaatiota, mutta eivät osallistu vuorovaikutukseen (Heikkilä & Lassenius, 2015).

### **2.5.1.2 Iteratiivinen vaatimusmäärittely**

Cao & Ramesh (2010) mukaan ketterien menetelmien periaatteena on kehittää ohjelmistoa iteratiivisesti. Ketterissä menetelmissä ohjelmistokomponentit jaetaan pieniin osakokonaisuuksiin, joita julkaistaan kehitysprojektin aikana testattavaksi asiakkaille. Sillitti ja Succi

(2005) argumentoivat, että iteraatioiden avulla asiakkaat voivat antaa palautetta kehittäjille, jolloin mahdollisiin muutoksiin pystytään reagoimaan nopeammin.

Ketterissä menetelmissä pyritään ohjelmiston elinkaaren alussa tunnistamaan tärkeät ja tarvittavat korkean tason vaatimukset. Elinkaaren alussa kehitystiimille luodaan ymmärrys kehitettävän järjestelmän kriittisistä ominaisuuksista. Syy lähestymistavalla on se, että kokonaisvaltaista ja täysin veden pitävää vaatimusmäärittelyä on erittäin hankala toteuttaa. Vaatimusten korkean volatilitiitin ja jatkuvien muuttuvien tarpeiden vuoksi ketterissä menetelmissä nojataan iteratiiviseen kehitykseen kokonaisvaltaisen ennalta tehdyn vaatimusmäärittelyn sijaan. Jokaisen kehityssyklin alussa asiakkaat keskustelevat kehitystiimin kanssa kehitettävistä ominaisuuksista ja kehitettävien kohteiden prioriteeteista. Kehityssyklin alussa tapahtuvan vuorovaikutuksen kautta vaatimuksiin saadaan tarkennuksia iteraatioiden kautta. Iteraatioiden kautta tulleet yksityiskohtaiset vaatimukset varmistavat sen, että vaatimukset ovat jatkuvasti ajan tasalla, joka johtaa siihen, että vanhentuneita ja tarpeettomia vaatimuksia ei pääse valumaan kehitykseen asti. Vuorovaikutuksen kautta toteutuva vaatimusmäärittely parantaa myös asiakastytyvyyttä tapauksissa. (Cao & Ramesh, 2010; Paasivaara & Lassenius, 2006; Bjarnason et.al, 2011; Soares et.al, 2015).

Iteratiivisessa vaatimusmäärittelyssä on myös omat haasteensa. Cao & Ramesh (2010) mukaan menetelmää soveltaessa on aikataulun ja budjetin kanssa esiintyy monessa organisaatiossa haasteita. Kehityksen elinkaaren alussa havaittuihin vaatimuksiin saattaa tulla muutoksia, sillä hankkeen aikataulutus ja budjetointi nojaa alkuperäisten käyttäjätarinoiden varassa. Kehityssykliden aikana asiakkailla saattaa tulla lisää vaatimuksia tai vaatimuksiin saattaa tulla isojaakin muutoksia, jotka vaikuttavat projektin aikatauluun ja kokonaisbudjettiin (Cao & Ramesh, 2010; Bjarnason et.al, 2011). Cao & Ramesh (2010) tehdyn tutkimuksen mukaan hanketta ostavan yrityksen johdon tuen saaminen on haasteellista. Minimaalinen dokumentaatio saattaa myös aiheuttaa haasteita ja hämmennyksiä tapauksissa, missä vaatimuksiin tulee radikaaleja muutoksia tai jos esimerkiksi työntekijät vaihtuvat joko toimittajan tai asiakkaan puolella. (Cao & Ramesh, 2010; Soares et.al, 2015)

### 2.5.1.3 Dokumentointi

Lähestymistavat tavoitteisiin ovat hyvin erilaiset, vaikka ketterien ja perinteisten menetelmien päätavoitteet ovat yhtenäisiä. Cao & Ramesh (2010) mukaan dokumentoinnin määrää pyritään pitämään mahdollisimman vähäisenä ketterissä menetelmissä. Bjarnason et.al (2011) mukaan kaikkia vaatimuksia ei välttämättä edes dokumentoida. Sen sijaan tietoa vaihdetaan intensiivisten keskustelujen avulla. Ketterissä vaatimusmäärittelyissä dokumentaationa toimii yleensä käyttäjätarinat. Toisinaan dokumentaation minimaalisuus saattaa olla joissakin ympäristöissä haasteellista. Paetsch et.al (2003) mukaan dokumentaation tarkoitus on vaihtaa tietoa eri sidosryhmien välillä. Cao & Ramesh (2010) mukaan kasvotusten käytävään viestintään saattaa tulla häiriöitä tapauksissa, missä esimerkiksi kehitystiimin työntekijän vaihtuvat, tai vaatimukset muuttuvat ennakoitua monimutkaisemmiksi. Tämän lisäksi vaatimusten dokumentoinnin puute osoittautuu haasteelliseksi tapauksissa, missä vaatimuksia täytyy jäljittää. Hajautetuissa ohjelmistokehitystiimeissä dokumentoinnin puute saattaa aiheuttaa väärinymmärryksiä tai pahimmassa tapauksessa virheellisesti kehitettyjä ohjelmistokomponentteja. (Cao & Ramesh, 2010; Soares et.al, 2015). Heikkilä et.al (2015) mukaan minimaalinen dokumentaatio johtaa hiljaisen tiedon kasvamiseen, joka puolestaan kasvattaa riskejä, mikäli henkilöstöön tulee muutoksia.

## 2.6 Ketterien ja perinteisten vaatimusmäärittelymenetelmien vertailu

Sekä ketterien että perinteisten vaatimusmäärittelymenetelmien päätavoite on tuottaa vaatimuksia, jotka aidosti vastaavat sidosryhmien tarpeita. Lähestymistavoissa on kuitenkin eroja, vaikka päätavoitteet ovatkin samat. Perinteinen, eli vesiputousmallinen vaatimusmäärittely perustuu siihen, että kaikki vaatimukset tiedostetaan ennen varsinaisen kehitystyön alkua. Mahapatra & Mangalaraj (2005) mukaan vesiputousmallinen vaatimusmäärittely perustuu lineaariseen prosessiin, jossa kaikki sidosryhmien toiminnalliset ja ei-toiminnalliset vaatimukset määritellään varhaisessa vaiheessa ennen varsinaista kehitystyön aloittamista. Cao & Ramesh (2010) mukaan ketterä vaatimusmäärittelyprosessi perustuu ajatukseen, missä kehitystyön alkuvaiheissa tärkeintä on tiedostaa kehitettävän oh-

jelmiston kriittiset toiminnallisuudet. Näin ollen ketterissä menetelmissä pyritään ohjelmiston elinkaaren alussa tunnistamaan tärkeät ja tarvittavat korkean tason vaatimukset. Tarkemmat vaatimukset järjestelmälle luodaan iteraatioiden kautta kehitystyön aikana.

Robertson (2006) mukaan ohjelmistoa tuotetaan inkrementaalisesti, jossa asiakkaan sidosryhmät ovat mukana kehitysprojektin jokaisessa vaiheessa. Sommerville (2004) mukaan perinteisessä vaatimusmäärittelyssä sen sijaan luotetaan kokonaisvaltaiseen vaatimusmäärittelytyöhön ennen hankkeen alkua, jolloin asiakkaalle toimitetaan kerralla ohjelmisto testattavaksi. Ketterien menetelmien kehityssyklit ovat erillisiä projekteja, jolloin määritellään vaatimukset, kehitetään toiminnallisuuksia sekä toimitetaan asiakkaalle testattavaksi.

Petersen et.al (2009) mukaan perinteinen vaatimusmäärittely perustuu raskaan dokumentaation luomiseen ja ylläpitämiseen. Cao & Ramesh (2010) mukaan ketterissä menetelmissä pyritään minimoimaan dokumentaation määrää. Ketterissä menetelmissä raskas dokumentaatio korvataan asiakkaiden ja kehittäjien välisellä intensiivisellä vuorovaikutuksella ja yksinkertaisilla käyttäjätarinoilla. Näin ollen Schön (2018) argumentoi, että ketterien menetelmien inkrementaalisen kehityksen luonteen vuoksi mahdollisiin muutoksiin pystytään reagoimaan nopeasti. Sommerville (2004) mukaan perinteisessä vaatimusmäärittelyssä jokainen pienikin muutos käynnistää raskaan prosessin, jonka takia muutosten tekeminen on erittäin vaikeaa ja usein kasvattaa hankkeen budjettia. Perinteiset vaatimusmäärittelymenetelmät toimivat parhaiten hankkeissa, joissa vaaditaan jäykkyyttä. Esimerkiksi korkeasti säännellyillä toimialoilla vaaditaan tietynlaista jäykkyyttä, jolloin ketteriä menetelmiä ei välttämättä voitaisi sellaisenaan soveltaa. Perinteiset menetelmät sopivat myös hankkeille, joissa kaikki vaatimukset ovat tosiasiallisesti selviä eikä muutoksiin tarvitse varautua. Perinteiset menetelmät toimivat myös esimerkiksi hankkeilla, joissa on tarkkaan määritelty budjetti ja aikataulu. Ketterissä menetelmissä sen sijaan varaudutaan muutoksiin, jonka vuoksi menetelmän soveltaminen tuo joustavuutta. Cao & Ramesh (2010) mukaan iteraatioiden avulla ohjelmistokehityksen laatu kasvaa, mutta aikataulut ja budjetointi on usein haasteellista, sillä vaatimukset tarkentuvat vasta kehitystyön aikana. Soares et.al (2015) mukaan ketterissä menetelmissä vaatimusten priorisointi saattaa myös olla haasteellista tapauksissa, missä asiakkaan sidosryhmät ovat vaatimusten tarpeellisuudesta

ja priorisoinnista eri mieltä. Cao & Ramesh (2010) mukaan ketterät menetelmät toimivat parhaiten hankkeissa, jossa asiakkaiden sidosryhmät ovat jatkuvassa vuorovaikutuksessa kehittäjien kanssa. Bjarnason et.al (2011) mukaan ohjelmiston laatu heikkenee, mikäli asiakkaan sidosryhmät eivät ole sitoutuneita ohjaamaan kehitystyötä.

## 2.7 Yhteenveto

Tämän diplomityön tavoitteena on luoda yhtä yritystä ja toimialaa laajempi ymmärrys ohjelmistoliiketoiminta-alalla käytetyistä vaatimusmäärittelymenetelmistä ja niihin liittyvistä haasteista. Tämän vuoksi luvun edellisissä kappaleissa on havainnoitu kattavasti olemassa olevia tutkimuksia ja alan kirjallisuutta, jotka liittyvät erilaisiin vaatimusmäärittelymenetelmiin ja niihin liittyviin haasteisiin. Aihetta lähestytään yhden case-yrityksen kautta, joka on erikoistunut tuotteistettuun ohjelmistoliiketoimintaan ja toimii ketterien menetelmien mukaisesti. Kohdeyrityksen toiminta on täysin hajautettua, sillä kaikki työntekijät kommunikoiivat pääosin digitaalisten työkalujen kautta. Tutkimuksen pääasiallisena tavoitteena on selvittää, millaisia haasteita kohdeyrityksen vaatimusmäärittelyprosessissa esiintyy, sekä havaita mahdollisia ratkaisukeinoja, jotka pohjautuvat olemassa oleviin tutkimuksiin ja alan kirjallisuuteen. Näin ollen luvussa havainnoidut menetelmät ja haasteet ovat olennaisia tutkimuskysymysten kannalta. Tässä luvussa koostetaan yhteen edellisissä osioissa havainnoidut vaatimusmäärittelyihin liittyvät menetelmät ja haasteet.

Vaatimusmäärittelyt ovat tärkein tekijä onnistuneiden ohjelmistoprojektien kannalta. Polini (2010) suorittaman tutkimuksen mukaan noin 37 % haastavien järjestelmien kehitysvaiheen haasteista liittyy tavalla tai toisella vaatimusmäärittelyissä tehtyihin virheisiin. Vaatimusmäärittelyt voidaan jakaa sekä perinteisiin että ketteriin vaatimusmäärittelyihin. Molempien menetelmien päätavoitteena on tuottaa vaatimuksia, jotka tosiasiallisesti vastaavat loppukäyttäjien tarpeita. Tavoitteiden lähestymistavoissa on kuitenkin suuria eroja, sillä perinteisen vaatimusmäärittelyprosessin tavoitteena on saada kaikki asiakkaiden vaatimukset ennen varsinaisen kehitystyön alkua, kun taas ketterä vaatimusmäärittelyprosessin tavoitteena on tuoda asiakkaiden vaatimuksia kehitystyön aikana iteraatioiden kautta. Pe-

rintainen vaatimusmäärittelyprosessi nojautuu raskaan dokumentaation luomiseen ja ylläpitämiseen, kun taas ketterä vaatimusmäärittelyprosessi nojautuu korkealaatuiseen intensiiviseen vuorovaikutukseen asiakkaiden ja kehittäjien välillä. Alla olevassa listassa havainnoidaan ketterien vaatimusmäärittelymenetelmien hyötyjä ja haasteita:

#### 1. Vuorovaikutus kehitystiimin ja asiakkaan välillä

- + Kehitystiimin ja asiakkaiden sidosryhmien intensiivisen vuorovaikutuksen kautta minimoidaan mahdolliset väärinymmärrykset. (Sillitti & Succi, 2016)
- + Jatkuvan vuorovaikutuksen ansiosta asiakkaat ovat mukana koko elinkaaren aikana. Intensiivisen vuorovaikutuksen kautta asiakkaiden sidosryhmät ovat ohjainmassa toiminnallisuuksien vaatimuksia oikeaan suuntaan (Cao & Ramesh, 2010; Sillitti & Succi, 2016).
- + Korkealaatuisten vuorovaikutusten ansiosta dokumentaation määrää voidaan minimoida (Cao & Ramesh, 2010).
- Jatkuvan viestinnän tehokkuus riippuu vuorovaikutuksen laadusta: mittaavat riskit, kuten väärinymmärrykset vaatimukset, mikäli intensiivisestä vuorovaikutuksesta ei voida saavuttaa (Cao & Ramesh, 2010).
- Vaatimusten kannalta tärkeiden sidosryhmien poissaolo määrittelyissä (Heikkilä & Lassenius, 2015).
- Dokumentaation puute voi johtaa väärinymmärryksiin ja hankaloittaa vaatimusten jäljitettävyyttä (Cao & Ramesh, 2010; Soares et.al, 2015)
- Hiljaiseen (tacit) tietoon nojaaminen kasvattaa riskejä henkilöstön muutoksissa (Heikkilä et.al, 2015).

#### 2. Iteratiivinen vaatimusmäärittely

- + Iteraatioiden kautta asiakkaat kykenevät antamaan palautetta toiminnallisuuksista, jolloin mahdollisiin muutoksiin voidaan reagoida nopeasti. (Schön, 2018)

- + Tarkentavien vaatimusten esilletuonti iteraatioiden kautta kasvattaa ohjelmiston laatua ja asiakastytyvääisyyttä (Bjarnason et.al, 2011).
- + Kehityssyklin aikana priorisoidaan ja katselmoidaan vaatimuksia, jolloin mahdollisesti jo vanhentuneita vaatimuksia voidaan jättää kehittämättä (Cao & Ramesh, 2010).
- Iteraatioiden kautta tulleet vaatimukset hankala arvioida hankkeen kokonaiskustannuksia ja aikataulua (Cao & Ramesh, 2010).
- Hanketta ostavan johdon tuen saaminen haasteellista (Cao & Ramesh, 2010).

Perinteiset menetelmät toimivat parhaiten hankkeissa, jossa vaaditaan tietynlaista jäykkyyttä sekä ohjelmistotuotteen toiminnallisuuksien osalta. Ketterät menetelmät sopivat taas parhaiten hankkeissa, jossa asiakkaiden vaatimusten muutospyyntöihin on kyettävä reagoimaan nopealla aikataululla. Alla olevassa taulukossa havainnoidaan näiden menetelmien lähestymistapojen eroja:

Aktiveetti	Perinteinen	Ketterä (Scrum)
<b>Kartoittaminen</b>	Kaikkien vaatimusten perusteellinen selvittäminen ennen varsinaisen kehitystyön alkua (Glinz, 2007).	Kehitystyön elinkaaren alussa tiedettävä kriittiset vaatimukset. Tarkemmat vaatimukset muodostuvat iteraatioiden kautta (Cao & Ramesh, 2010).
<b>Analysointi</b>	Kartoitettujen vaatimusten, eli dokumentoitujen vaatimusten perusteellinen analyysi. Onko vaatimukset kirjattu oikealla tavalla? (Sommerville, 2004).	Kehitysjonoon dokumentoitujen vaatimusten priorisointi ja vaatimusten tarkennukset iteraatioiden kautta (Cao & Ramesh, 2010).
<b>Validointi</b>	Analysoitujen vaatimusten validointi. Vastaavatko dokumentoidut vaatimukset tosi-aasiallisesti asiakkaiden tarpeita? (Kotonya & Somerville, 1998).	Vaatimusten jatkuva katselmointi ja tuotettujen ohjelmistoinkrementtien hyväksyntätestaus. Asiakkaiden mahdolliset palautteet ja muutosehdotukset. (Cao & Ramesh, 2010)
<b>Dokumentointi</b>	Raskas dokumentaatioprosessi, jossa kaikki vaatimukset on dokumentoitava ennen kehitystyön alkua.	Intensiivinen vuorovaikutus kehittäjien ja asiakkaiden välillä. Minimaalinen dokumentointi käyttäjätarinoiden muodossa (Schön, 2017).

<b>Hallinta</b>	Varmistetaan, että mahdolliset muutokset kulkeutuvat ennalta määrätyn prosessin läpi. Vaatimusdokumentaation ylläpito jäljitettävyyden kannalta (Sommerville, 2004).	Vaatimukset dokumentoituna tuotteen kehitysjonoon käyttäjätarinoiden muodossa, jossa vaatimuksia hallitaan. Suunnittelukokouksissa suoritetaan mahdolliset muutokset vaatimuksiin. (Cao & Ramesh, 2010)
-----------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Kuva 6. Perinteisten ja ketterien vaatimusmäärittelyaktiviteettien vertailu

	Perinteinen	Ketterä
<b>Perusteet</b>	Järjestelmät ovat täysin määriteltäviä, ennakoitavia, ja ne voidaan rakentaa huolellisella ja perusteellisella suunnittelulla.	Kooltaan pienet tiimit kykenevät kehittämään laadukkaita ja käyttäjien tarpeisiin mukautuvia ohjelmistoja hyödyntämällä jatkuvan suunnittelun, kehittämisen ja testaamisen periaatteita.
<b>Hallinta</b>	Prosessipainotteinen	Ihmiskeskeinen
<b>Johtamistapa</b>	Hierarkkinen	Yhteistyöhön perustuva
<b>Tiedon hallinta</b>	Muodollinen, dokumentaatioon perustuva	Epävirallinen, intensiiviseen vuorovaikutukseen perustuva
<b>Kommunikaatio</b>	Virallinen	Epävirallinen
<b>Asiakkaan rooli</b>	Tärkeä	Kriittinen
<b>Sopiva organisaatiorenne</b>	Byrokraattinen ja hierarkkinen	Joustava, osallistumiseen kannustava ja yhteistyöhön perustuva

Kuva 7. Perinteisten ja ketterien ohjelmistokehitysmenetelmien vertailu (Nerur et.al, 2005).

