



TAMPEREEN TEKNILLINEN YLIOPISTO

JYRI VUORINEN
SCRUM-MENETELMÄN KÄYTTÖ PIRKANMAALAISISSA
OHJELMISTOYRITYKSISSÄ
Diplomityö

Tarkastaja: professori Tommi Mikkonen
Tarkastaja ja aihe hyväksytty
Tieto- ja sähkötekniikan
tiedekuntaneuvoston
kokouksessa 8.12.2010

TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO

Tietotekniikan koulutusohjelma

VUORINEN, JYRI: Scrum-menetelmän käyttö Pirkanmaalaisissa ohjelmistoyrityksissä

Diplomityö, 50 sivua, 7 liitesivua

Tammikuu 2011

Pääaine: Ohjelmistotuotanto

Tarkastaja: professori Tommi Mikkonen

Avainsanat: Scrum, Agile, ohjelmistotuotantomenetelmä, ketterät menetelmät, haastattelututkimus

Ketterät ohjelmistotuotantomenetelmät ovat laajasti käytössä ohjelmistoteollisuudessa. Yksi suosituimmista menetelmistä on Scrum, joka sisältää tyypillisen ketterän menetelmän tapaan sovelluskehityksen iteratiivisuutta ja tiiviissä yhteistyössä toimivan tiimin. Huolimatta ketterien menetelmien laajasta levinneisyydestä niiden käytöstä tehty tutkimus on hyvin hajanaista.

Tämän diplomityön tarkoituksena on lisätä yleistä tietoutta siitä, miten Scrumia käytetään oikeassa yritys-elämässä sekä tunnistaa menetelmän hyötyjä ja heikkouksia. Diplomityössä esitellään ketterät menetelmät ja Scrumin toimintaperiaatteet, sekä kootaan ensin systemaattisella artikkeliarvioinnilla Scrum-menetelmän käytöstä saadut aiemmat tutkimustulokset yhteen. Tämän lisäksi toteutetaan haastattelututkimus yhdelletoista Pirkanmaan alueen pienelle tai keskisuurelle ohjelmistoyrityksille Scrum-menetelmän käytöstä.

Tuloksina saadaan selville millaisia ketteriä käytäntöjä nämä yritykset soveltavat, miten paljon Scrumin käytäntöjä on omaksuttu ja miten tarkasti sen menettelytapoja yrityksissä noudatetaan. Tärkeänä tuloksena kuvataan myös Scrum-menetelmällä saavutettuja hyötyjä sekä sen käyttöön liittyviä ongelmia ja haasteita. Työssä kuvataan myös yritysten omia hyviä ketteriä käytäntöjä. Lisäksi haastattelututkimuksen aineistoa verrataan myös aiempiin Scrum-tutkimustuloksiin.

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY

Master's Degree Programme in Information Technology

VUORINEN, JYRI: The use of the Scrum method in IT companies in Pirkanmaa area

Master of Science Thesis, 50 pages, 7 Appendix pages

January 2011

Major: Software engineering

Examiner: Professor Tommi Mikkonen

Keywords: Scrum, Agile, software development method, survey

Agile software development methods have been widely adopted in software industry. One of the most popular Agile methods is Scrum. Like many other Agile methods Scrum emphasizes iterative software development and closely working team. Despite the popularity of these methods, the research results about the use of agile methods remain scarce.

This thesis aims at increasing the information about the real usage of Scrum in software companies and also to recognize the benefits, problems and challenges of Scrum. The thesis introduces Agile software development methods and the principles of Scrum-method. A systematic article study is then used to gather information and results of previous Scrum studies. In addition to that, an interview of eleven Finnish small and middle sized software companies is used to find out the experiences of using Scrum-method.

The thesis describes what kind of Agile methods these companies use, how are the Scrum practices adopted, and how well the Scrum practices are followed. An important result is also to describe the benefits achieved using Scrum as well as the associated problems and challenges. The thesis also describes various good agile practices from these companies. Furthermore, the results of the interviews are compared to the previous Scrum studies.

ALKUSANAT

Tämä diplomityö on tehty Tampereen teknillisen yliopiston ohjelmistotekniikan laitokselle. Työssä keräsin tietoa Scrum-menetelmästä tehdyistä tutkimuksista, sekä haastattelin yhtätoista Pirkanmaalaista yritystä. Ketteriä menetelmiä ja Scrumia ei ehkä vielä täysin osata hyödyntää kaikissa yrityksissä, joka ehkä kuvastuu seuraavista valikoidusta haastattelukommenteista:

"Usein projektimallimme muistuttaa 1800-lukulaista höyryjunamallia, jossa on yhdet suorat raiteet käytössä, joita pitkin mennään."

Sovelluskehittäjän kommentti yrityksen projektimallista.

"Aiemmin ei ollut käytössä mitään menetelmää sovelluskehitykseen. Hommat piti saada parempaan kuosiin."

Laatupäällikkö totesi Scrumiin siirtymisestä.

"Scrum lähti alun perin yhden miehen ristirekstä, ja muutos on kestänyt kauan."

Kehityspäällikkö Scrumiin siirtymisestä.

"Ei kiinnosta kunhan tulee valmista. Älkää kertoko softakehityksestä vaan valmiista tuotteesta."

Asiakkaan kommentti kun heille esiteltiin Scrum-menetelmää.

"Ettekö te tuotakaan heti sitä mitä me halutaan?"

Asiakkaan kommentti iteratiiviseen kehitysmalliin.

"Älkää häiritkö meitä ennen kuin on valmista."

Asiakkaan kommentti palaveripyntöihin.

"Ennen nakitettiin ihmisiä, nyt tiimejä ja tuoteomistajaa."

Kehityspäällikkö Scrumin varjopuolista.

"Scrum auttaa projektien työntekijöiden valvonnassa."

Johdon kommentti Scrumista.

"Meillä on Scrumiin hiljainen johdon hyväksyntä, johto ei haittaa meitä – ainakaan merkittävästi."

ScrumMaster toteaa johdon suhtautumisesta Scrumiin.

"Tietokonenörtit eivät aina puhu edes vieruskaverinsa kanssa."

Sovelluskehittäjä kertoo kommunikaatiovaikeuksista Scrum-palaverissa.

Tahtoisin kiittää professori Tommi Mikkosta loistavasta ohjauksesta, tutkija Veli-Pekka Elorantaa hienoista kommenteista sekä avusta haastatteluissa ja professori Ilkka Haikalaa tärkeistä neuvoista tämän työn suhteen. Kiitokset kuuluvat myös niille kaikille yrityksille, jotka osallistuivat innokkaasti haastattelututkimukseen. Toivon että nämä tulokset päätyvät ahkeraan käyttöön yrityksissä.

Suuri kiitos kuuluu myös koko perheelleni sekä avopuolisolleni Sinille tärkeästä kannustuksestanne opiskelujeni aikana.

Tampereella 31.12.2010

Jyri Vuorinen
jyri.vuorinen@gmail.com

SISÄLLYS

1.	Johdanto	1
2.	Ketterät ohjelmistotuotantomenetelmät	2
2.1	Ketterien menetelmien tausta	2
2.2	Erot perinteisiin menetelmiin	3
2.3	Ketterien menetelmien hyödyt	4
2.4	Ketterien menetelmien ongelmat	5
2.5	Scrum	6
2.5.1	Toimintaperiaate	7
2.5.2	Roolit	7
2.5.3	Tapaamiset	8
2.5.4	