Kustannussäästöjen ja kokeneiden ohjelmistokehityssammattilaisten saavuttamisen takia organisaatiot pyrkivät hyödyntämään globaalia työvoimaa, joiden takia varsinainen kehi-



tystyö hajautetaan fyysisesti eri paikkoihin (Vallon et.al, 2018). Ketterien menetelmien soveltamisessa hajautetuissa tiimeissä on haasteita, sillä ketterien menetelmien mukaisesti toimivin ja tehokkain tapa tiedon välittämiseksi on kasvokkain käytävät keskustelut (Cao & Ramesh, 2010). Hajautetuissa tiimeissä ei usein pystytä toteuttamaan korkeanlaatuista ja intensiivistä vuorovaikutusta. Tämän lisäksi haasteina ovat dokumentaation minimaalisuus yhdistettynä huonolaatuisen viestintään, joka voi pahimmillaan johtaa vaatimusten vääринymmärryksiin. Shrivastava & Date (2010) mukaan monet offshore-kehitystiimit suosivat näin ollen perinteistä lähestymistapaa, joka nojautuu dokumentoituihin vaatimuksiin. Ketteryyden säilyttämisen vuoksi dokumentaation määrä ei saa kasvaa kuitenkaan liian yksityiskohtaiseksi. Näin ollen hajautettujen ketterien ohjelmistokehitystiimit kohtaavat haasteita, jotka liittyvät kommunikointiin ja määrämuotoisten dokumentaation tasapainottamiseen.

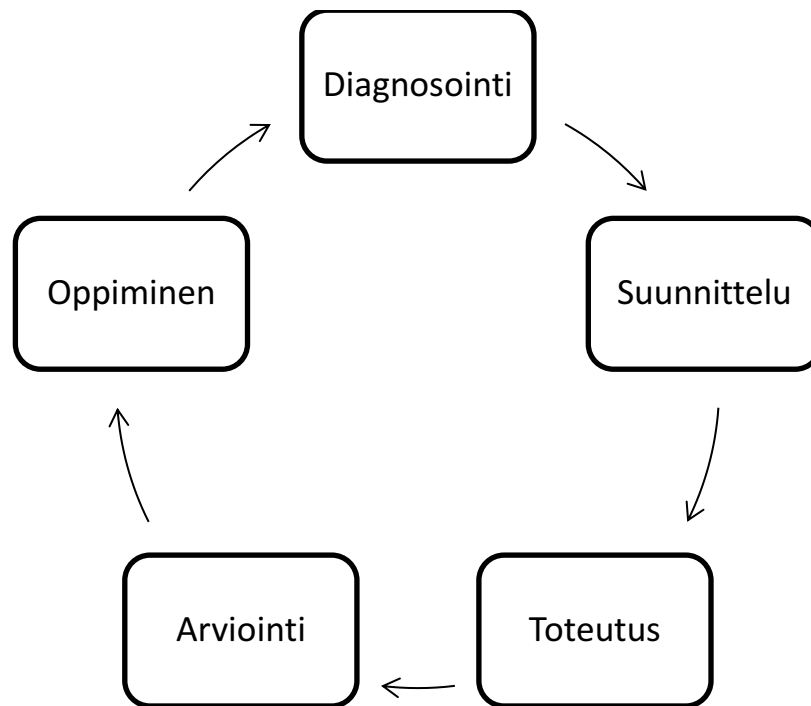
### 3 Metodologia

Tässä kappaleessa havainnoidaan tarkemmalla tasolla mitä metodologiaa ja metodeja tutkimustyössä käytetään, sekä argumentoidaan metodologiavalintojen puolesta. Tämän lisäksi tässä kappaleessa havainnoidaan tutkimuksen empiirisen osan tiedonkeruu- ja analysointimenetelmiä. Tutkimuksen ensisijainen tavoite oli selvittää nykyisen vaatimusmäärittelyprosessin haasteita haastateltavan näkökulmasta keräämällä laadullista tietoa haastattelujen avulla. Kohdeyrityksen vaatimusmäärittelyprosessin nykyisten haasteiden selvittämistä varten tiedonkeruumenetelmänä käytettiin pääosin puolistrukturoituja haastatteluja. Temaattista analyysiä käytettiin tiedon analysoinnin viitekehyksenä. Tiedonkeruumenetelmiä havainnoidaan tarkemmalla tasolla kappaleessa **3.2**. Haasteiden avulla tutkimuksen tavoitteena oli selvittää, millaisilla käytännöillä voitaisiin parantaa kohdeyrityksen vaatimusmäärittelyprosessia. Näiden edellä mainittujen asioiden takia tutkimuksessa sovelletaan tutkimusstrategiana toimintatutkimusta, jota hyödyntämällä voidaan luoda mahdollisia muuttavia toimenpiteitä tarkastelemalla ensin nykyisiä käytäntöjä, joiden perusteella pyritään muuttamaan nykyisiä toimintotapoja iteratiivisella yhteistyöprosessilla. Näin ol-

len, tutkimuksessa hyödynnetään osallistavaa toimintatutkimusta, joka luokitellaan toimintatutkimuksen alaluokaksi. Osallistavaa toimintatutkimusta hyödyntämällä pyritään synnyttämään muuttavaa toimintaa tutkijan ja kohdeyrityksen iteratiivisen yhteistyön avulla. Tutkimuksen lähestymistavat havainnoidaan tarkemmalla seuraavassa kappaleessa.

### **3.1 Tutkimuksen lähestymistapa**

Toimintatutkimuksen soveltaminen tutkimusstrategiana on perusteltua niissä tutkimuksissa, joissa pureudutaan organisaatioiden todellisiin haasteisiin, joiden perusteella pyritään löytämään ratkaisukeinoja näihin haasteisiin. Toimintatutkimus luokitellaan iteratiiviseksi prosessiksi, jossa pyritään yhdistämään teoreettinen tutkimus käytännön toimintaan. (Avison et. al, 1999; Baskerville & Wood-Harper, 1996). Haasteiden syvällisen ymmärtämisen lopputuloksena toimintatutkimuksen tavoitteena on kehittää organisaation toimintotapoja tutkijan ja kohdeyrityksen yhteistyön avulla (Argyris, Putman & Smith, Avison et. al, 1999). Toimintatutkimus koostuu syklisestä prosessista, jossa voidaan erottaa viisi erillistä ja tunnistettavaa vaihetta. Toimintatutkimusprosessin vaiheita ovat: diagnosointi, suunnittelu, toteutus, arviointi ja oppiminen (Baskerville & Harper, 1996). Näin ollen toimintatutkimuksessa pyritään iteratiivisen prosessin avulla tunnistamaan mahdollisia ongelmia, suunnittelemaan parannusehdotuksia, toimeenpanemaan suunniteltuja parannusehdotuksia sekä arvioimaan näiden lopputulosta. Alla olevassa kuvassa havainnoidaan toimintatutkimuksen syklistä prosessia:



Kuva 8. Toimintatutkimuksen vaiheet (Susman & Evered, 1978).

Toimintatutkimusprosessi alkaa mahdollisen ongelman tunnistamisella tai määrittämisellä. Tätä vaihetta kutsutaan ongelman diagnosoinniksi, jonka aikana kerätään mahdollisimman paljon tietoa tunnistetusta ongelmasta erilaisten laadullisten tai määrällisten metodien avulla, sekä analysoidaan kerättyä tietoa. Tiedonkeruumenetelmät avataan tarkemmalla tasolla kappaleessa **3.3**. Diagnostointi-vaiheen pääasiallinen tarkoitus on saada syvä ymmärrys sen hetkistä haasteista. Diagnostointi-vaiheen aikana tutkija kerää ja analysoi teoreettista kirjallisuutta, jotka liittyvät diagnostointivaiheeseen esiintyneisiin haasteisiin. Diagnostointi-vaiheen jälkeen suunnitellaan ja tarkastellaan vaihtoehtoisia toimintatapoja, jotka poikkeavat kohdeyrityksen nykyisistä käytännöistä. Tämän vaiheen aikana tutkija analysoi nykytilannetta kerättyjen tietojen avulla, sekä yrittää löytää vaihtoehtoisia ratkaisuja, jotka perustuvat tieteellisiin tutkimuksiin. Suunnitteluvaiheen jälkeen toteutetaan suunniteltu toiminta. Tämän vaiheen aikana tutkija fasilitoi yhteisen työpajan, johon osallistuu kohdeyrityksen vaatimusmäärittelyistä vastaavat henkilöt. Kaikkia työpajaan osallistuvia henkilöitä on haastateltu diagnostointivaiheen aikana. Työpajan tarkoituksena on luoda syväisiä iteraatioita, jotka liittyvät kohdeyrityksen vaatimusmäärittelyprosessin haasteisiin vaatimusmäärittelijöiden näkökulmasta. Työpajan alussa tutkija esittää muille

osallistujille löydökset sekä yhtenäiset ja yksittäiset haasteet. Esitetyt haasteet luovat pohjan yhteiselle keskustelulle, jonka tarkoituksena on luoda yhteisymmärrys työpajaan osallistuvien välillä. Iteraatioiden päätarkoituksena on löytää jokin tietty kipukohta kohdeyrityksen vaatimusmäärittelyprosessissa, johon pyritään hakemaan konkreettisia ratkaisukeinoja ja ehdotuksia tuleville toimenpiteille yhteisvoimin. Tämän jälkeen toteutettua toimintaa arvioidaan tarkastelemalla toimintatutkimuksen toteutuksen tuloksia. Arviointivaiheen jälkeen pyritään oppimaan ja havainnoimaan jatkotoimenpiteet yleisten havaintojen perusteella.

## **3.2 Case-yritys**

Tämän tutkielman empiirisessä osuudessa keskitytään kohdeyrityksen tuoteliiketoimintaosaston vaatimusmäärittelyprosessiin kehitykseen. Tässä alikappaleessa havainnoidaan kohdeyrityksen rakennetta sekä ohjelmistotuotetta.

### **3.2.1 Kohdeyrityksen kuvaus ja tuotteistettu ohjelmisto**

Tämä diplomityö suoritetaan kansainväliselle kohdeyritykselle, joka tarjoaa laajasti erilaisia digitaalisia palveluja eri toimialoilla toimiville yrityksille. Kohdeyrityksen toiveiden mukaan yrityksen nimeä ei julkaista tässä diplomityössä. Kohdeyrityksen tarjontaa kuuluvat muun muassa ohjelmistokehitys- ja konsultointipalvelut, elinkaari palvelut sekä tuotteistetut ohjelmistotuotteet. Kohdeyritys on Suomalaislähtöinen pk-yritys, jonka päätoimipiste sijaitsee Helsingissä. Kohdeyrityksessä työskentelee maailmanlaajuisesti yli viisisataa työntekijää, joista suurin osa sijaitsee Suomessa.

Tämä diplomityö suoritetaan kohdeyrityksen tuotteistetun ohjelmistoliiketoiminnan osastolle. Tuotetta kehittävä osasto toimii itsenäisesti, mutta on juridisesti osana isompaa yritystä. Kohdeyritys on kehittänyt modernin toiminnanohjausjärjestelmän, jota asiakkaat käyttävät materiaalien, kunnossapidon, työntekijöiden ja palveluiden hallintaan. Kohdeyrityksen asiakkaat toimivat pääsääntöisesti kunnossapidon tai elintarvike- ja energiateollisuuden liiketoiminta-alalla. Tämän lisäksi kohdeyrityksen asiakkaisiin kuuluvat myös eri

aloilla toimivat tuotantolaitokset. Kohdeyrityksen kehittämä tuote on jo monella yrityksellä menestyksekkäästi tuotantokäytössä, mutta vaatimusmäärittelyiden prosessissa kohdeyritys toivoo vankempaa määrämuotoisuutta. Kohdeyrityksen tavoitteena on yhtenäistää ohjelmistojen vaatimusmäärittelyprosessia selvittämällä nykyiset haasteet vaatimusmäärittelijöiden näkökulmasta sekä ottamalla käyttöön uusia, määrämuotoisia ja hyväksi havaittuja vaatimusmäärittelykäytäntöjä. Tämän diplomityön laatija on työskennellyt kohdeyrityksessä yli kolme vuotta ja ollut mukana monessa eri asiakasprojektissa konsultin roolissa. Näin ollen diplomityön aihe syntyi tutkijan suorittamien havaintojen perusteella. Havaintojen perusteella kohdeyrityksen ylemmän johdon kanssa käytiin iteratiivista keskustelua, jonka tuloksena syntyi fokusoitu aihe tälle diplomityölle.

Asiakkaiden liiketoiminta-alojen monimuotoisuuden vuoksi kohdeyrityksellä on ollut haasteita heidän vaatimusmäärittelyprosessissansa. Tämän lisäksi vaatimusmäärittelyistä vastaavilla henkilöillä on toisistaan poikkeavat käytännöt vaatimusten hallintaan ja määrittelyihin liittyvissä tehtävissä. Vaatimuksista vastaavat henkilöt työskentelevät asiakasprojekteissa projektipäällikköinä, mutta näiden henkilöiden osaamisalueet ja taustat poikkeavat toisistaan. Osalla on muutaman vuosikymmenen ajan kertynyt vahva liiketoiminnan substanssiosaaminen, kun taas osalla vahvuudet piilevät teknisten ohjelmistoprojektien läpiviennissä. Taustasta ja osaamisesta johtuen näillä henkilöillä on omat, toisistaan poikkeavat tavat käsitellä ja dokumentoida asiakkaiden vaatimuksia. Näiden edellä mainittujen tekijöiden takia kohdeyrityksellä on ollut haasteita vaatimusten läpiviennissä ja hallinnassa.

Kohdeyrityksessä noudetaan pääsääntöisesti ketterän ohjelmistokehityksen periaatteita. Scrum, Kanban ja Scrumban menetelmiä sovelletaan eri projekteissa ja kehitystöissä laajasti. Näiden edellä mainittujen menetelmien lisäksi kohdeyrityksessä sovelletaan myös joiltakin osin vesiputousmallisia projektinhallintamenetelmiä. Tuotetta sen sijaan kehitetään puhtaasti ketterien menetelmiä, erityisesti Scrum-viitekehystä hyödyntäen. Projektipäälliköistä riippuen toimitusprojekteissa käytetään erilaisia menetelmiä. Kohdeyrityksen tavoitteena on yhtenäistää ohjelmiston vaatimusmäärittelyprosessia ensinnäkin selvittämällä nykyiset haasteet vaatimusmäärittelijöiden näkökulmasta sekä ottaa käyttöön uusia, määrämuotoisia ja hyväksi havaittuja vaatimusmäärittelykäytäntöjä.

Kohdeyrityksen tuotteen toimitusprojekteissa toivotaan ja vaaditaan enemmän jäykkyyttä. Tämän siksi, koska kohdeyrityksen asiakkaat toimivat usein hyvin säännellyillä toimialoilla. Tämän lisäksi toimitusprojekteissa on useimmiten määritelty sekä tarkka budjetti että aikataulu projektin toimittamiselle. Näin ollen puhtaasti ketterien menetelmien käyttö ei sovellus kohdeyrityksessä näiden edellä mainittujen tekijöiden vuoksi. Toisaalta kohdeyrityksessä ei voida soveltaa vesiputousmallisia menetelmiä sellaisenaan, sillä joidenkin asiakkaiden muuttuvien tarpeiden vuoksi toimitusprojekteissa on oltava mahdollisuus myös muutoksiin nopealla aikataululla. Kohdeyrityksellä on haasteita löytää sopivia vaatimusmäärittelymenetelmiä, sillä projekteissa tarvitaan menetelmiä, joissa on tarpeeksi jäykkyyttä ja määrämuotoisuutta, mutta kuitenkin sillä ehdolla, että vaatimuksia pystytään tietyiltä osin muuttamaan ilman raskasta muutosprojektia.

### **3.2.2 Henkilöstön rakenne**

Kohdeyrityksellä on työntekijöitä hajautetusti eri toimipisteissä. Hajautettu työympäristö tuo jo itsessään monia, pääosin kommunikaatioon liittyviä haasteita. Vielä haasteellisempaa on ketterien menetelmien soveltaminen näin hajautetussa tiimissä. Alla olevassa listassa havainnoidaan kohdeyrityksen eri toimipisteitä:

1. Tampereen toimipisteellä olevat henkilöt
2. Helsingin toimipisteellä olevat henkilöt
3. Etätyötä tekevät henkilöt
4. Ulkoistetut yksittäiset ulkomailla sijaitsevat toimijat

Tampereen toimipisteellä työskentelevät pääosin liiketoiminnan substanssiasiantuntijat. Näiden substanssiasiantuntijoiden lisäksi kohdeyrityksen toimitusjohtajan ja innovaatiopäällikön työpiste sijaitsee Tampereella. Helsingin toimipisteen työntekijät koostuvat pääosin teknisistä konsulteista, ohjelmistokehittäjistä sekä teknisistä projektipäälliköistä. Tämän lisäksi kohdeyrityksen työntekijät työskentelevät usein myös etänä. Hajautetusti ympäri maailmaa sen sijaan sijaitsevat tuotteen ulkoistetut yksittäiset ohjelmistokehittäjät.

Näin ollen tuotteen ohjelmistokehitystiimi muodostuu ulkomaalaista, yksittäisistä toimijoista. Kohdeyrityksen työntekijät kommunikoivat toistensa kanssa useimmiten videokousten ja viestintäsovellusten kautta. Vallitsevan koronatilanteen vuoksi kohdeyrityksen työntekijät ovat jo yli vuoden verran kommunikoineet toistensa kanssa näiden erilaisten sovellusten välityksellä.

### 3.3 Tiedonkeruumenetelmät

Tutkielman tiedonkeruumenetelmän-osiossa havainnoidaan käytettyjä tiedonkeruumenetelmiä. Toimintatutkimuksessa voidaan käyttää sekä laadullisia että määrällisiä tiedonkeruumenetelmiä. Menetelminä voidaan käyttää havaintoja, haastatteluja, kyselylomakkeita, muistiinpanoja tai kohderyhmiä (Gillis & Jackson, 2002; McNiff & Whitehead, 2006). Yin (2018) mukaan haastatteluita pidetään yhtenä tärkeimpänä tiedonkeruumenetelmänä laadullisissa tutkimuksissa. Tuomi & Sarajärvi (2018) argumentoivat, että haastattelujen avulla minimoidaan mahdolliset väärinymmärrykset. Tiedonkeruuta varten valittu menetelmä riippuu täysin tutkittavan asian kontekstista. Valintaa varten tutkijan on pohdittava, minkä tyyppistä tietoa halutaan kerätä, ja minkä tyyppiin tutkimuskysymyksiin halutaan vastata. Tutkimuksen ensisijainen tavoite on saada syvälinen ymmärrys haasteista, joita esiintyy kohdeyrityksen nykyisessä vaatimusmäärittelyprosessissa haastateltavien näkökulmasta. Näin ollen, tutkimuksessa sovelletaan laadullista lähestymistapaa. Tutkielmassa tutkimuskysymyksiin vastataan käyttäen laadullisia tiedonkeruumenetelmiä.

Tässä tutkielmassa käytettiin pääosin puolistrukturoituja haastatteluja tiedonkeruunmetodeina. Puolistrukturoitujen haastatteluiden lisäksi tiedonkeruunmetodeina käytettiin myös tutkijan suorittamia havaintoja, epävirallisia keskusteluja, nykyisiä vaatimusmäärittelydokumentteja, muistiinpanoja, sekä täydentäviä, epämuodollisia lisäkysymyksiä. Puolistrukturoitujen haastatteluiden tavoite on löytää mahdollisia yhteisiä tai yksittäisiä haasteita, joita vaatimusmäärittelyistä vastaavat henkilöt kohtaavat vaatimusmäärittelyprosessin aikana. Laadullinen lähestymistapa sekä tutkimustyössä käytetyt tiedonkeruunmetodit

mahdollistavat tutkijan saamaan syvällisen ymmärryksen siitä, että mitä vaatimusmäärittelykäytäntöjä kohdeyrityksessä käytetään sekä ymmärryksen siitä, mitä haasteita kohdeyrityksen nykyisessä vaatimusmäärittelyprosessissa esiintyy.

### **3.3.1 Diagnosointivaiheen tiedonkeruumenetelmät**

#### **3.3.1.1 Puolistrukturoidut haastattelut**

Tutkielmassa valittiin haastattelumetodiksi puolistrukturoidut haastattelut ensimmäisen tutkimuskysymyksen luonteen vuoksi. Ensimmäisessä tutkimuskysymyksessä pyritään selvittämään, minkälaisia haasteita kohdeyrityksen nykyisessä vaatimusmäärittelyprosessissa esiintyy haastateltavien näkökulmasta. Haastateltaviksi valittiin ne henkilöt, jotka ovat vastuussa kohdeyrityksen vaatimusmäärittelyprosessista. Vaatimusmäärittelyistä vastuulla olevilla tarkoitetaan henkilöitä, jotka ovat olleet osallisena vaatimusten prosessoinnissa, eli esilletuonnissa, analysoinnissa, dokumentoinnissa, priorisoimisessa tai hallinnassa. Osa haastateltavista on ollut mukana kaikissa vaiheissa, kun taas henkilöistä vain tietyissä vaiheissa. Haastattelut suoritettiin vuoden 2020 lokakuun aikana. Haastateltavat henkilöt työskentelevät asiakasprojekteissa projektipäällikköinä, mutta näiden haastateltavien kompetenssit ja taustat poikkeavat toisistaan. Haastateltavien tausta on näin ollen hyvin heterogeeninen. Projektipäälliköiden lisäksi haastateltiin myös tuotekehityksen Scrum-Maasteria. Scrum-Master valittiin myös haastateltavaksi siitä syystä, koska kaikki projektipäälliköiden määrittelemät vaatimukset valuvat Scrum-Tiimille Scrum-Masterin kautta kehitettäväksi. Tällä tavoin saatiin kokonaisvaltainen ymmärrys kohdeyrityksen nykytilasta sekä vaatimusten laatijoilta että vaatimusten kehittäjiltä. Alla olevassa taulukossa havainnoidaan haasteltavia henkilöitä:



Työtehtävä	Työuran kesto yrityksessä (v)	Kokemus Vaatusmäärittelyistä (v)
Projektipäällikkö 1	3,5	20
Projektipäällikkö 2	4	18
Projektipäällikkö 3	1	18
Projektipäällikkö 4	3,5	4
Projektipäällikkö 5	5	5
Projektipäällikkö 6	4,5	18
Projektipäällikkö 7	3,5	4
Scrum Master	4	12

Kuva 9. Haastateltavat henkilöt

Tutkielmassa suoritettu puolistrukturoitu haastattelu jaettiin kahteen osaan. Ensimmäisessä osassa kaikilta haastateltavilta kysyttiin yleiset kysymykset, jotka liittyivät haastateltavien taustaan. Haastattelut suoritettiin yksitellen henkilökohtaisesti etäpalaverissa, joka nauhoitettiin kokonaisuudessaan. Tutkija ei kirjoittanut haastattelun aikana muistiinpanoja, sillä tutkijan tavoite oli keskittyä haastatteluun muodostamalla täydentäviä, lisäkysymyksiä, jonka avulla vaatimusmäärittelyihin liittyvät haasteet saatiin esille. Näin ollen nauhoitteet kirjoitettiin puhtaaksi sanasanalta, joiden avulla tietoja analysoitiin. Etäpalaveri jouduttiin järjestämään Suomen valtion sen hetkisten etätyösuositusten mukaisesti vallitsevan koronavirus tilanteen vuoksi. Haastatteluihin varattiin tunti aikaa, mutta aikaa kului haastateltavan mukaan viidestäkymmenestä minuutista puoleentoista tuntiin.

Ensimmäisen osa tavoitteena oli saada perusteellinen ymmärrys haasteltavien aikaisemmista ja nykyisistä työtehtävistä. Näiden kysymysten avulla tutkija sai käsityksen siitä, minkä tyyppisissä vaatimusmäärittelyihin liittyvissä tehtävissä haastateltavat olivat olleet. Haastattelujen ensimmäisen vaiheen kysymykset kysyttiin samassa järjestyksessä kaikilta haastateltavilta. Haastattelujen toisen vaiheen kysymykset liittyivät haastateltavien nykykäsitykseen kohdeyrityksen vaatimusmäärittelyprosessista. Haasteltavien taustojen poikkeavaisuudesta lisäkysymykset esitettiin eri järjestyksessä, sillä tutkijan tavoite oli luoda

vapaata keskustelua, jolla yritettiin selvittää haastateltavien tavat käsitellä vaatimusmäärittelyjä, sekä selvittää niihin liittyviä haasteita. Toisen vaiheen haastattelukysymykset rakentuivat kahden pääkysymyksen ympärille, joita havainnoidaan alla olevassa listassa:

1) **Minkä tyyppisissä vaatimusmäärittelyihin liittyvissä tehtävissä haastateltavat ovat olleet?**

Tämän vaiheen tavoite oli selvittää nykyisen vaatimusmäärittelyprosessiin liittyviä haasteita haastateltavan näkökulmasta.

2) **Mitä vaiheita ja käytäntöjä vaatimusmäärittelyprosessi sisältää haastateltavan kohdalla?**

Tämän vaiheen tavoite oli selvittää, millaisia vaatimusmäärittelykäytäntöjä kyseisen haastateltava soveltaa tällä hetkellä.

Toisen vaiheen ensimmäisen kysymyksen tarkoitus on saada perusteellinen käsitys siitä, että minkä tyyppisissä vaatimusmäärittelytehtävissä sekä ja rooleissa haastateltavat ovat olleet työuransa aikana. Haastateltavilta pyydettiin omin sanoin kertomaan aikaisemmat ja nykyiset tehtävät, jotka liittyivät vaatimusmäärittelyihin. Tarpeen tullen tutkija ohjasi keskustelua ja esitti lisäkysymyksiä, jotta nämä haastateltavan edelliset ja nykyiset vaatimusmäärittelyihin liittyvät tehtävät ja roolit tulivat tarpeeksi hyvin selville. Näiden tietojen avulla tutkija ymmärtää haastateltavien taustat ja pystyy luomaan haastattelun aikana mahdollisia lisäkysymyksiä paremmin. Toisen kysymyksen tarkoituksena oli luoda vapaata keskustelua tutkijan ja haastattelijan välillä. Tavoitteena oli ymmärtää, mitä vaatimusmäärittelykäytäntöjä haastateltavan vaatimusmäärittelyprosessi sisältää, sekä tunnistaa nykyisen prosessin haasteita haastateltavan näkökulmasta. Tutkija pyysi haastateltavia kertomaan omin sanoin, mitä vaiheita ja käytäntöjä vaatimusmäärittelyprosessi heidän kohdallaan sisältää siitä lähtien, kun vaatimus tulee esille ja siihen asti, kun vaatimus on toteutettu tuotteeseen. Tutkija muodosti haastattelun aikana lisäkysymyksiä, joiden tarkoituksena oli saada kokonaisvaltainen kuva vaatimusmäärittelyprosessista haastateltavan näkökulmasta. Lisäkysymykset määrittyivät pitkälti haastateltavan mukaan, sillä jotkut haastateltavista kertoivat perusteellisesti kaikki prosessin vaiheet, kun taas joiltakin tutkija joutui

kysymään todella yksityiskohtaisia asioita, jotka liittyivät vaatimusmäärittelyvaiheisiin. Tämän vaiheen tavoite oli selvittää vaatimusmäärittelyprosessiin liittyviä haasteita haastateltavan näkökulmasta. Ennen varsinaisia haastatteluja tutkijalla oli oma yleiskäsitys kohdeyrityksen vaatimusmäärittelyhaasteista. Vastausten luotettavuuden varmistamista varten tutkija ei tuonut haastatteluissa esille haasteita, jotka olivat valideja tutkijan mielestä. Tutkija ei myöskään maininnut muiden haastattelujen aikana esiin nousseita haasteita. Näin ollen haastatteluissa haastateltavat nostivat haasteita tietämättä mitä muut haasteltavat olivat haastatteluissa vastanneet. Näillä edellä mainituilla tavoilla tutkija sai selvitettyä yhtenäiset ja yksittäiset vaatimusmäärittelyihin liittyvät haasteet.

### **3.3.2 Toimintavaiheen tiedonkeruumenetelmät**

Toimintatutkimuksen kolmas, eli toteutus ja interventio suoritettiin järjestelmällä työpaja tutkijan ja kohdeyrityksen valittujen työntekijöiden kesken. Diagnosointivaiheen aikana nousseiden haasteiden perusteella tutkielmassa päädyttiin iteratiiviseen työpajaan, jonka tarkoituksena oli esittää kohdeyrityksen nykyiset vaatimusmäärittelyihin liittyvät haasteet, sekä löytämään mahdollisia ratkaisukeinoja yhteistyön voimin tutkijan ja osallistujien välillä. Diagnosointivaiheessa löydetyissä haasteissa oli paljon päällekkäisyyksiä, joka vaikutti myös tutkijan päätökseen järjestää iteratiivinen työpaja, johon osallistui henkilöt, jotka ovat vastuussa kohdeyrityksen vaatimusmäärittelyjen toteutuksesta ja hallinnasta.

Kaiken kaikkiaan iteratiiviseen työpajaan osallistui tutkijan lisäksi seitsemän henkilöä. Työpajaan osallistuivat samat henkilöt, joita haastateltiin diagnosointivaiheen aikana. Työpajaan osallistuvilla henkilöillä on toisistaan poikkeavat käytännöt vaatimusten hallintaan ja määrittelyihin liittyvissä tehtävissä. Kaikki osallistujat lukuun ottamatta tutkija työskentelevät pääsääntöisesti asiakasprojekteissa teknisinä projektipäällikköinä, mutta näiden henkilöiden osaamisalueet ja näin ollen vastuut poikkeavat toisistaan. Osa henkilöistä työskentelee tiiviisti tuotekehityksen parissa, kun taas osa työskentelee pääsääntöisesti hallinnollisina tai teknisinä projektipäällikköinä. Osallistujajoukon heterogeenisuus mahdollisti hyvinkin rikkaan ja tuottoisan työpajan, jonka tulokset avataan kappaleessa 4.

Työpajan pääasiallinen tarkoitus oli luoda rikasta keskustelua nykyisistä vaatimusmäärittelyhaasteista osallistujajoukon välillä, sekä tunnistaa kipukohtia, joihin haettaisiin vaihtoehtoisia ratkaisutapoja. Työpaja kesti yhteensä kolme tuntia ja se oli jaoteltu kolmeen eri osaan, joita havainnoidaan alla olevassa listassa:

- Ensimmäisessä vaiheessa tutkija listasi haasteita, joita esiintyi diagnosointivaiheen aikana. Näistä esiin tulleista haasteista keskusteltiin yhdessä työpajaan osallistuneiden jäsenien kesken.
- Toisessa vaiheessa tutkija esitti kohdeyrityksen nykyisen vaatimusmäärittelyprosessin vuokaaviona, jota käytiin yhdessä läpi. Vuokaavio piirrettiin haastatteluista saatujen tietojen avulla.
- Kolmannessa vaiheessa havainnoitiin vaihtoehtoisia ratkaisuja ja tulevia toimenpiteitä

Ensimmäisessä vaiheen tavoitteena oli luoda rikasta keskustelua haasteista ja käytännöistä esitettyjen haasteiden perusteella. Toisen vaiheen tarkoituksena oli esittää osallistujille kohdeyrityksen nykyiset käytännöt prosessikaaviona, ja keskustella haasteista osallistujien näkökulmasta, jotka peilasivat nykyiseen prosessiin. Tämä vaiheen tavoitteena oli havaita prosessissa tiettyjä kipukohtia iteraatioiden kautta, jotka synnyttivät runsasta keskustelua. Kolmannen vaiheen tarkoitus oli havainnoida vaihtoehtoisia ratkaisuja ja tulevia toimenpiteitä. Toimintavaiheen työpajan pääasiallisina tiedonkeruumenetelminä käytettiin tutkijan suorittamia havaintoja ja työpajan aikana otettuja muistiinpanoja. Työpaja pidettiin etäpalaverina vallitsevan koronatilanteen takia. Työpaja nauhoitettiin kokonaisuudessaan, jotta tutkijalla olisi mahdollisuus palata kuuntelemaan nauhoitteita tiedon analysointia varten.

### **3.4 Analysointimenetelmät**

Tässä kappaleessa havainnoidaan tutkimustyössä käytettyjä tiedon analysointimenetelmiä. Analysointimenetelmien lisäksi tässä kappaleessa havainnoidaan tutkimuksessa käytettyjä muita materiaaleja.

### 3.4.1 Diagnosointivaiheen tiedon analysointi

#### 3.4.1.1 Haastattelujen temaattinen analyysi

Tässä tutkimuksessa diagnosointivaiheessa kerättyyn tiedon analysointiin sovellettiin temaattista tiedonanalysointimetodia. Temaattinen analyysi on yleisesti käytetty menetelmä laadullisissa tutkimuksissa. Temaattisessa analyysissä tähdätään tiedosta löydettyjen teemojen ja tietomallijoukkojen tutkimiseen. Teemojen alle kerätään tärkeitä tietoja kerätystä tiedosta, joilla pyritään vastaamaan tutkimuskysymyksiin. Temaattisessa analyysissä järjestetään, ryhmitellään ja kuvataan aineistosta havaittuja löydöksiä. (Braun & Clarke, 2006; Turunen & Bondas, 2013). Alla olevassa kuvassa havainnoidaan temaattisen analyysin prosessia:



Kuva 10. Temaattisen analyysin prosessi (Braun & Clarke, 2006).

Temaattisen analyysin prosessi aloitetaan tutustumalla kerättyyn aineistoon. Tutkimuksessa haastattelut nauhoitettiin, jonka jälkeen nauhoitteet kirjoitettiin puhtaaksi sana-sanalta. Tässä vaiheessa tutkija jäsenteli kerätyn tiedon, jonka perusteella tutkija aloitti aineiston koodaamisen. Laadullisen analyysin kontekstissa koodilla viitataan joko kokonaiseen lauseeseen tai yksittäiseen sanaan, jotka auttavat tutkimuskysymyksen selvityksessä. Tässä vaiheessa tutkija perehtyi systemaattisesti haastattelumateriaaleihin ja korosti lauseita tai sanoja lihavoimalla tekstiä. Tutkija korosti tekstistä sellaisia lauseita tai sanoja, jotka liittyivät tutkimuskysymysten selvittämiseen. Seuraavassa vaiheessa tutkija loi kategorioita, jotka perustuivat ensimmäisen vaiheen löydöksiin. Tässä vaiheessa tutkija luki haastattelumateriaalit uudestaan läpi, jonka perusteella lihavoidut lauseet korostettiin tietyllä värillä, joka perustui siihen, mihin kategoriaan koodi kuului. Tätä vaihetta kutsutaan koodien ryhmittelyksi tai koodien kategorisoinniksi. Koodien ryhmittelyn jälkeen tutkija loi alustavia teemoja, joiden alle kerättiin aineistosta kerättyjä koodeja. Tämän alustavan tee-

makokoelman jälkeen tutkija tarkasteli kriittisesti alustavia teemoja ja niihin liittyviä koodia. Koodien merkitysten suhdetta tarkasteltiin alustaviin teemoihin, jonka tuloksena osa teemoista poistettiin kokonaan ja näiden koodit siirrettiin toisten teemojen alle. Tämän prosessin aika teemojen nimiä myös vaihdettiin. Temaattisen analyysin prosessi on iteratiivista, jossa koodia ja teemoja tarkennetaan jatkuvasti niin kauan, jotta ne vastaavat mahdollisimman hyvin tutkimuskysymyksiin.

#### **3.4.1.2 Diagnosointivaiheen muut materiaalit**

Tässä tutkimuksessa käytettiin kohdeyrityksen vaatimusmäärittelyprosessien nykytilan selvitykseen pääosin puolistrukturoituja haastatteluja. Haastatteluiden lisäksi tutkimuksessa käytettiin myös muita tiedonkeruumenetelmiä, jotka on havainnoitu alla olevassa listassa:

- Tutkijan suorittamat havainnot
- Epäviralliset keskustelut
- Nykyiset vaatimusmäärittelydokumentaatiot
- Muistiinpanot

Nämä edellä mainitut aineistot kerättiin ennen varsinaista tutkimustyön aloitusta. Tutkija havaitsi ollessaan töissä kohdeyrityksessä, että kohdeyrityksellä on tiettyjä haasteita vaatimusmäärittelykäytännöissä. Näin ollen tutkijan suorittamien havaintojen perusteella syntyi aihe koko tutkimukselle. Epävirallisilla keskusteluilla viitataan keskusteluihin, joita on käyty sekä sisäisesti kollegoiden välillä. Näiden keskustelujen aihe liittyi nykyisen vaatimusmäärittelyprosessin haasteisiin ja mahdollisiin kehityskohteisiin. Tutkija on myös perehtynyt perusteellisesti kohdeyrityksen nykyisiin vaatimusmäärittelymateriaaleihin. Materiaaleilla viitataan eri toimihenkilöiden vaatimusmäärittelydokumentaatioihin. Näin ollen, nämä edellä mainitut tiedonkeruumenetelmät ovat toimineet esitietona varsinaiselle tutkimustyön aiheen muodostamiselle.

. Puolistrukturoitujen haastatteluiden lisäksi tiedonkeruunmetodeina käytettiin myös tutkijan suorittamia havaintoja, epävirallisia keskusteluita, nykyisiä vaatimusmäärittelydokumentteja, muistiinpanoja, sekä täydentäviä, epämuodollisia lisäkysymyksiä

### 3.4.2 Toiminnan tiedon analysointi

Toimintavaiheen työpajan pääasiallisina tiedonkeruumenetelminä käytettiin tutkijan suorittamia havaintoja ja muistiinpanoja. Analyysin aikana tutkija kuunteli työpajan nauhoitteet kahteen kertaan, jonka aikana tutkija täydensi muistiinpanojaan ja validoi työpajassa tehdyt havainnot. Tämän jälkeen muistiinpanot organisoitiin analyysiä varten jakamalla muistiinpanot pienempiin osakokonaisuuksiin. Nämä osakokonaisuudet jaettiin esiin nousseiden haasteiden mukaisesti, joiden avulla tehtiin johtopäätökset.

## 3.5 Tutkimusetiikka

Tässä tutkimuksessa pyritään havainnoimaan kohdeyrityksen nykyiset vaatimusmäärittelyhaasteet haastateltavien näkökulmasta (TK1), sekä löytämään kipukohtia nykyisestä prosessista iteraatioiden kautta, johon pyritään hakemaan vaihtoehtoisia ratkaisuja. Toimintatutkimuksen iteratiivisen luonteen vuoksi tutkimuksessa käytetään laadullisia tiedonkeruumenetelmiä, kuten puolistrukturoituja haastatteluja, jonka vuoksi tässä kappaleessa havainnoidaan tutkimuksen etiikkaan liittyviä tekijöitä. Avison et.al (1999) mukaan toimintatutkimuksessa pyritään yhdistämään teoreettinen tutkimus käytännön toimintaan. Bryman & Bell (2018) mukaan haastateltavien henkilöiden yksityisyyden suojaaminen on yksi merkittävimmistä asioista, joka täytyy ottaa huomioon. Orb et.al (2000) jokaisessa tutkimuksessa on tekijöitä, joissa täytyy pohtia eettisiä kysymyksiä. Orb et.al (2000) mukaan tutkijan ja tutkimukseen osallistuvien ihmisten väliseen suhteeseen liittyy eettisiä ongelmia. Tutkimuksessa tutkijan täytyy pitää huoli siitä, että tutkimustulokset esitetään totuudenmukaisesti ilman puolueellisuutta. Tämän lisäksi tutkimuksessa on huolehdittava ihmisarvon ja itsemääräämisoikeuden kunnioituksesta (Pelkonen & Louhiala, 2002).

Tutkija selitti perusteellisesti haastateltaville henkilöille tutkimuksen sisällön, tarkoituksen ja tavoitteet ennen varsinaisia haastattelukutsuja. Tällä tavalla tutkija varmisti sen, että organisaation valitut henkilöt olivat tietoisia haastattelujen pääteemasta ja tarkoituksenperästä. Tämän lisäksi tutkija pyysi haastateltavilta suostumuksen nauhoittaa keskustelut tiedon analysointia varten. Tutkimuksen diagnosointivaiheessa tutkija haastatteli yksitellen kohdeyrityksen vaatimusmäärittelyprosessista vastaavia henkilöitä. Puolistrukturoitujen

haastattelujen tarkoituksena oli saada perusteellinen ymmärrys, millaisia haasteita yrityksen vaatimusmäärittelyprosessissa esiintyy haastateltavien näkökulmasta. Tämän vuoksi tutkimuksessa käsiteltiin asioita, jotka liittyivät osittain haastateltavien omiin haasteisiin. Haastateltavien omien haasteiden lisäksi haastateltavat mainitsivat myös heidän havaitsemia kollegoidensa haasteita, jonka vuoksi tutkimuksissa käsitellään erityisen herkkiä aiheita. Tutkija on vastuussa kerätyn aineiston yksityisyydensuojaamisesta. Tutkimuksen tuloksissa anonymisoitiin mahdollisimman hyvin haastateltavat henkilöt. Diagnosointivaiheen tuloksissa käytetyt suorat lainaukset anonymisoitiin täysin yksityisyyden suojaamista varten. Kaikkia tutkimukseen osallistuneita henkilöitä kutsutaan ”projektipäälliköiksi”. Tuotekehityksen Scrum-Masterin suoria lainauksia ei pystytty tässä tutkimuksessa anonymisoida, sillä samassa roolissa ei työskentele muita henkilöitä. Sen sijaan tutkija haki suostumuksen tulosten esittämiseen Scrum-Master:ilta. Kohdeyrityksen yksityisyyden suojaamista varten tässä diplomityössä ei julkaista kohdeyrityksen nimeä. Tutkijan työuran vaikutusta tutkimustuloksiin ja niiden luotettavuuteen havainnoidaan tarkemmalla tasolla kappaleessa 3.2 ja 5.5

## 4 Tulokset

Tässä kappaleessa havainnoidaan tutkimuksen empiirisen osan tuloksia. Tulokset perustuvat alla oleviin tutkimuskysymyksiin, joita havainnoidaan alla olevassa listassa:

**TK1.** Minkälaisia haasteita esiintyy kohdeyrityksen nykyisessä vaatimusmäärittelyprosesseissa vaatimusmäärittelijöiden näkökulmasta?

**TK2.** Millaisilla käytännöillä voitaisiin parantaa kohdeyrityksen tuoteliiketoiminnan hajautettujen ohjelmistokehityshankkeiden vaatimusmäärittelyprosessia?

Kappaleessa 4.1 vastataan ensimmäiseen tutkimuskysymykseen, jonka vastaukset saatiin puolistrukturoitujen haastatteluiden avulla sekä laadullisen datan analysoinnilla. Haastateluiden kerätyn tiedon perusteella kappaleessa 4.1 havainnoidaan kohdeyrityksen nykyistä



vaatimusmäärittelyprosessia sekä esitetään, minkälaisia haasteita nykyisessä vaatimusmäärittelyprosessissa esiintyy. Kappaleessa 4.2 vastataan toiseen tutkimuskysymykseen havainnoimalla toimintatutkimuksen toteutuksen tuloksia. Toimintatutkimuksen toteutusvaiheessa pidettiin iteratiivinen työpaja tutkijan ja haastateltavien välillä. Tutkimuksen viidennessä luvussa havainnoidaan tutkimuksen yleistettävyyttä peilaamalla tutkimuksen empiirisen osan tuloksia olemassa oleviin tutkimuksiin.

## **4.1 Diagnosointivaiheen tulokset**

Seuraavassa alikappaleissa havainnoidaan kohdeyrityksen nykyistä vaatimusmäärittelyprosessia sekä nykyisessä prosessissa esiintyviä haasteita. Ennen kuin havainnoidaan kohdeyrityksen vaatimusmäärittelyprosessissa esiintyvät haasteet, on suotavaa esittää kohdeyrityksen nykyinen vaatimusmäärittelyprosessi. Nykyisen prosessin havainnollistamisella luodaan ymmärrys kohdeyrityksen tämänhetkisistä käytännöistä.

### **4.1.1 Nykyinen vaatimusmäärittelyprosessi**

Nykyisen vaatimusmäärittelyprosessi havainnollisestettiin haastatteluista kerättyjen tietojen avulla. Vaatimusmäärittelyprosessi kuvattiin prosessikaavioksi, jonka toteutukseen käytettiin Microsoft Visio -työkalua. Prosessikaavion yläreunassa havainnoidaan aktiviteetteja, kun taas vasemmalla sivureunassa havainnoidaan ryhmiä, jotka ovat vaatimusmäärittelyprosessissa mukana. Prosessikaavio löytyy diplomityön liitteestä 1. Alla olevassa listassa havainnoidaan vaatimusmäärittelyihin liittyviä aktiviteetteja ja vastuuryhmiä:

#### **Aktiviteetit**

- Vaatimusten esilletuonti
- Spesifikaatiot
- Työmääräarviot
- Kehitystyö
- Testaus

#### **Vastuuryhmät**

- Asiakkaan edustajat
- Toimitusprojektin henkilöstö
- Tuotteenhallinta
- Tuotekehitystiimi (Scrum-tiimi)

Kohdeyrityksen kontekstissa vaatimusmäärittelyt aloitetaan uuden asiakkuuden kohdalla toimitusprojektista. Toimitusprojekti on jaoteltu erillisiin osakokonaisuuksiin. Ohjelmiston toimitusprojektissa koostuu vaatimusmäärittelyistä, konsultoinnista, määrittelyistä ja toteutustöistä. Tämän lisäksi toimitusprojektin aikana kohdeyritys opastaa ja kouluttaa asiakasta käyttämään uutta järjestelmää. Toimitusprojektin aikana tai sen päätyttyä asiakkaat saattavat esittää vaatimuksia joko uuden toiminnallisuuden muodossa tai vaihtoehtoisesti asiakkaat saattavat pyytää muokkauksia nykyisiin toiminnallisuuksiin.

### **Vaatimusten esilletuonti ja spesifikaatiot**

Haastatteluista ja havainnoista ilmeni, että usein vaatimusmäärittelyprosessi lähtee liikkeelle asiakkaan pyytämästä vaatimuksesta. Tällä hetkellä asiakkaiden tapa esittää vaatimuksia ja projektipäälliköiden tapa kirjata vaatimuksia poikkeaa hyvinkin paljon toisistaan. Kohdeyrityksessä projektipäälliköt ovat loppukädessä vastuussa vaatimusten hallinnasta ja läpiviennistä. Jotkut asiakkaat esittävät vaatimuksia excel-tilukossa, jonka jälkeen niitä käydään asiakkaan kanssa läpi, yleisesti työpajan muodossa. Jotkut taas esittävät vaatimuksia toimitusprojektin työpajoissa suullisena, jolloin vastuu vaatimusten kirjaamisesta jää kohdeyrityksen projektipäälliköille ja konsulteille. Suurin projektipäälliköistä kirjaa asiakkaiden vaatimukset omatoimisesti, kun taas muutama projektipäälliköistä vaatii asiakkailta tietyn tasoista selvitystä ennen vaatimuksen esilletuontia. Äärimmäisissä tapauksissa asiakkaat saattavat esittää vaatimuksia puhelinsoittojen tai sähköpostiviestien välityksellä. Vaatimusten esilletuonnissa ja kirjaamisessa ei tällä hetkellä ole määrämuotoisuutta organisaation tasolla.

Vaatimusten esilletuonnin jälkeen sekä asiakkaan sidosryhmät että kohdeyrityksen projektin jäsenet osallistuvat vaatimuksen tarkempaan spesifiointiin. Spesifioinnin aikana kirjaan asiakkaan vaatimukset sellaiseen muotoon, jotta projektipäällikkö pystyisi viemään vaatimukset tuotteenhallintaan ja scrum-tiimille arvioitavaksi. Tuotteenhallinta koostuu henkilöistä, jotka loppukädessä päättävät, mitä toiminnallisuuksia otetaan kehitettäväksi. Tuotteenhallinnan palaveriin osallistuu projektipäälliköiden lisäksi sekä liiketoimintajohtaja että tuotepäällikkö. Tuotteenhallinnassa myös arvioidaan, voidaanko asiakkaan vaatimukset generalisoida tuotteeseen yleisesti käytettäväksi, vai toteutetaanko vaatimus räätälöitynä asiakkaalle. Haastatteluista kuitenkin ilmeni, että joissakin tapauksissa vaatimuksista ei välttämättä keskustella tuotteenhallinnan palaverissa, vaan uuden ominaisuuden kehittämisestä saatetaan myös keskustella pienemmässä porukassa. Pääsääntöisesti kuitenkin kaikista uusista toiminnallisuuksista ja niihin liittyvistä tehtävistä keskustellaan yhdessä läpi tuotteenhallinnan palaverissa. Jotta scrum-tiimi pystyisi antamaan työmääräarvion ja teknisen toteutustavan vaatimuksista, on vaatimusmäärittelystä vastaavan henkilön kirjattava vaatimukset alustavaksi käyttäjätarinoiksi. Toimitusprojekteissa on myös olemassa vaatimuksia, jotka on määritelty jollain tasolla jo sopimusvaiheessa. Ne vaatimukset eivät sisälly tähän prosessiin, joista on tarkka työmääräarvio annettu asiakkaalle ennen projektin alkua.

### **Työmääräarvio ja ketterä kehitys**

Scrum-tiimi arvioi kirjattuja käyttäjätarinoita, jonka jälkeen scrum-tiimi antaa joko työmääräarvion ja teknisen toteutustavan, tai vaatii lisäselvityksiä käyttäjätarinoihin, mikäli niissä on epäselvyyksiä. Tässä vaiheessa scrum-tiimi vaatii lisäselvityksiä projektipäälliköiltä. Joissain tapauksissa projektipäälliköt pystyvät itse tarkentamaan vaatimuksia, kun taas joissain tapauksissa vaatimuksista keskustellaan uudestaan asiakkaan kanssa. Projektipäälliköillä on toisistaan poikkeavat tavat kirjata käyttäjätarinoita; jotkut kirjottavat pidempiä ja tarkempia selostuksia, kun taas jotkut kirjottavat muutaman sanan tarinoita. Scrum-tiimin mukaan asiakkaan vaatimuksia ei kaikissa tapauksissa kirjata tarpeeksi tarkalla tasolla, jolloin scrum-tiimin on mahdotonta antaa minkäänlaista arvioita. Scrum-tiimi on kokonaisuudessaan hajautettu, joka työ myös omat haasteet tähän asiaan. Arvion ohella scrum-tiimi myös tarkastelee, onko vaaditulla vaatimuksilla jonkinlaisia seuraamuksia. Seuraamuksilla

voidaan esimerkiksi viitata ylläpidon lisääntymiseen tai jo valmiina olevien toiminnallisuuden ohjelmistokoodin muokkaukseen.

Scrum-tiimi antaa työmääräarvion sekä kertoo mahdolliset seuraamukset, jotka voivat vaikuttaa lopulliseen työmäärään. Tämän jälkeen joko projektipäällikkö yksin tai yhdessä tuotehallinnan kanssa katselmoivat työmääräarviota, jonka jälkeen arvio viedään asiakkaalle hyväksyttäväksi. Projektipäälliköillä on toisistaan poikkeavat tavat validoida vaatimuksia asiakkaiden kanssa. Määrämuotoisuutta validointiin ei ole olemassa organisaatotasolla. Mikäli asiakas ei hyväksy arviota, niin projektipäällikkö ja asiakas aloittaa neuvottelut vaatimuksista. Tämän aikana selvitetään syyt arvion hylkäämiselle. Neuvottelujen jälkeen mennään joko takaisin spesifioimaan vaatimuksia tai sitten päätetään, ettei vaatimusta lähdetä ollenkaan toteuttamaan. Mikäli asiakas hyväksyy arvion, niin vaatimus viedään scrum-tiimille kehitettäväksi. Itse kehitystyö noudattaa ketteriä menetelmiä, jossa vaatimukset jaetaan kahden viikon sprintteihin. Kehitystyön aikana käyttäjätarinoita määritellään tarkemmalla tasolla sekä vaatimuksia priorisoidaan. Tässä diplomityössä keskitytään vaatimusmäärittelyihin prosessimielessä, jonka vuoksi ohjelmiston kehitystyöprosessiin liittyviä nyansseja ei käydä tässä diplomityössä tarkalla tasolla läpi.

### **Testaus ja toimitus**

Kehitystyöprosessin päätyttyä kehitetty inkrementti toimitetaan sisäistä testausta varten kehitysympäristöön. Scrum-tiimi suorittaa manuaaliset testaukset, jonka jälkeen inkrementti toimitetaan projektitiimille testattavaksi. Tässä vaiheessa vaatimus joko hyväksytään tai hylätään projektitiimin toimesta. Projektipäälliköillä on myös testauksiin liittyvissä tehtävissä erilaisia käytäntöjä. Jotkut kirjaavat tarkat hyväksyntäkriteerit käyttäjätarinoita tehdessä, kun taas jotkut testaavat omilla, toisistaan poikkeavilla käytännöillä. Vaatimuksen hylkäämisen yhteydessä projektitiimi arvioi, että piilekö hylkäämisen syy määrittelyissä vai onko kyseessä yksinkertaisesti ohjelmistovirhe. Mikäli kyseessä on ohjelmistovirhe, eli bugi, niin virheestä luodaan virheraportti tuotteen kehitysjonoon. Mikäli kyseessä on määrittelyvirhe, niin joissain tapauksissa projektitiimi itse määrittelee vaatimuksen uu-

destaan, ja joissain tapauksissa otetaan asiakkaaseen yhteyttä ja määritellään vaatimus ennen kuin vaatimus viedään kehitettäväksi uudestaan. Mikäli projektitiimi hyväksyy vaatimuksen, niin inkrementti toimitetaan asiakkaalle testattavaksi. Tämän jälkeen asiakas joko hyväksyy tai hylkää vaatimuksen. Mikäli asiakas hylkää, niin hylkäämisen syystä neuvotellaan projektitiimin kanssa. Neuvottelujen aikana selvitetään, että onko kyseessä määrittelyvirhe vai ohjelmistovirhe. Harvoissa tapauksissa löytyy ohjelmistovirheitä siinä kohdassa, kun asiakkaalle toimitetaan inkrementtejä testattavaksi. Hylkäyksen syyt tässä vaiheessa liittyvät usein määrittelyvirheisiin.

#### 4.1.2 Nykyiset haasteet (TK1)

Tässä kappaleessa esitellään toimintatutkimuksen diagnosointivaiheen tulokset. Diagnosointivaiheessa haastateltiin seitsemän vaatimusmäärittelyistä vastaavaa henkilöä sekä tuotekehityksen scrum-masteria. Diagnosoinninvaiheen tarkoitus oli löytää kohdeyrityksessä esiintyvät vaatimusmäärittelyprosessin haasteet puolistrukturoitujen haastattelujen avulla. Diagnosointivaiheessa tunnistetut haasteet esiintyivät laajalti myös alan kirjallisuudessa. Alla olevassa listassa havainnoidaan diagnosointivaiheessa tunnistettuja haasteita:

Haasteet
1. Vaatimusten dokumentoinnissa ja seurannassa haasteita
2. Viestintäkuilu asiakkaiden ja kehittäjien välillä
3. Haasteita vaatimusten esilletuonnissa ja analysoinnissa
4. Toteuttavuustutkimuksen puute
5. Vähäinen ajankäyttö asiakkaiden todellisten tarpeiden ja prosessien ymmärtämisessä
6. Asiakkaiden vähäinen osallistuminen toimitusprojekteissa
7. Ketterien menetelmien soveltaminen hajautetussa ohjelmistokehitystiimissä

Kuva 11. Kohdeyrityksessä esiintyvät vaatimusmäärittelyhaasteet.

Yllä mainitut haasteet ovat sekä yksittäisiä että yhteisiä. Vaatimuksista vastaavilla henkilöillä on toisistaan poikkeavat tavat käsitellä ja hallita vaatimuksia, jonka takia osa haasteista koskee kaikkia, kun taas osa haasteista koskee vain tiettyjä henkilöitä. Kohdeyritys on kovassa kasvuvaiheessa ja toimii turbulenssisessa ympäristössä, jonka vuoksi toimintatavat kehittyvät jatkuvasti. Haasteita väheksymättä kohdeyritys on kuitenkin onnistunut toimittamaan ohjelmistotuotetta onnistuneesti monelle eri yritykselle. Seuraavissa alikappaleissa esitellään haasteet tarkemmalla tasolla.

#### **4.1.2.1 Vaatimusten dokumentointi ja seuranta**

Haastatteluista selvisi, että vaatimusten dokumentoinnissa ja seurannassa on haasteita. Haastatteluissa myös selvisi, että vaatimuksista vastaavilla henkilöillä on toisistaan poikkeavat käytännöt dokumentoida vaatimuksia. Haastattelujen lisäksi tutkija kävi perusteellisesti kohdeyrityksen nykyiset vaatimusmäärittelydokumentaatiot, jotka vahvistivat haastatteluissa esiin tulleita dokumentointihaasteita.

Kohdeyrityksessä käytetään laajalti erilaisia projektinhallintamenelmiä, jotka on hyvä tiedostaa, jotta voidaan ymmärtää kohdeyrityksessä esiintyviä haasteita. Kohdeyrityksessä kehitetään ohjelmistoa sekä toimitetaan projekteja ketterien menetelmien mukaisesti. Tuotetta kehitetään scrum-menetelmällä, kun taas toimitusprojekteihin sovelletaan vaihtelevasti scrum, kanban tai scrumban -menetelmiä. Toimitusprojekteissa käytetyn projektinhallinnanmenetelmän saa projektista vastaava valita itse. Toiset priorisoivat tehtäviä ja seuraavat kahden viikon aikajaksolla niiden etenemistä, kun taas toiset projektijohtajat delegoivat tehtäviä tekijöille, eivätkä niinkään aseta kahden viikon aikaraja.

Ketterien tapojen mukaisesti kohdeyrityksessä tuotetaan dokumentaatiota minimaalisesti. Haastatteluista ja nykyisten vaatimusmäärittelymateriaalien perusteella selvisi, että kohdeyrityksessä dokumentoidaan vaatimuksia pääosin käyttäjätarinoiden avulla. Haastatteluista myös selvisi, että käyttäjätarinoiden lisäksi kohdeyrityksessä käytetään muitakin dokumentointimenetelmiä. Kaksi haastateltavista mainitsi, että he käyttävät vaatimusten do-

kumentointiin heidän itse kehittämää dokumenttitemplaattia, mikäli vaatimus on sen verran laaja, jolloin käyttäjätarinat eivät enää riitä vaatimusten dokumentointiin. Näissä tapauksissa vaatimusdokumentaatiosta luodaan käyttäjätarinoita tuotekehitystiimille. Suurin osa haastateltavista kuitenkin kertoi dokumentoivansa vaatimukset käyttäjätarinoiden muotoon. Haastatteluista selvisi, että osa haastateltavista käyttää itse kehitettyä dokumenttitemplaattia käyttäjätarinoiden lisäksi, mikäli vaatimus on sen verran laaja, jolloin käyttäjätarinat eivät enää riitä vaatimusten dokumentointiin.

Haastatteluista selvisi, että vaatimuksista vastaavat henkilöt kirjaavat vaatimuksia käyttäjätarinoiksi toisistaan poikkeavalla tavalla. Käyttäjätarinat luodaan tuotteen kehitysjonoon, jossa annetaan tiettyjä parametreja, kuten esimerkiksi mihin asiakkuuteen vaatimus liittyy ja mikä on vaatimuksen prioriteetti. Näiden parametrien lisäksi vaatimus täytyy luonnollisesti kuvata. Vaatimus pitäisi lähtökohtaisesti kuvata ennalta määritellyn pohjan avulla, jossa määritellään ketterien menetelmien mukaisesti käyttäjärooli, haluttu toiminnallisuus sekä syy toiminnallisuudelle. Haastatteluista kuitenkin selvisi, että joissain tapauksissa vaatimuksia kirjataan äärimmäisissä tapauksissa vain sanalla tai kahdella ilman minkäänlaista selitystä. Tuotekehityksen scrum-masteri mainitsi, että vaatimuksia kaikissa tapauksissa ei kuvata aina tarpeeksi tarkalla tasolla.

”Välillä kun katon noita itseni kirjoittamia tarinoita, niin en yhtään ihmettele, ettei niistä saa mitään selvää. Hyvä että itsekkään edes ymmärrän mitä tarkoitin.”

- **Projektipäällikkö**

”Vaatimuksia aina kaikissa tapauksissa kuvata tarkalla tasolla. Sana tai kaksi ei riitä vaatimuksen kuvaamiseen. Meidän täytyy ymmärtää tarkemmalla tasolla vaatimus, jotta pystymme arvioimaan kaikki tarvittavat asiat.”

- **Scrum Master**

Haastatteluista selvisi, että määrämuotoisen dokumentaation puute johtaa myös tilanteisiin, missä vaatimuksia on mahdotonta jäljittää. Tämän lisäksi haastatteluissa selvisi, että

vaatimuksiin liittyviä materiaaleja ei ole keskitetty yhteen paikkaan. Vaatimusmäärittelyihin liittyvää dokumentaatiota saattaa olla sähköposteissa, slack-keskusteluissa, asiakkaiden tiedostojärjestelmissä tai kohdeyrityksen omassa tiedonhallintajärjestelmässä.

#### **4.1.2.2 Viestintäkuilu asiakkaiden ja kehittäjien välillä**

Kohdeyrityksen kohdalla kehittäjät ovat harvoin suorassa kontaktissa asiakkaiden kanssa. Haastatteluista selvisi, että projektipäälliköt toimivat tiedonvälittäjinä asiakkaiden ja kehittäjien välillä, joka luo omat haasteet vaatimusten esilletuontiin ja dokumentointiin. Toimitusprojekteja saattaa olla käynnissä monta yhtäaikaaisesti, jolloin suoran viestinnän järjestäminen kehittäjien ja asiakkaiden välille on mahdotonta. Ketterien menetelmien mukaan asiakkaan jatkuva osallistuminen sekä suora viestintä kehittäjien kanssa on ehtona menestyneille ketterille projekteille. Kuten edellisessä kappaleessa mainittiin, niin kohdeyrityksessä vaatimusten dokumentoinnissa ei ole määrämuotoisuutta organisaatiossa, joka tuo myös omat haasteensa vaatimusten käsittelyssä.

Projektipäälliköiden toimiessa tiedonvälittäjinä ilman määrämuotoista dokumentointi ja validointiprosessia aiheuttaa joissakin tapauksissa tilanteita, missä syntyy informaatiokatkoista. Asiakkaiden todellisten tarpeiden ymmärtäminen on jo haasteellista itsessään. Haastatteluista selvisi, että monissa projekteissa ei ole varattu tarpeeksi aikaa tai budjettia asiakkaiden prosessien syvälliseen läpikäyntiin. Yleisesti kaikilla vaatimusmäärittelyprosessiin liittyvillä sidosryhmillä on oma osaamisalueensa. Asiakas yleisesti ymmärtää omat liiketoimintatarpeensa, eikä melkein missään tapauksessa ole ohjelmistokehityksen asiantuntija, kun taas kehittäjien osaaminen painottuu pääosin tuotteen ohjelmistokehityskomponentteihin. Kohdeyrityksen kontekstissa projektipäälliköiden osaaminen painottuu projektinhallintaan ja osittain myös asiakkaiden liiketoiminnan ymmärtämiseen, mutta harvoin syvällisesti ohjelmistokehitykseen. Informaatiokatko saattaa aiheuttaa vaatimusten ymmärryksessä väärinkäsityksiä, jolloin kehittäjät saattavat äärimmäisissä tapauksissa arvailla vaatimuksia.



#### 4.1.2.3 Haasteita vaatimusten esilletuonnissa ja analysoinnissa

Haastatteluista selvisi, että vaatimusten esilletuonnissa ja analysoinnissa esiintyy haasteita. Vaatimuksista vastaavilla henkilöillä on toisistaan poikkeavat käsitellä, dokumentoida ja analysoida asiakkaiden vaatimuksia. Haastatteluista selvisi, että monella projektipäälliköllä on tapana ottaa vaatimuksia vastaan asiakkailta toimitusprojektin työpajoissa. Alla olevassa listassa havainnoidaan tapoja, joilla asiakkaat esittävät toimitusprojektissa vaatimuksia:

- Työpajat ja haastattelut
- Suorat keskustelut
- Excel-listat
- Nykyisten prosessien läpikäynti

Osa asiakkaista kirjaa vaatimukset niin sanottuun ”toivelistaan” excel-taulukoon, jonka jälkeen niitä sitten käydään yhdessä läpi työpajoissa. Osa asiakkaista taas esittää vaatimuksia suullisesti työpajoissa ilman minkäänlaista etukäteisselvitystä. Tämä johtaa tilanteisiin, jossa äärimmäisissä tapauksissa asiakkaiden henkilöstöllä on erimielisyyksiä vaatimuksen tarpeesta tai sisällöstä. Joissain tapauksissa asiakkaat esittävät vaatimuksia puhelimitse tai muiden, epävirallisten kanavien välityksellä kuten sähköpostilla. Haastatteluista kuitenkin selvisi, että osalla vaatimuksista vastaavilla henkilöillä on selkeä sapluuna vaatimusten esilletuontiin ja analysointiin, mutta organisaatiossa ei tähän ole määrämuotoisuutta.

Epävirallisten kanavien kautta tulleet vaatimukset johtavat äärimmäisissä tapauksissa tilanteisiin, missä vaatimuksia ei välttämättä muisteta ottaa huomioon. Vaatimusten ollessa hajautettuna monessa eri paikassa johtaa tilanteisiin, jossa vaatimuksia on mahdotonta seurata ja ylläpitää. Osa vaatimuksista sijaitsee käyttäjätarinoina scrum-varastossa, osa sähköposteissa, osa excel-listoissa ja niin edelleen.

Haastatteluista myös selvisi, että joissakin tapauksissa uusien ominaisuuksien tai muokkauksien aiheuttamat seuraamusten miettiminen on jäänyt pienemmälle prioriteetille. Aina kun tuotteeseen kehitetään jokin uusi toiminnallisuus, niin sillä melkein poikkeuksetta aina on jonkinlaisia seuraamuksia. Seuraamuksilla viitataan tilanteisiin, jossa esimerkiksi jokin

asiakas vaatii muokkausta olemassa olevaan ominaisuuteen, jonka toteutus aiheuttaa esimerkiksi ylläpidollisia haasteita ja täten ylläpitokustannusten kasvua.

”On ehdottoman tärkeää, että vaatimuksista vastaava henkilö yrittää ottaa huomioon kaikki mahdolliset seuraamukset uudesta halutusta ominaisuudesta. Kaikissa tapauksissa tämä ei ole mahdollista, ja me kyllä scrum-tiimissä autamme parhaamme mukaan. Vaatimuksista vastaavien henkilöiden on kuitenkin hyvä tiedostaa tämä asia”

- **Scrum Master**

#### **4.1.2.4 Vähäinen ajankäyttö asiakkaiden todellisten tarpeiden ja prosessien ymmärtämisessä**

Haastatteluista selvisi, että kohdeyrityksessä ei kaikissa tapauksissa käytetä tarpeeksi aikaa asiakkaiden todellisten tarpeiden ja liiketoimintaprosessien ymmärtämiseen. Haastatteluista selvisi, että vaatimuksista vastaavilla henkilöillä löytyy riittävää osaamista ja kokemusta asiakkaiden liiketoiminta-alalta. Sen sijaan haastattelujen mukaan projekteissa ei ole varattu riittävästi aikaa ja budjettia asiakkaiden prosessien syvälliseen läpikäyntiin.

Melkein kaikissa tapauksissa kohdeyrityksen asiakkailta on entuudestaan käytössä lähdejärjestelmä, joka on hyvin usein ollut käytössä monia vuosia. Haastateltavien mukaan monissa projekteissa asiakkaat olettavat, että kohdeyrityksen henkilöstö ymmärtää heidän tarpeensa ja tuntee lähdejärjestelmän täydellisesti, josta ollaan siirtymässä pois. Väite on osittain totta, sillä kohdeyrityksen henkilöstö tuntevat osittain lähdejärjestelmän, josta asiakkaat ovat siirtymässä pois. Vaikka lähdejärjestelmien tietomalli ja toiminnallisuudet ovat melko samanlaisia kaikilla asiakkuuksilla, niin asiakkaat kuitenkin käyttävät näitä järjestelmiä eri tavalla, joka taas johtaa siihen, että nämä järjestelmän kuitenkin aina ovat hieman erilaisia.

Haastatteluista selvisi, että kaikissa projekteissa ole mahdollista suorittaa näitä vaatimusmäärittelyyn liittyviä tehtäviä kovinkaan tarkalla tasolla. Aikataulupaineet ja budjetti usein

ovat estäviä tekijöitä. Kohdeyrityksen asiakkaat eivät ole ohjelmistokehityksen ammattilaisia, jonka vuoksi vaatimusmäärittelyihin liittyviä tehtäviin ei ole varattu tarpeeksi budjettia ja aikaa.

”Olisi todella mukavaa, jos tähän voitaisiin käyttää enemmän aikaa. Mutta kuka tämän lystin maksaa? Asiakkailta on hankala laskuttaa tämän tyyppisistä tehtävistä..”

- **Projektipäällikkö**

Asiakkailla syntyy monesti myös erimielisyyksiä siitä, että mitä toiminnallisuuksia uudelta järjestelmältä toivotaan. Monesti asiakkaiden tietämys on vähäinen uuden järjestelmän mahdollisuuksista, jolloin monesti syntyy tilanteita, jossa asiakkaat vaativat vanhan järjestelmän toiminnallisuuksien olevan myös uudessa järjestelmässä. Haastatteluista selvisi, että joissakin äärimmäisissä tapauksissa vanhojen järjestelmien toiminnallisuuksia kehitettiin kohdeyrityksen järjestelmään, joka tuotti taas omat haasteensa.

#### **4.1.2.1 Toteutettavuustutkimuksen puute**

Haastatteluista selvisi, että toteutettavuustutkimusta uusista toiminnallisuuksista tai nykyisten toiminnallisuuksien muokkaamisesta ei välttämättä määrätietoisesti suoriteta. Haastatteluista selvisi, että vaatimusmäärittelyprosessi yleensä aloitetaan vaatimusten esilletuonnilla, jolloin toteutettavuustutkimus jää tekemättä. Jotta toteutettavuustutkimus voitaisiin suorittaa, täytyy kohdeyrityksen kyetä ymmärtämään tarkalla tasolla asiakkaan liiketoimintaprosessit sekä niitä tukevat vaaditut toiminnalliset ja ei-toiminnalliset vaatimukset. Haastatteluista selvisi, että toteutettavuustutkimukselle ei yleisesti ole varattu toimitusprojekteissa budjettia eikä aikaa. Tämän vuoksi joissain tapauksissa asiakkaiden vaatimuksia toteutetaan ymmärtämättä niiden todellista tarvetta.

”Mielestäni jokainen projekti pitäisi aloittaa toteutettavuustutkimuksella. En ole tottunut tällaiseen menetelmään, missä asiakkaiden vaatimuksia toteutetaan ilman, että niiden tarkoituksenperää mietitään kunnolla”

- **Projektipäällikkö**

Haastatteluista selvisi, että kaikki vaatimuksista vastaavat henkilöt haluaisivat tehdä toteutettavuustutkimusta, mutta projekteissa ei ole varattu aikaa eikä budjettia sen suorittamiseen. Kolme haastateltavasta mainitsi, että jo myyntivaiheessa tämä asia pitäisi ottaa huomioon ja sisällyttää toteutettavuustutkimukseen menevä aika projektin kokonaisbudjettiin.

#### **4.1.2.2 Asiakkaiden vähäinen osallistuminen toimitusprojekteissa**

Haastatteluista selvisi, että monissa toimitusprojekteissa asiakkaiden sidosryhmien osallistuminen oli puutteellista. Haastateltavien mukaan asiakkaat monesti luulevat, että kohdeyrityksen henkilöstö ymmärtää täydellisellä tasolla heidän vaatimuksensa ja tarpeensa, jos niistä on joskus lyhyesti keskusteltu joko myyntiprosessin aikana tai projektin alkuvaiheessa. Haastateltavien mukaan asiakkaat eivät monesti tiedosta kuinka syvälle täytyy mennä, jotta jostain tietystä vaateesta voidaan saada tarpeellinen ymmärrys. Pintapuolinen läpikäynti ei yksinkertaisesti ole riittävä ohjelmistokehitysprojekteissa. Ketterien projektien onnistuvuuden kannalta tärkein asia on asiakkaan sidosryhmien jatkuva osallistuminen projektin jokaisessa vaiheessa. Haastatteluista kuitenkin selvisi, että tämä ei koske läheskään kaikkia projekteja. Joissakin projekteissa asiakkaat kyllä tiedostavat, että heidän jatkuva osallistuminen on edellytys projektin onnistumisen kannalta. Haastateltavien mukaan monet asiakkaista eivät ymmärrä osallistumisen tärkeysastetta, ja heidän mielestään tätä tietoisuutta pitäisi lisätä jo projektin myyntivaiheessa.

#### **4.1.2.3 Ketterien menetelmien soveltaminen hajautetussa ohjelmistokehitystiimissä**

Haastatteluista selvisi, että kohdeyrityksen kohdalla ketterien menetelmien soveltamisesta hajautetussa ohjelmistokehitystiimissä syntyy haasteita. Kohdeyrityksen työntekijät ovat maantieteellisesti hajautettu eri toimipisteisiin. Suurin osa kommunikaatiosta tapahtuu erilaisten digitaalisten työkalujen kautta. Suurin osa kohdeyrityksen työntekijöistä ovat nimellisesti Helsingin tai Tampereen toimipisteellä. Osa työntekijöistä sijaitsee taas hajautettuna ympäri maailmaa eri paikoissa. Ketterien menetelmien mukaisesti kohdeyri-

tyksessä dokumentoidaan todella vähän. Dokumentaation kautta tapahtuvan tiedonvälityksen sijaan kohdeyrityksessä pyritään kommunikoimaan tarvittavien jäsenten kesken suullisesti. Ohjelmistokehitystiimin scrum-masterin mukaan vaatimukset ovat ajoittain epäselviä, jolloin niistä tarvitaan lisäselvityksiä ennen varsinaisen määrittelytyön aloittamista. Samankaltaisia tapauksia on ollut monta, jossa vaatimuksia ei ole joko tarpeeksi tarkalla tasolla tai sitten kehitystiimillä on ollut hankaluuksia ymmärtää vaatimuksia. Kohdeyrityksessä kehittäjät harvoin ovat suorassa kontaktissa asiakkaiden kanssa. Sen sijaan projektipäälliköt toimivat niin sanottuina viestinvälittäjinä asiakkailta kehitystiimille. Sama pätee myös toiseen suuntaan, sillä kehittäjät saattavat tarvita lisätietoja vaatimuksista myös asiakkailta. Projektipäälliköiden toimiessa viestinvälittäjinä ilman määrämuotoista dokumentointia ja prosessia on johtanut äärimmäisissä tapauksissa tilanteisiin, jossa on lähdetty kehittämään ominaisuutta siten, että se ei ole vastannut ollenkaan asiakkaan tarpeita.

## **4.2 Toiminnan toteutuksen tulokset**

Seuraavassa alikappaleissa havainnoidaan toiminnan toteutuksen tuloksia. Toiminnan toteutus suoritettiin järjestelmällä iteratiivinen työpaja tutkijan ja kohdeyrityksen valittujen työntekijöiden kesken. Valituilla työntekijöillä viitataan ihmisiin, joita haastateltiin diagnostivaiheen aikana.

### **4.2.1 Iteratiivisen työpajan käytännöt ja menetelmät**

Toiminnan toteutuksen tavoitteet voidaan jakaa kahteen osaan, joita havainnoidaan alla olevassa listassa:

- 1) Luoda iteratiivinen keskustelu diagnostivaiheen aikana esiin tulleista haasteista
- 2) Löytää diagnostivaiheen aikana esiin tulleista haasteista jokin tietty kipukohta, johon haetaan yhteisvoimin konkreettisia ratkaisukeinoja

Iteratiivisen keskustelun tavoitteena oli saada työpajaan osallistuneet henkilöt keskustelemaan keskenään diagnosointivaiheen aikana esiin tulleista haasteista. Esiin nousseiden haasteiden lisäksi työpajan tarkoituksena oli luoda ilmapiiri, jossa työpajaan osallistuneet henkilöt pystyisivät kertomaan omista vaatimusmäärittelykäytännöistään ja sitä kautta haakea suuntaviivaa kehitykselle. Työpajan alussa käydyssä yleisessä keskustelussa selvisi, että näistä asioista ei ole aikaisemmin keskusteltu tasolla, jossa kaikki vaatimuksista vastaavat henkilöt olisivat istuneet saman pöydän ääressä keskustelemassa vaatimusmäärittelykäytännöistä ja haasteista. Toisena tärkeimpänä tavoitteena oli löytää diagnosointivaiheen aikana tulleista haasteista jokin tietty kipukohta, johon haettaisiin konkreettisia ratkaisukeinoja ja ehdotuksia tuleville toimenpiteille. Haastatteluista selvisi, että kohdeyrityksessä esiintyy haasteita laajalti monella eri aihealueella. Kipukohdan, eli jonkin tietyn yhteisen haasteen löytäminen ja konkreettisten ratkaisukeinojen löytäminen valittiin toiminnan toteutuksen päätavoitteeksi tutkijan ja työpajaan varatun rajallisen aikataulun vuoksi. Toimintatutkimuksen toteutuksen työpajan vaiheet havainnollistetaan alla olevassa listassa:

- 1) Haastatteluissa esiin tulleiden haasteiden esittäminen ja niistä keskusteleminen
- 2) Kohdeyrityksen nykyisen vaatimusmäärittelyprosessin läpikäynti ja peilaaminen nykyisiin haasteisiin
- 3) Vaihtoehtoisten ratkaisujen havainnointi / tiedostaminen ja kipukohdan löytäminen

#### **4.2.2 Haastatteluissa esiin tulleiden haasteiden esittäminen ja niistä keskusteleminen**

Toimintatutkimuksen toteutuksen ensimmäisessä vaiheessa tutkija esitteli osallistujille haastatteluiden aikana esiin tulleita haasteita. Työpajaan osallistuivat samat henkilöt, joita haastateltiin diagnosointivaiheen aikana. Haasteet olivat sekä yhtenäisiä että yksittäisiä, osa haasteista koski kaikkia henkilöitä, kun taas osa haasteista koski vain tiettyjä henkilöitä. Tutkija esitti osallistujille diagnosointivaiheen aikana esiin tulleita haasteita ja mahdollisia parannusehdotuksia, jotka pohjautuivat aikaisempiin tutkimuksiin ja alan kirjallisuuteen. Jokaisen haasteen esittämisen jälkeen työpajaan osallistuneet henkilöt toivat esiin omia näkökulmia ja mielipiteitä. Haasteista, mahdollisista parannusehdotuksista ja kehitystä es-tävistä asioista käytiin iteratiivista keskustelua osallistujien kesken. Keskustelujen aikana

tutkija havainnoi jatkuvasti osallistujia ja heidän näkökulmiansa asioista. Joidenkin haasteiden kohdalla keskustelu jäi hieman vähemmälle. Nämä olivat pääpiirteittäin sellaisia haasteita, jotka eivät koskeneet kaikkia osallistujia. Sen sijaan tiettyjen haasteiden kohdalla osallistujat keskustelivat keskenään pitkään ja yrittivät löytää yhteisymmärryksen kyseiselle haasteelle. Moni näistä esiin tulleista haasteista ovat päällekkäisiä, eli liittyvät hyvin vahvasti toisiinsa. Työpajassa tehtyjen havaintojen tarkoituksena oli löytää jokin tietty yhtenäinen kipukohta, johon haettiin ratkaisukeinoja yhteisvoimin.

### **Viestintäkuilu ja ketterien menetelmien soveltaminen hajautetussa tiimissä**

Viestintä ohjelmistokehittäjien ja asiakkaiden välillä koettiin myös haasteelliseksi monissa tapauksissa. Jopa asiakkaiden sisäinen viestintä on joissain tapauksissa haasteellista, kun vaatimusten tarpeissa ja priorisoinneissa on erimielisyyksiä. Haastatteluista myös selvisi, että asiakkaat eivät monesti itse tiedä mitä he haluavat. Kohdeyrityksen kohdalla vaatimuksesta vastaavat henkilöt toimivat niin sanottuina viestinvälittäjinä. Nämä henkilöt välittävät asiakkaiden tarpeet ohjelmistokehittäjille usein käyttäjätarinoiden muodossa. Osallistujien mukaan haasteena on myös kielenkäännökset, sillä kohdeyrityksen asiakkaat ovat melkein kaikissa tapauksissa suomenkielisiä, kun taas sen sijaan ohjelmistokehittäjät englanninkielisiä. Kohdeyrityksen kohdalla tämä tarkoittaa sitä, että asiakkaat tuovat vaatimuksen esiin Suomen kielellä, jotka sitten käännetään Englanninkieliseksi, kun niistä tehdään käyttäjätarinoita.

”Projektipäälliköiden toimiessa tiedon välittäjinä kehitystiimille on mielestäni erittäin haasteellista. Kielikäännöksistä johtuvat seikat hankaloittavat tätä asiaa vielä enemmän, sillä kehittäjät saattavat ymmärtää vaatimuksen täysin eri tavalla kuin oli alun perin tarkoitus.”

#### **- Projektipäällikkö**

Kohdeyrityksen työntekijät ovat maantieteellisesti hajautettuina eri paikkoihin, joka osallistujien mukaan synnyttää myös omat haasteensa. Tämän lisäksi iteratiivisen keskustelun

tuloksena osallistujien välillä ketterien menetelmien soveltaminen hajautetussa ohjelmistokehitystiimissä on myös haasteellista. Ketterien menetelmien mukaisesti dokumentaatiota tuotetaan minimaalisesti, mikä osallistujien ja tuotteen scrum-masterin mukaan on tuottanut jonkin verran epäselvyyksiä vaatimuksissa.

”Dokumentaation pitäisi olla sillä tasolla, että sieltä löytyy tarpeeksi yksityiskohtaista tietoa, mutta ei kuitenkaan liian yksityiskohtaista. Rehellisesti kukaan ei jaksakaan lukea liian pitkiä dokumentteja.”

- **Scrum-master**

Keskusteluista selvisi, että viestintään ja vaatimusten esittämiseen ja dokumentointiin tarvittaisiin vankempaa määrämuotoisuutta. Osallistajat keskustelivat, että käyttäjätarinoiden epäselvyyden vuoksi vaatimusten ymmärtämisessä saattaa tulla epäselvyyksiä. Scrum-masterin mukaan sen sijaan vaatimusmäärittelijöiden kirjaamat käyttäjätarinat ovat ajoittain epäselviä. Scrum-masterin mukaan liika dokumentaatio ei myöskään ole suotuisaa, sillä liian pitkistä dokumentaatiosta on hankala löytää se olennainen tieto. Näin ollen dokumentaation määrää täytyy myös tasapainottaa sopivaksi. Iteratiivisen keskustelun perusteella osallistajat tulivat siihen yhteisymmärrykseen, että vaatimusten esittämiseen ja dokumentointiin täytyy löytää yhtenäiset käytännöt. Erityisesti dokumenttien laajuuteen on löydettävä sopiva tasapaino, jossa asiat on kuvattu tarpeeksi tarkalla tasolla, mutta ei kuitenkaan tarpeettoman yksityiskohtaisesti.

### **Dokumentointiin ja vaatimusten esilletuontiin liittyvät haasteet**

Ensimmäisenä listalla oli esitetty dokumentointiin ja vaatimusten esilletuontiin liittyvät haasteet, joka synnytti runsaasti keskustelua osallistujien välillä. Tämä tietty haaste oli yhtenäinen haaste, joka koski kaikkia osallistujia. Osallistajat tutkijan voimin kävivät iteratiivista keskustelua dokumentointiin ja esilletuontiin liittyvistä haasteista ja parannusehdoksista.



”Vaatusmääritystistä pitäisi jo ennen tarjouspyyntöjä olla tehtynä ensimmäinen versio.”

- **Projektipäällikkö**

”Meidän pitäisi puhua ratkaisukuvauksen toimittamisesta. Mielestäni menemme tässä niin sanotusti latvasta puuhun, kun me toimittajana kirjoitamme asiakkaalle vaatimusmääritystä.”

- **Projektipäällikkö**

Osallistujien mukaan vaatimusmääritystistä pitäisi olla jo ennen tarjouspyyntöjä tehtynä ensimmäinen versio. Muut osallistajat olivat myös tästä samaa mieltä. Osallistujien mukaan kuitenkin tämä ei kohdeyrityksen kohdalla ole kaikissa tapauksissa mahdollista, sillä useimmissa tapauksissa asiakkailla ei ole osaamista tehdä vaatimusmääritystä ilman ulkopuolista konsultointia. Osaamisen puuttumisen lisäksi monet osallistajat nostivat esiin sen, että vaatimusmääritysten myyminen ennen tarjouksen tekemistä on erittäin hankalaa. Yhden osallistujan mukaan kohdeyrityksessä mennään usein niin sanotusti latvasta puuhun, eli lähdetään kirjoittamaan asiakkaalle vaatimusmääritystä. Asiakkailta täytyisi tulla vaatimusmääritykset ja kohdeyrityksen tehtävänä olisi kirjoittaa ratkaisukuvaus vaatimusmääritysten perusteella. Muut osallistajat olivat asiasta myös samaa mieltä. Estävänä tekijänä tässä kuitenkin usein on asiakkaiden riittämätön tietotaito vaatimusten laatimisessa sekä rajallinen aika toimittaa projektit asiakkaalle. Osallistujien mukaan tähän pitäisi pystyä tulevaisuudessa varaamaan sekä aikaa ja budjettia, jotta asiakkaan vaatimukset saataisiin aikaisessa vaiheessa selville. Osallistujien mukaan projektit myydään niin sanotusti vesiputousmallisesti, eli määritellään tarkka budjetti ja aikataulu projekteille ilman tarkkoja etukäteismäärityksiä. Toisin sanoen, useimmiten projektit vedetään läpi ketterästi mutta myydään vesiputousmallisesti. Tämä tarkoittaa sitä, että vaatimusmääritysten dokumentoinnissa sovelletaan ketteriä vaatimusmääritysmenetelmiä, kuten käyttäjätarinoiden käyttämistä. Osallistujien mukaan etukäteisvaatimusmääritysten puute projekteissa tuo haasteita, sillä monesti sitoudutaan tiukkoihin aikatauluihin ja budjetteihin tietämättä tarkkoja vaatimuksia. Tämän lisäksi iteratiivisessa työpajassa selvisi, että vaatimusten määrällisen dokumentaation puute on johtanut monesti väärinymmärryksiin asiakkaan

puolella. Väärinymmärryksillä viitataan tilanteisiin, missä asiakas on olettanut, että jokin tietty spesifinen toiminnallisuus kuuluu tuotteeseen tai sovittuun toimitusprojektiin.

### **Vähäinen ajankäyttö asiakkaiden todellisten tarpeiden ja prosessien ymmärtämisessä**

Osallistujien mukaan projekteissa ei ole varattu tarpeeksi budjettia ja aikaa asiakkaiden todellisten tarpeiden ja prosessien ymmärtämiseen. Monesti asiakkaat olettavat, että toimittajat ymmärtävät täysin heidän tarpeensa ja prosessinsa. Osallistujien mukaan tämä väite on osittain totta, sillä kohdeyrityksen työntekijöillä on pitkä kokemus ja tietotaito asiakkaiden liiketoiminta-alalta. Osallistujien mukaan ongelmaksi kuitenkin muodostuu se, että melkein jokaisella asiakkaalla on toisistaan poikkeavat tavat käyttää lähdejärjestelmiä. Lähdejärjestelmien lisäksi asiakkailla on useimmiten monia muita järjestelmiä, joiden avulla liiketoimintaprosesseja hoidetaan. Nämä erilaiset asiakkaiden käyttämät järjestelmät tuovat omat haasteensa.

”Meidän pitäisi pystyä lisäämään asiakkaan tietoisuutta siitä, että minkälaisia toimenpiteitä tarvitaan, jotta projekti saadaan onnistuneesti suoritettua. Monesti asiakkaat pitävät meitä toimittajia ajatustenlukijoina.”

#### **- Projektipäällikkö**

”Onnistuneen projektin isoin edellytys on se, että toimittaja on ymmärtänyt asiakkaan prosessit ja tarpeet. Asiakkaan pitää myös ymmärtää ja tiedostaa omat tarpeensa”.

#### **- Projektipäällikkö**

Osallistujien mukaan kohdeyrityksellä on tietotaitoa ja kokemusta ohjata asiakkaita ja haastaa heidän vaatimustarpeitaan. Edellytyksenä kuitenkin on se, että projektin alussa varattaisiin tarpeeksi aikaa ja budjettia prosessien ymmärtämiseen. Osallistujien mukaan monissa tapauksissa asiakkaatkaan eivät osaa artikuloida tarpeitaan tarpeeksi selvästi. Osallistajat kävivät keskustelua ja tulivat siihen lopputulokseen, että niin sanottujen perustavanlaatuisien kysymysten ja prosessien läpikäyntiin on varattava tarpeeksi aikaa, jotta asiakkaiden tarpeet ymmärretään mahdollisimman aikaisessa vaiheessa.

Keskusteluissa päädyttiin siihen, että etukäteisvaatimusmäärittelyjen järjestäminen myyntivaiheessa on ainakin tällä hetkellä mahdotonta toteuttaa. Osallistujat tulivat kuitenkin yhteisymmärrykseen, jossa projektin alussa asiakkaiden prosessien ja todellisten tarpeiden ymmärtämiseen voitaisiin käyttää enemmän aikaa ja hakea vaihtoehtoisia ratkaisukeinoja.

”Nyt kun ajattelee, niin tämän vaiheen tarpeellisuus on ihan itsestäänselvyys. Meidän täytyy aina projektin alussa varata aikaa ja käydä keskusteluja siitä, että millä tavalla asiat hoidetaan nyt, mitä asiakkaat tavoittelevat, mikä on se liiketoiminnallinen syy miksi asiakkaat tarvitsevat jotain toiminnallisuutta jne. Tämänhetkisten aikataulusyiden ja budjetin vähyden vuoksi meillä ei ole ollut muuta mahdollisuutta, kun lähteä esittämään meidän järjestelmäämme asiakkaille ja kysyä heiltä esittelyjen aikana minkälaisia ominaisuuksia asiakkaat haluavat käyttää. Tällä hetkellä ongelma on se, ettei tälle vaiheelle ole myyty yhtään aikaa. Meillä ei siis ole ollut tähän asti muuta vaihtoehtoa, kun lähteä kouluttamaan asiakkaita ja sitten vaatimukset ja tarpeet tulevat asiakkailta ti-poittain”.

- **Projektipäällikkö**

#### **4.2.3 Vaihtoehtoisten ratkaisujen havainnointi ja kipukohtien löytäminen (TK2)**

Osallistujat löysivät yhteisymmärryksen ja kipukohdat haasteista, joista käytiin keskustelua. Osallistujien mukaan asiakkaiden liiketoimintaprosessien ymmärtämiseen täytyy ehdottomasti varata projektin alussa enemmän aikaa ja budjettia, sekä kehitettävä yhtenäiset käytännöt, joiden avulla saadaan ymmärrys asiakkaiden prosesseista ja vaatimustarpeista. Tämän lisäksi teknisen vaatimusmäärittelydokumentaation laajuuteen ja käyttäjätarinoiden kriteereihin tarvitaan muutosta. Toisin sanoen, yhtenäisiksi kipukohtiksi muodostui tekijät, joita havainnoidaan alla olevassa listassa:

1. **Puute asiakkaiden prosessien ja tarpeiden ymmärtämisessä**
2. **Vaatimusmäärittelydokumentaation laajuus**

Tutkielman aikataulurajoitteiden ja resurssien vuoksi keskitytään löytämään mahdollisia ratkaisukeinoja testattavaksi näihin yllä mainittuihin haasteisiin.

### **Ratkaisukeinoja asiakkaiden prosessien ymmärtämiseen**

Osallistujien mukaan toimitusprojektien projektinhallintamenetelmiin tarvitaan määräämuotoisuutta ja selkeyttä. On mahdotonta käyttää pelkästään ketteriä vaatimusmenetelmiä, jos projektit on myyty asiakkaille vesiputousmallisesti tiukan budjetin ja aikataulun kera. Osallistujien mukaan toimitusprojekteissa pitäisi isossa kuvassa soveltaa vesiputousmallista projektinhallintamenetelmää, joka sisältää tuotteen toimituksen asiakkaalle ilman mitään ylimääräisiä, asiakaskohtaisia muutospyyntöjä.

”On erittäin hankalaa pysyä aikataulussa ja budjetissa, jos projekteja johdetaan ketterästi ja annetaan asiakkaille jatkuvasti mahdollisuus vaatia sellaisia asioita, joista me emme olleet tietoisia projektin alussa.”

#### **- Projektipäällikkö**

Pääpiirteittäin asiakkaat tulevat loppujen lopuksi käyttämään monia perusominaisuuksia, jotka kohdeyrityksen tarjoamassa tuotteessa löytyy vakiona. Näiden perustoiminnallisuuksien lisäksi asiakkailla on useimmiten muutospyyntöjä, jotka erityisesti tukevat juuri heidän liiketoimintaansa. Osallistujien mukaan täysin vesiputousmallisia vaatimusmäärittelyjä ei voida toteuttaa, sillä projekteissa täytyy kuitenkin olla mahdollisuus muutoksille, mikäli asiakkailla syntyy tarpeita toimitusprojektin aikana. Näissä erillisissä muutostöissä voitaisiin osallistujien mukaan soveltaa ketteriä vaatimusmäärittelymenetelmiä, jotka kulkevat sitten omaa polkuunsa toimitusprojektin kyljessä. Kohdeyrityksen tuotekehitystiimi kehittää ohjelmistoa kahden viikon aikaikkunassa ketterien menetelmien mukaisesti, jolloin on luonnollista, että myös muutospyynnöt ja tarvittavat konfiguraatiomuutokset noudattavat myös samoja ketteriä periaatteita. Toisin sanoen keskusteltiin, että kohdeyrityksen projekteissa pyrittäisiin soveltamaan niin sanottu hybridimallia, jossa vesiputousmenetelmiä käytettäisiin yhdessä ketterien menetelmien kanssa.

Työpajassa keskusteltiin ja haettiin yhteisvoimin alustavaa määrämuotoista ja yhtenäistä tapaa, jonka avulla saataisiin ymmärrys asiakkaiden nykytilasta, prosesseista ja vaatimustarpeista. Keskusteluissa päädyttiin siihen, että kehitetään määrämuotoinen ja yhtenäinen vaatimusten spesifikaatiopohja, eli määrittelymalli, jolla yritettäisiin heti toimitusprojektin alussa saada selkoa siihen, miten asiakkaat nykyään käyttävät lähdejärjestelmiään ja miten ne sopeutuvat kohdeyrityksen tarjoamaan ohjelmistoon, johon asiakkaat ovat siirtymässä. Määrittelymalli sisältäisi muun muassa perustavanlaatuisia kysymyksiä asiakkaiden prosesseista ja kohdeyrityksen lähdejärjestelmän ohjelmistokomponenteista ja toiminnallisuuksista. Osallistujien välillä tiedostettiin, että yksi ainoa määrittelymalli ei tule sopimaan kaikkiin hankkeisiin asiakkaiden liiketoiminta-alojen heterogeenisten taustojen vuoksi. Toimintatutkimuksen toteutuksen työpajassa ei ollut aikaa suunnitella määrittelymallin sisältöä. Sen sijaan työpajassa sovittiin jatkotoimenpiteet määrittelymallin kehitykselle. Osallistujien välillä myös tiedostettiin, että iteraatiokierroksia tarvitaan monta, jotta määrittelymalli vastaa kohdeyrityksen tarpeita.

### **Ratkaisukeinoja hajautetun ketterän menetelmän haasteisiin**

Osallistujien mukaan vaatimusten dokumentaation laajuuteen ja tarkkuuteen tarvitaan muutoksia. Tähän asti käytetyt osittain strukturoimattomat käyttäjätarinat eivät riitä. Tämän lisäksi käyttäjätarinoiden hyväksymiskriteereihin tarvitaan muutosta. Dokumentaation määrää täytyy tasapainottaa niin, että se vastaa kohdeyrityksen tarpeita. Osallistujat tiedostivat sen, että liikadokumentointi ei myöskään ole sopiva ratkaisu, sillä se vie ensinnäkin aikaa vaatimusmäärittelijältä sekä vaatimuksen toteuttajalla saattaa tulla hankaluuksia löytää tarvittavat tiedot. Dokumentaation muutosprosessi ei saisi myöskään olla liian raskas, sillä vaatimuksia täytyy pystyä muuttamaan ilman raskasta muutosprosessia.

Työpajassa keskusteltiin tavoista, jonka avulla tuotekehitystiimille annettujen vaatimusten dokumentointia ja viestintää voitaisiin tehostaa. Osallistujien välillä tiedostettiin, että kohdeyrityksen organisaatorakenteen ja työntekijöiden maantieteellisten seikkojen vuoksi vaatimusmäärittelyihin tarvitaan määrämuotoisuutta ja yhtenäisyyttä. Vaatimusten jäljitettävyyden takia työpajassa myös keskusteltiin vaatimusmateriaalien keskittämisestä,

jonka avulla vaatimukseen voitaisiin tarvittaessa palata. Työpajassa sovittiin, että käyttäjätarinoiden hyväksymiskriteerien muutosta varten järjestetään alustava työpaja, jossa määritellään tarvittavat kriteerit. Osallistujien välillä tiedostettiin, että yksi ainoa työpaja ei riitä, vaan iteraatiokierroksia on oltava monta, jotta käyttäjätarinoista ja dokumentaation laajuudesta saadaan sen malliset, jotka sopivat kohdeyrityksen käyttöön. Näihin työpajoihin osallistuu vaatimusmäärittelijöiden lisäksi tuotekehityksen scrum-tiimi. Iteraatiokierroksia arvioidaan ja niihin tehdään palautteiden mukaan parannuksia.

## 5 Johtopäätökset

Tämän diplomityön ensimmäisenä tavoitteena oli tutkia kohdeyrityksen vaatimusmäärittelyprosessissa esiintyviä haasteita vaatimusmäärittelijöiden näkökulmasta. Diplomityön toisena tavoitteena oli löytää nykyisestä vaatimusmäärittelyprosessista jokin tietty kipukohta, johon voitaisiin soveltaa uusia, määrämuotoisia menetelmiä. Tässä kappaleessa havainnoidaan diplomityön empiiristen tulosten johtopäätöksiä peilaamalla haastatteluista saatua aineistoa olemassa oleviin tutkimuksiin, joita havainnoidaan luvussa 2 (tausta ja kirjallisuus). Tämän lisäksi tässä kappaleessa havainnoidaan diplomityön vaikutusta kohdeyrityksessä sekä suosituksia mahdollisille lisätutkimuksille.

### 5.1 TK1: Haasteet nykyisessä vaatimusmäärittelyprosessissa

Ensimmäisen tutkimuskysymyksen tavoitteena oli selvittää, minkälaisia haasteita kohdeyrityksen nykyisessä vaatimusmäärittelyprosessissa esiintyy vaatimusmäärittelijöiden näkökulmasta. Näin ollen haastateltaviksi valittiin ne henkilöt, jotka ovat vastuussa kohdeyrityksen vaatimusmäärittelyprosessin hallinnasta. Ensimmäiseen tutkimuskysymyksen tieto kerättiin puolistrukturoiduilla haastatteluilla, jonka tuloksia havainnoidaan kappaleessa 4.1.

Haastatteluista kerätyn tiedon avulla saatiin perusteellinen ymmärrys kohdeyrityksen vaatimusmäärittelyprosessin haasteista. Asiakkaiden liiketoiminta-alojen heterogeisuuden vuoksi kohdeyrityksellä on ollut haasteita heidän vaatimusmäärittelyprosessissansa. Tä-

män lisäksi vaatimusmäärittelyistä vastaavilla henkilöillä on toisistaan poikkeavat käytännöt vaatimusten hallintaan ja läpivientiin liittyvissä tehtävissä. Näin ollen kohdeyrityksessä ei ole organisaatiotasolla määrämuotoista tapaa käsitellä ja hallita vaatimuksia. Kohdeyrityksessä kuitenkin pääsääntöisesti noudatetaan ketterän ohjelmistokehityksen periaatteita. Kohdeyrityksen työntekijät ovat maantieteellisesti hajautettuna ympäri maailmaa. Tämän vuoksi suurin osa kommunikaatiosta tapahtuu erilaisten digitaalisten viestintätyökalujen kautta. Vallitsevan koronatilanteen vuoksi kaikki viestintä myös kohdeyrityksen ja heidän asiakkaiden välillä tapahtuu digitaalisesti, joka on tuonut myös omat haasteensa. Näin ollen kohdeyrityksen toimintaa voidaan luonnehtia hajautetuksi ohjelmistokehitykseksi.

Kohdeyrityksessä tuotteen kehittäjät eivät kommunikoi suoraan asiakkaiden kanssa. Suoran kommunikoinnin sijaan hankkeiden projektipäälliköt välittävät asiakkaiden tarpeet kehittäjille. Asiakkaiden tarpeista tuotetaan käyttäjätarinoita projektipäälliköiden toimesta, jotka toimitetaan tuotekehitystiimille kehitettäväksi. Tämän lisäksi ketterien menetelmien mukaisesti kohdeyrityksessä tuotetaan vaatimusmäärittelydokumentaatio minimaalisesti. Dokumentaatiota tuotetaan yleensä käyttäjätarinoiden muodossa, joka on tuonut omat haasteensa. Haastatteluista selvisi, että käyttäjätarinoita ei aina kaikissa tapauksissa ole kirjattu tarkalla tasolla, mikä on johtanut pahimmillaan vaatimusten väärinymmärrykseen. Vaatimusten dokumentoinnin määrämuotoisuuden puutteen vuoksi vaatimusten jäljitettävyydessä on ollut myös haasteita. Tämän lisäksi haastatteluista selvisi, että vaatimuksiin liittyviä materiaaleja ei ole keskitetty yhteen paikkaan. Näin ollen kohdeyrityksellä on ollut moninaisia haasteita vaatimusmäärittelymenetelmien soveltamisessa hajautetussa ohjelmistokehitystiimissä.

Samankaltaisia haasteita löytyy alan kirjallisuudesta runsaasti. Cao & Ramesh (2010) argumentoivat, että hajautetuissa tiimeissä dokumentaation puute saattaa aiheuttaa väärinymmärryksiä vaatimuksissa. Soares et.al (2015) mukaan dokumentaation puute saattaa johtaa tilanteeseen, jossa kehittäjät joutuvat arvailemaan vaatimuksia, jolloin pahimmassa tapauksessa kehittäjät kehittävät vääränlaisia toiminnallisuuksia. Alqahtani et.al (2013) mu-

kaan organisaatioilla on haasteita löytää tasapainoa dokumentaation laajuudessa. Haastatteluista selvisi, että kohdeyrityksessä törmätään samaan haasteeseen, missä dokumentaation tasapainoa on hankala löytää. Minimaalinen dokumentointi on johtanut kohdeyrityksessä tilanteisiin, missä kehittäjillä on ollut hankaluuksia ymmärtää vaatimuksen kokonaiskuvaa. Shristava & Date (2010) mukaan ketterissä menetelmissä olisi suotavaa dokumentoida siinä määrin, jotta voitaisiin minimoida mahdollisia väärinymmärryksiä. Cao & Ramesh (2010) argumentoivat, että määrämuotoisen dokumentaation rooli kasvaa tiimeissä, jossa ei voida saavuttaa tarpeeksi korkeaa vuorovaikutusta kasvotusten. Haastatteluista selvisi, että kohdeyrityksen hajautetun luonteen vuoksi korkealaatuiseen ja intensiiviseen kasvokkain käytävää vuorovaikutusta ei ole mahdollista toteuttaa kaikissa tapauksissa. Cao & Ramesh argumentoivat, että vaatimusten dokumentaation puute osoittautuu haasteelliseksi tapauksissa, missä vaatimuksia täytyy jäljittää.

Suoran kommunikaation puutteesta löytyy myös alan kirjallisuudesta samankaltaisia haasteita. Measey et.al (2015) mukaan tehokkain ja toimivin tapa tiedon välittämiseksi tiimin jäsenten kesken on kasvokkain käytävä keskustelu. Paetsch et.al (2003) argumentoivat, että kehitystiimin ja asiakkaan välillä ei saisi olla erillistä viestintäkerrosta. Kohdeyrityksen toimitusprojekteissa kehittäjien ja asiakkaiden välissä on usein erillinen vaatimuksista vastaava henkilö, joka välittää vaatimukset asiakkailta kehittäjille. Cao & Ramesh (2010) ja Paetsch et.al (2003) mukaan kasvokkain käytävän viestintämenetelmän tehokkuus riippuu suuresti asiakkaiden ja kehittäjien välisestä intensiivisestä vuorovaikutuksesta. Cao & Ramesh (2010) argumentoivat, että tämä lähestymistapa aiheuttaa riskejä hankkeissa, joissa ei voida saavuttaa korkealaatuista intensiivistä vuorovaikutusta. Cao & Ramesh (2010) intensiivisen vuorovaikutuksen puute saattaa johtaa tilanteisiin, missä vaatimusten väärinymmärryksen tuloksena kehitetään vääränlaisia toiminnallisuuksia.

Haastatteluista selvisi, että kohdeyrityksellä on ollut hankaluuksia vaatimusten esilletuonnissa ja analysoinnissa. Samankaltaisia haasteita löytyy alan kirjallisuudesta runsaasti. Cao & Ramesh (2010) tekemän tutkimuksen mukaan organisaatioilla on haasteita määrittellä hankkeille tarkkaa aikataulua ja budjettia, sillä ketterien menetelmien mukaisesti kehitys-



kaaren alussa pyritään tunnistamaan vain korkean tason vaatimukset ja kriittisimmät toiminnallisuudet. Cao & Ramesh (2010) argumentoivat, että ketterissä menetelmissä ei käytetä resursseja vaatimusten kokonaisvaltaiseen havainnointiin, sillä asiakkaiden muuttuvien tarpeiden vuoksi tarkemmat vaatimukset syntyvät kehityksen elinkaaren aikana. Ketterien menetelmien mukaisesti toimitusprojektien alussa kohdeyrityksessä pyritään tunnistamaan korkean tason vaatimukset, jonka jälkeen tarkemmat vaatimukset luodaan iteraatioiden kautta. Tämä lähestymistavan toteuttaminen on kuitenkin kohdeyrityksessä osoittanut haasteelliseksi, sillä toimitusprojekteille usein asetetaan tarkka aikataulu ja budjetti. Ohjelmistotuotteen perustoiminnallisuuksien toimittamisen lisäksi asiakkailla on tarpeita modifikaatioille, joille ei yleensä ole annettu tarkkaa hintaa tai toteutusaikataulua. Haastattelujen mukaan asiakaskohtaisten modifikaatioiden vaatimusmäärittelyprosessissa esiintyy myös haasteita, mutta syynä eivät ole aikatauluihin tai budjettiin liittyvät tekijät. Sen sijaan haasteena näissä asiakkaiden modifikaatioissa piilee määrämuotoisen dokumentaation puutteessa sekä vähäinen suora kommunikointi asiakkaiden ja kehittäjien välillä. Shristava & Date (2010) argumentoivat, että hajautetuissa tiimeissä suositetaan perinteisiä menetelmiä, jossa yksityiskohtaiset vaatimukset dokumentoidaan ja toimitetaan kehitystiimille kehitettäväksi. Haastatteluista kuitenkin selvisi, että kohdeyrityksessä ei haluta joutua tilanteeseen, jossa minimaalisen dokumentaation sijaan dokumentoitaisiin liian yksityiskohtaisesti.

## **5.2 TK2: Sovellettavat vaatimusmäärittelykäytännöt ja kipukohtien havainnointi**

Toisen tutkimuskysymyksen tavoitteena oli selvittää, millaisilla käytännöillä voitaisiin parantaa kohdeyrityksen tuoteliiketoiminnan hajautettujen ohjelmistokehityshankkeiden vaatimusmäärittelyprosessia. Toisen tutkimuskysymyksen päätavoitteena oli löytää diagnosointivaiheen aikana esiin tulleista haasteista kipukohtia, johon haettaisiin mahdollisia ratkaisukeinoja ja ehdotuksia tuleville toimenpiteille. Toiseen tutkimuskysymykseen saatiin vastaus järjestelmällä iteratiivisen työpaja tutkijan ja kohdeyrityksen valittujen työntekijöiden kesken. Työpajan ensimmäisenä tavoitteena oli luoda intensiivistä vuorovaikutusta osallistujien kesken diagnosointivaiheen aikana esiin tulleista haasteista. Työpajan toisena

tavoitteena oli löytää diagnosointivaiheen aikana esiin tulleista haasteista kipukohtia, johon haettiin konkreettisia ratkaisukeinoja ja ehdotuksia tuleville toimenpiteille. Iteratiivisen työpajan alussa käydyssä keskustelussa selvisi, että näistä asioista ei ole aikaisemmin keskusteltu tasolla, jossa kaikki vaatimuksista vastaavat henkilöt olisivat istuneet saman pöydän ääressä keskustelemassa näistä haasteista. Iteratiivisen työpajan aikana ei suunniteltu määrittelymallin sisältöä. Sen sijaan työpajassa sovittiin jatkotoimenpiteet ja vastuuhenkilöt määrittelymallin kehitykselle.

Iteratiivisessa työpajassa keskusteltiin diagnosointivaiheen aikana esiin tulleista haasteista. Joistakin haasteista syntyi hieman vähemmän keskustelua, kun taas joissakin haasteissa keskustelua syntyi runsaasti. Iteraatioiden kautta työpajassa oivallettiin kipukohtat ja niiden priorisoinnissa päästiin yhteisymmärrykseen. Osallistujien mukaan asiakkaiden liiketoimintaprosessien ymmärtämiseen ja sitä kautta käyttäjien vaatimusten havainnointiin täytyy varata projektin alussa enemmän aikaa. Tavoitetta lähestyttäisiin kehittämällä yhtenäiset määrittelymenetelmät. Tämän lisäksi vaatimusmäärittelydokumentaatioiden laajuutta täytyy selventää sekä käyttäjätarinoiden hyväksymiskriteerejä tiukentaa. Näin ollen kipukohdiksi muodostuivat puute asiakkaiden prosessien ja todellisten tarpeiden ymmärtämisessä sekä dokumentaation tasapainon löytäminen.

Määrämuotoisilla ja yhtenäisillä määrittelymenetelmillä kohdeyrityksessä kyettäisiin tehostamaan asiakkaiden todellisten tarpeiden havainnointia. Toimitusprojektien perustoinnallisuuden toimittamisen osalta asiakkaiden vaatimukset täytyy havaita projektin alkuvaiheissa, sillä toimitusprojekteissa on usein määritelty tarkka aikataulu ja budjetti, jolloin tietynlainen jäykkyys on tavoiteltavaa. Toimitusprojektin ulkopuolisiin modifikaatioihin tarvitaan joustavuutta, sillä usein asiakkaiden modifikaatiovaatimukset muodostuvat toimitusprojektin elinkaaren aikana. Tämän vuoksi kohdeyrityksessä olisi suotavaa soveltaa sekä perinteisiä että ketteriä vaatimusmäärittelymenetelmiä. Ketterissä vaatimusmäärittelyissä tarvitaan myös jäykkyyttä kohdeyrityksen hajautetun luonteen vuoksi. Käyttäjätarinat eivät yksinään riitä, sillä kohdeyrityksessä ei ole mahdollista toteuttaa korkealaatuista vuorovaikutusta asiakkaiden ja kehittäjien välillä, kuten ketterissä menetelmissä on yleisesti tapana. Goddesdiener (2003) ja Sommerville & Kotonya (1998) argumentoivat, että ei

ole yhtä ainoa menetelmää, jolla saataisiin tehokkaasti vaatimukset kaikissa ympäristöissä. Sen sijaan vaatimusmäärittelymenetelmien valinnassa täytyy ottaa monia tekijöitä huomioon, kuten organisaatioiden rakenne, asiakkaiden liiketoiminta-alat sekä henkilöstöjen kompetenssit. Näin ollen sopivien määrittelymenetelmien käyttö kohdeyrityksessä riippuu siitä, minkälaisia vaatimuksia ollaan sillä hetkellä spesifioimassa. Alan kirjallisuudessa mainitaan, että vaatimusten spesifiointi on hankkeen onnistuvuuden kannalta erittäin tärkeä vaihe, sillä spesifioinnin aikana pyritään havainnoimaan loppukäyttäjien tosiallisia tarpeita. Davis et al (2006) argumentoivat, että vaatimusten esilletuonti ja spesifiointi ovat vaatimusmäärittelyprosessin tärkein vaihe. Noraini & Adullah (2011) mukaan vaatimusten esilletuonti on yksi vaatimusmäärittelyprosessin haastavimmista tehtävistä, sillä siihen sisältyy moninaisia tehtäviä, kuten asiakkaiden tarpeiden ja haasteiden ymmärtämistä, sekä vaatimusten toimittamista kehittäjille, sillä ehdolla, ettei niissä synny mahdollisia väärinymmärryksiä.

Näin ollen kohdeyrityksen vaatimusmäärittelyhaasteet liittyvät pitkälti sekä sisäiseen että ulkoiseen viestintään. Ketteriin vaatimusmäärittelymenetelmien lisäksi kohdeyrityksen täytyy myös soveltaa perinteisiä menetelmiä siinä määrin, kun se on tarpeellista. Vaatimusten esilletuontiin täytyy kuitenkin soveltaa määrämuotoisia määrittelymalleja. Määrämuotoisilla ja yhtenäisillä määrittelymalleilla kohdeyritys kykenee entistä paremmin havainnoimaan asiakkaiden todelliset tarpeet. Oikeanlaisten määrittelymallien havainnoimiseen tarvitaan kuitenkin useita iteraatiokierroksia. Aikataulurajoitteiden vuoksi tässä tutkielmassa ei ollut aikaa ja resursseja suunnitella määrittelymallin sisältöä. Sen sijaan toisen tutkimuskysymyksen päätavoitteena oli löytää kipukohdat ja sopia jatkotoimenpiteet ja vastuuhenkilöt menetelmien kehitykselle.

### **5.3 Tutkimuksen vaikutus kohdeyrityksessä ja toimenpidesuositukset**

Tässä kappaleessa havainnoidaan tutkimuksen vaikutuksia sekä toimenpidesuosituksia, joiden avulla kohdeyrityksen vaatimusmäärittelyä voitaisiin tehostaa. Yleisesti organisaatioille sopivien vaatimusmäärittelymenetelmien havainnointi on haasteellista, sillä menetel-

mien toimivuuteen vaikuttaa muun muassa organisaation kulttuuri, hierarkiarakenne, henkilöstöjen kompetenssit sekä liiketoiminta-ala. Tämän lisäksi tässä kappaleessa yleistetään tutkimustulosten kautta havaittuja toimensuosituksia. Nämä yleistettävät toimenpidesuosituksukset ovat aika- ja organisaatiosidonnaisia, joka tarkoittaa sitä, että tässä luvussa havainnoidut toimenpidesuosituksukset ovat valideja vain tässä yhdessä tietyssä yrityksessä tiettyyn aikaan.

### **Selkeät menetelmät eri tarkoituksiin**

Alan kirjallisuudessa löytyy runsaasti tutkimuksia, jossa havainnoidaan, että sopivien menetelmien toimivuus on vahvasti organisaatiosidonnaista. Yousuf & Asger (2015) mukaan ei ole olemassa täydellistä vaatimusmäärittelysapluunaa, joka toimisi kaikissa yrityksissä ja kaikissa tilanteissa. Sama pätee myös tutkittavalle kohdeyritykselle. Jäykkyyttä siellä, missä sitä vaaditaan, sekä toisinaan joustavuutta, missä sitä tarvitaan. Ei ole yhtä ainuttakaan oikeaa menetelmää, joka täydellä varmuudella toimisi kaikissa kohdeyrityksen hankkeissa ja kehitystöissä. Toimitusprojektien perustoiminnallisuuksien osalta vaaditaan jäykkyyttä ja vahvaa laajuuden (scope) hallintaa, mikäli toimitusprojektit myydään asiakkaille ennalta määrättyyn hintaan sekä toimitukselle asetetaan tarkka aikataulu. On erittäin vaikeaa hallita projekteja ketterästi ja antaa asiakkaille jatkuvasti mahdollisuuksia vaatia sellaisia asioita, joita ei tiedetty hankkeen alussa, mikäli näissä halutaan pysyä aikataulussa ja budjetissa. Jäykkyyttä voidaan lähestyä toimeenpanemalla määrämuotoisia ja perinteisempiä määrittelymalleja, joita havainnoitiin tutkimuksen tuloksissa. Sen sijaan toimitusprojektien ulkopuolisten lisätoiminnallisuuksien kehittämiseen vaaditaan joustavuutta ja intensiivistä vuorovaikutusta asiakkaiden kanssa. Cao & Ramesh (2010) mukaan joustavuudella voidaan saavuttaa korkeampi asiakastyytyväisyys, sillä asiakkaat ovat mukana kehityksen elinkaaren aikana ohjaamassa toiminnallisuuksia oikeaan suuntaan. Kohdeyrityksen tuotekehitysorganisaatio käyttää pääasiallisesti ketteriä menetelmiä, jolloin toiminnallisuuksien vaatimusmäärittely pitäisi olla ketterää ja joustavaa.

### **Määrämuotoisuutta ja tasapainoa dokumentointiin**

Kohdeyrityksessä on tähän asti käytetty pääasiallisena dokumentaation muotona osittain strukturoimattomia käyttäjätarinoita. Nämä eivät kuitenkaan riitä kaikissa tapauksissa. Kohdeyrityksen työntekijät ovat hajautettuna maantieteellisesti ympäri maailmaa, jonka vuoksi kohdeyrityksessä joudutaan kommunikoidaan erilaisten digitaalisten viestintätyökalujen kautta. Vallitsevan koronatilanteen vuoksi näiden digitaalisten viestintätyökalujen asema on vielä vahvistunut entisestään, jonka johdosta viestintä kohdeyrityksen ja heidän asiakkaidensa välillä tapahtuu myös täysin digitaalisesti. Näin ollen kohdeyritys toimii täysin hajautetusti. Muutokset pakottavat kaikkia organisaatioita kehittämään menetelmiä, mikäli kilpailukyky halutaan säilyttää.

Kohdeyrityksen hankkeissa vaatimuksista vastaavat henkilöt toimivat niin sanottuina viestinvälittäjinä. Nämä henkilöt välittävät asiakkaiden tarpeet ohjelmistokehittäjille, jonka vuoksi syntyy iso kuilu asiakkaiden ja kehittäjien välille, jolloin vuorovaikutuksen laatu kärsii. Tämän lisäksi kohdeyrityksen asiakkaat ovat suomenkielisiä, kun taas moni kehittäjä englanninkielisiä. Ilman korkealaatuista ja intensiivistä kasvokkain käytävää keskustelua kohdeyrityksen täytyy tuoda enemmän määrämuotoisuutta ja tasapainoa dokumentaatioihin. Näin voidaan minimoida mahdolliset väärinymmärrykset ja epäselvyydet, joita syntyy väkisin kehitystyön aikana. Näitä väärinymmärryksiä syntyy myös hankkeissa, missä asiakkaat ovat suorassa kontaktissa kehittäjien kanssa, jolloin voidaan päätellä, että ilman korkealaatuista vuorovaikutusta ja määrämuotoista dokumentaatiota syntyy suuri riski vaatimusten väärinymmärryksille. Väärinymmärrysten riski myös kasvaa, kun vaatimuksista keskustelevilla henkilöillä on täysin eri osaamisalueet. Asiakkaiden edustajilla on tietynlaiset kompetenssit, jotka liittyvät heidän liiketoiminta-alaansa. Kohdeyrityksessä sen sijaan projektipäälliköillä on omat osaamisalueensa, ja kohdeyrityksen kehittäjille taas omat. Mikäli vuorovaikutuksessa olevilla henkilöillä ei ole samaa jaettua kokemusta, niin viestin ymmärtämiseen vaaditaan entistä yksityiskohtaisempaa tietoa.

Kohdeyrityksen rakenne, projektien luonteet sekä asiakkaiden liiketoiminta-alat määrittävät sen, kuinka tarkalle tasolla vaatimuksia täytyy dokumentoida. Tasapainon löytäminen

ei ole helppoa, sillä liian minimaalinen dokumentaatio saattaa aiheuttaa väärinymmärryksiä. Toisaalta liian yksityiskohtaisessa dokumentoinnissa piilee myös omat haasteensa, sillä toiminnallisuuden kehittämisen kannalta tarvittavaa tiedon löytäminen voi olla pitkän työn takana. Tasapainon löytäminen vaatii lukuisia iteraatiokierroksia ja retrospektiivejä, jotka edesauttavat dokumentaation tasapainon havainnoimista.

### **Asiakkaiden todellisten tarpeiden havainnointi ja prosessien ymmärrys määrämuotoisella määrittelymallilla**

Onnistuneen projektin isoin edellytys on se, että toimittaja on ymmärtänyt asiakkaan prosessit ja tarpeet tosiasiallisesti. Kohdeyrityksessä tarvitaan määrämuotoisia ja yhtenäisiä määrittelymalleja, joiden avulla voidaan saada selkoa asiakkaiden prosessiin ja tosiasiallisiin vaatimuksiin. Kohdeyrityksessä täytyy projektien alussa varata entistä enemmän aikaa ja budjettia näiden asioiden läpikäyntiin, jotta mahdollisilta väärinymmärryksiltä voidaan välttyä. Menestymisen kannalta kohdeyrityksen täytyy tiedostaa millä tavalla asiakkaat käyttävät nykyisiä järjestelmiä, mitkä ovat heidän tavoitteensa ja mitkä ovat ne liiketoiminnalliset syyt, miksi tiettyjä modifikaatioita vaaditaan. Alan kirjallisuudessa mainitaan, että vaatimusten esilletuonti (elicitation) on vaatimusmäärittelyprosessin tärkein vaihe (Davis et.al, 2006). Vaatimusten esilletuonnissa pyritään tuomaan käyttäjien tai organisaatioiden vaatimuksia esiin. Kohdeyrityksen toimitusprojektien perustoiminnallisuuksien toimittamisessa vaatimusten esilletuonti olisi suotavaa noudattaa perinteisempää menetelmää, jossa kaikkien asiakkaiden sidosryhmien vaatimukset saataisiin tietää toimitusprojektin alussa. Näin vältetään mahdollisilta yllätyksiltä, jotka vaikuttavat toimitusprojektin aikatauluun ja budjettiin. Asiakkaiden modifikaatiopyynnöissä sen sijaan suositellaan edelleen ketterien menetelmien soveltamista, sillä koko tuotekehitysorganisaatio toimii ketterien menetelmien mukaisesti.

### **Asiakkaiden tietoisuuden lisääminen**

Kohdeyrityksen asiakkaat toimivat pääsääntöisesti kunnossapidon tai elintarvike ja energiateollisuuden liiketoiminta-aloilla. Tämän vuoksi kohdeyrityksen asiakkaat eivät ole ohjelmistokehityksen ammattilaisia, jolloin asiakkaiden tietoisuus hankkeiden onnistumiseen

vaadittavista tekijöistä on vähäistä. Näin ollen kohdeyrityksen vastuulla on asiakkaiden tietoisuuden lisääminen niistä tekijöistä, joita vaaditaan onnistuneisiin projekteihin. Asiakkaiden sidosryhmien osallistuminen on elintärkeää menestyneen hankkeen kannalta sekä ketterissä että perinteisissä menetelmissä. Perinteisissä menetelmissä asiakas on vahvasti mukana ennen kehitystyön alkua, kun taas ketterissä menetelmissä asiakas on mukana koko ohjelmiston elinkaaren aikana. Kohdeyrityksen asiakkaat eivät aina tiedosta, että heidän intensiivinen osallistumisensa on elintärkeää projektin onnistumisen kannalta. Monissa hankkeissa kohdeyrityksen asiakkaat olettavat, että kohdeyritys ymmärtää kaikki tarpeet ja prosessit ilman tosiasiallista läpikäyntiä. Syynä on todennäköisesti asiakkaiden tiedostamattomuus tekijöistä, joita näihin menestyneisiin hankkeisiin vaaditaan. Toisena syynä voi olla se, että asiakkaiden sidosryhmät keskittyvät omaan työtehtäviin, jolloin heillä ei ole tarpeeksi osallistua näihin vaatimusmäärittelyihin. Asiakkaiden tietoisuuden lisääntymisen myötä vaatimusmäärittelyihin voitaisiin myös mahdollisesti käyttää enemmän aikaa.

#### **5.4 Tutkimuksen kontribuutio**

Tässä luvussa havainnoidaan tutkimustyön kontribuutiota. Tämän diplomityön tavoitteena oli ensinnäkin luoda yhtä yritystä ja toimialaa laajempi ymmärrys ohjelmistoliiketoimintalalla käytetyistä vaatimusmäärittelykäytännöistä selvittämällä alalla sovellettuja käytäntöjä olemassa olevien tutkimusten ja kirjallisuuden kautta. Aihetta lähestyttiin suorittamalla toimintatutkimus kohdeyritykselle, joka on erikoistunut tuotteistettuun ohjelmistoliiketoimintaan. Toimintatutkimuksen tavoitteena oli selvittää, millaisia haasteita kohdeyrityksen vaatimusmäärittelyprosessissa esiintyy määrittelijöiden näkökulmasta, sekä löytää kipukohtia, joihin haettiin vaihtoehtoisia ja määrämuotoisia ratkaisukeinoja.

Nykyisten haasteiden selvittämistä varten tutkimuksessa haastateltiin kohdeyrityksen vaatimusmäärittelyprosessista vastaavia henkilöitä, sekä kehitystiimin scrum-masteria. Empiiristen tutkimustulosten aineistoa peilattiin olemassa oleviin tutkimuksiin, joiden kautta tehtiin lopullisia tulkintoja. Empiiriset tutkimustulokset osoittivat, että alan kirjallisuudessa havaittuja vaatimusmäärittelyhaasteita esiintyi myös kohdeyrityksen kohdalla. Ketterien vaatimusmäärittelymenetelmien soveltaminen hajautetussa ohjelmistokehityksessä on

haasteellista, sillä hajautetun luonteen vuoksi suurin osa kommunikaatiosta tapahtuu erilaisten digitaalisten työkalujen kautta. Ketterien menetelmien mukaista intensiivistä vuorovaikutusta ei ole mahdollista toteuttaa hajautetuissa tiimeissä, jonka vuoksi haasteiden taklaamiseen vaaditaan määrämuotoisempia menetelmiä, kuten alan kirjallisuudessa on mainittu. Ketterien menetelmien mukaiset minimaaliset dokumentointikäytännöt saattavat synnyttää väärinymmärryksiä vaatimusmäärittelyissä. Voidaan siis sanoa, että hajautetuissa tiimeissä vaaditaan tietynlaista jäykkyyttä, mitä yleisesti perinteisissä menetelmissä tavoitellaan. Jäykkyydelle täytyy kuitenkin löytyä sopiva tasapaino, sillä organisaatioiden muuttuvien liiketoimintatarpeiden vuoksi vaaditaan myös tietyissä osa-alueissa ketteryyttä.

Alan kirjallisuudessa mainitaan, että hajautettujen ketterien vaatimusmäärittelymenetelmien haasteet liittyvät pitkälti sidosryhmien väliseen kommunikointiin. Samankaltaisia haasteita esiintyy myös kohdeyrityksessä. Tutkimuksen kontribuutio on hyvin käytännönläheinen, sillä tutkimustulosten selvittämiseen haastateltiin henkilöitä, jotka ovat tosiasiallisesti tehneet vaatimusmäärittelyitä eri toimialoilla vuosikausia. Tutkimuksen aikana suoritetuissa haastatteluissa havainnoitiin hyvin perusteellisesti kohdeyrityksessä esiintyviä haasteita, joiden tuloksia hyödynnettiin peilaamalla esille tulleita haasteita olemassa oleviin tutkimuksiin. Tutkimustulosten perusteella löydetyt haasteet voidaan pitkälti yleistää alan kirjallisuudessa mainittuihin haasteisiin. Sen sijaan tutkimuksissa ehdotettuja toimenpidesuosituksia ei voida sellaisenaan yleistää tässä tutkimuksessa, sillä organisaatioihin sopivien vaatimusmäärittelymenetelmien löytäminen on vahvasti organisaatiosidonnaista. Jotkut menetelmät toimivat tietyissä ympäristöissä, kun taas niiden samojen menetelmien käyttö toisessa yrityksessä voisi aiheuttaa suuria riskejä.

## **5.5 Tutkimuksen rajoitteet**

Tutkimuksessa esiintyy rajoitteita, jotka luonnollisesti vaikuttavat tutkimustulosten luotettavuuteen. Ensinnäkin yleisesti toimintatutkimuksissa on haasteita tutkimuksen toistetta-



vuuden kanssa. Tämä diplomityö suoritettiin yhdelle kohdeyritykselle, jonka vuoksi tutkimuksen tulokset ovat vahvasti sidonnaisia kohdeyritykseen. Mikäli tutkimus olisi suoritettu toiselle yritykselle, niin tuloksissa ja suosituksissa saattaisi olla eroavaisuuksia, sillä jokaisella organisaatio on uniikki omalla tavallaan. Tutkimustuloksiin ja suosituksiin vaikuttavat muun muassa organisaation kulttuuri, hierarkiarakenne, henkilöstöjen kompetenssit sekä organisaation liiketoiminta-ala. Näin ollen tutkimuksen tulokset ovat vahvasti organisaatiosidonnaisia, jonka johdosta tutkimukset ovat valideja kohdeyrityksessä tutkimuksen teon aikana. Tutkimuksen tulosten luotettavuuteen vaikutti myös tutkijan rooli kohdeyrityksessä. Tutkielmaa aloittaessa tutkija oli työskennellyt yrityksessä yli kolme vuotta, jonka vuoksi tutkittavat haasteet olivat jo entuudestaan tuttuja, joka puolestaan saattoi aiheuttaa ennakkoluuloja haasteita kohti. Kohdeyrityksen prosessit ja työntekijät olivat tutkijalle jo entuudestaan tuttuja, joka edesauttoi aiheen fokuointia. Toisaalta tutkimustulokset saattaisivat olla erilaiset, mikäli tutkija olisi täysin ulkopuolinen henkilö.

Kohdeyrityksen vaatimusmäärittelyprosessin haasteiden selvittämistä varten tutkimuksessa haastateltiin vaatimusmäärittelyprosessista vastaavia henkilöitä sekä tuotekehityksen scrum-masteria. Tulosten luotettavuuden kasvattamiseksi myös kehittäjiä ja muita kohdeyrityksen työntekijöitä olisi voinut haastatella. Tutkimuksessa kuitenkin keskityttiin vaatimusmäärittelyihin liiketoiminnan ja prosessien näkökannalta, jonka takia tutkija fokuusi haastattelut liiketoimintatasolle. Täten tutkielmassa ei otettu huomioon vaatimusmäärittelyihin liittyviä haasteita, jotka liittyivät ohjelmistokehityksen teknisiin haasteisiin. Toimintatutkimuksessa suoritettiin vain yksi iteraatiokierros, mikä vaikuttaa tutkimuksen tuloksiin ja jatkosuositukseen. Tutkijan rajallisen aikataulun ja resurssien vuoksi tutkimuksessa ei ollut mahdollisuutta suorittaa monia iteraatiokierroksia, joiden avulla olisi saatu tarkempia vastauksia tutkimuskysymyksiin. Rajallisen aikataulun vuoksi tutkielmassa ei päästy testaamaan ja validoimaan parannusehdotuksia. Tämän vuoksi tutkimustulokset nojautuvat kipukohtien havainnointiin ja parannusehdotusten suosituksiin.

Kohdeyrityksessä tiedostettiin, että vaatimusmäärittelyprosessissa on haasteita, mutta tarkkoja yhtenäisiä syitä ei ennen tämän diplomityön suoritusta tiedostettu. Tiedostamisen lisäksi näistä haasteista ei ole aikaisemmin keskusteltu tasolla, jossa kaikki vaatimuksista

vastaavat henkilöt olisivat istuneet saman pöydän ääressä keskustelemassa vaatimusmäärittelykäytännöistä ja haasteista. Näin ollen organisaation johdolta saadun palautteen mukaan tutkimuksen tärkein opetus kohdeyritykselle oli haasteiden yleinen tiedostaminen.

## 6 Lähteet

Conboy, K., Coyle, S., Wang, X., Pikkarainen, M. (2010). People over process: Key People Challenges in Agile Development.

Cheng, B.H.C., Atlee, J.M. (2007). Research Directions in Requirements Engineering. *Future of Software Engineering*, 285-303, 200

Sillitti, A., Succi, G. (2005). Requirements Engineering for Agile Methods. *Engineering and Managing Software Requirements*, pp. 309-326.

Soares, H.F., Alves, N.S.R., Mendes, T.S., Mendonca, M., Spinola, R.O. (2015). Investigating the link between user stories and documentation debt on software projects. In: *2015 Proceedings of the 12th International Conference on Information Technology - New Generations*, pp. 385– 390.

Cao, L., Ramesh, B. (2010). Agile requirements engineering practices: An empirical study. *IEEE Software*, 25(1):60 –67, 2008. ISSN 0740-7459.

Davis, C., Fuller, R., Tremblay, M., Berndt, D. (2006). Communication challenges in requirements elicitation and the use of the repertory grid technique. *Journal of Computer Information Systems* 46, 78–86

Nerur, S., Mahapatra, R. & Mangalaraj, G. (2005). Challenges of migrating to agile methodologies. *Communications of the ACM*, 48(5), 72–78.

Paetsch, A., Eberlerin, A., Maurer, F. (2003). Requirements Engineering and Agile Software Development,” *Proc. 12th IEEE Int’l Workshops Enabling Technologies: Infrastructure for Collaborative Enterprises (Wetice 03)*, IEEE CS Press, 2003, p. 308.

Kotonya, G., & Sommerville, I. (1998). *Requirements engineering: Processes and techniques*. Wiley Publishing.

Zave, P. (1997). Classification of research efforts in requirements engineering. *ACM Computing Surveys (CSUR)*, 29(4), 315-321.

Nuseibeh, B., Easterbrook, S. (2000). Requirements engineering: A roadmap. In *Proceedings of the Conference on the Future of Software Engineering* (pp. 35-46). ACM.

Wohlin, C., Petersen, K., Baca, D. (2009). The Waterfall Model in Large-Scale Development. F. Bomarius et al. (Eds.): *PROFES 2009, LNBIP 32*. 386–400.

Nerur, S., Mahapatra, R. & Mangalaraj, G. (2005). *Challenges of migrating to agile methodologies*. *Communications of the ACM*, 48(5), 72–78.

Balaji, S., Sundarajan Murugaiyan, M. (2012). Waterfall Vs V-Model Vs Agile: A Comparative Study on SDLC. *International Journal of Information Technology and Business Management*. JITBM & ARF. ISSN 2304-0777.

Schwaber, K. (1997). *Scrum development process. Business object design and implementation*. In 10th Annual Conference on Object Oriented Programming Systems, Languages, and Applications Addendum to the Proceedings. ACM/SIGPLAN October. (s. 117–134) Springer.

Dybå, T., Dingsøy, T. (2008). *Empirical studies of agile software development: A systematic review*. *Information and Software Technology*, 50(9), 833–859.

Larman, G. (2007). *Agile and iterative development*. 9th edition. Addison-Wesley Professional. ISBN-0131111558.

Pressman, R.S. (2005). *“Software Engineering: A Practitioner’s Approach”*, Sixth Edition, McGraw-Hill Higher Education, ISBN: 0-07-285318-2

Robertson, S., Robertson, J. (2006). *Mastering the Requirement Process*, Second Edition, Addison-Wesley, ISBN: 9780321419491

Glinz, M. (2007). On non-functional requirements. In Requirements Engineering Conference, 2007. RE'07. 15th IEEE International (pp. 21-26). IEEE.

Sommerville, I. (2004). *Software Engineering, Seventh Edition*. Pearson Education.

Cobb, G. (2015). *The project manager's guide to mastering Agile: Principles and practices for an adaptive approach*. USA: John Wiley & Sons.

P.Abrahamsson, O., Salo, J., Ronkainen, J., Warsta. (2002). *Agile Software Development Methods – Review and Analysis*. VTT Electroniikka.

D. E. Avison, F., Lau, M., D. Myers., P. A. Nielsen. (1999). "Action research," *Communications of the ACM*, 42(1), pp. 94–97

R. L. Baskerville., A. T. Wood-Harper. (2016). "A Critical Perspective on Action Research as a Method for Information Systems Research," in *Enacting Research Methods in Information Systems: Volume 2*, Cham: Springer International Publishing, pp. 169–190.

Gillis, A., & Jackson, W. (2002). *Research methods for nurses: Methods and interpretation*. Philadelphia: F.A. Davis Company.

McNiff, J., Whitehead, J. (2006). *All you need to know about action research*. Thousand Oaks, CA: Sage

Braun, V. & Clarke, V. (2006). *Using thematic analysis in psychology*. *Qualitative Research in Psychology*, 3(2), 77-101.

Measey, P., Wolf, L., Berridge, C., Gray, A., Oliver, L., Roberts, B., Short, M., & Wilmshurst D. (2015). *Agile Foundations: Principles, practices and frameworks*. BCS Learning & Development Limited.

M.Paasivaara & C.Lassenius. (2006). Could Global Software Development Benefit from Agile Methods? *International Conference on Global Software Engineering*, p.109- 113, IEEE

Kamaruddin, N., N., Arshad, A. Mohamed. (2012) Chaos issues on communication in agile global software development [online] 2012 IEEE Business, Engineering & Industrial Applications Colloquium (BEIAC). [cited 6 Mar 2019] DOI: 10.1109/BEIAC.2012.622609

Alzoubi, Y., A. Gill & A. Al-Ani. (2016) Empirical studies of geographically distributed agile development communication challenges: A systematic review. [online] *Information and Management* 1 January 2016, Vol.53(1), pp.22-37. [cited 23rd Aug. 2018] <https://doi.org/10.1016/j.im.2015.08.003>

Alqahtani, A., J. Moore, D. Harrison & B. Wood (2013) The Challenges of applying distributed agile software development: A systematic review [online] *International journal of advances in Engineering & Technology*, Jan 2013, Vol. 5, issue 2, pp. 23-36. [cited 5th of Jan. 2019] ISSN: 2231-1963

Orb, A., Eisenhauer, L. ja Wynaden, D. (2001). " Ethics in qualitative research ". *Journal of nursing scholarship* 33.1, s. 93–96.

Kajko-Mattsson, M., Azizyan G., Magarian M. (2010) Classes of Distributed Agile Development Problems [online] 2010 Agile Conference [cited 6 Mar 2019] DOI 10.1109/AGILE.2010.14

Gottesdiener, E. (2002). *Requirements by Collaboration*. Boston: Addison-Wesley

M. Yousuf and M. Asger. (2015). "Comparison of various requirements elicitation techniques," *International Journal of Computer Applications*, vol. 116, issue 4, pp. 8–15, Apr. 2015.

C. P. Noraini and M. Z. Abdullah. (2011). "Requirement elicitation: identifying the communication challenges between developer and customer," *International Journal on New Computer Architectures and Their Applications*, pp. 371–3

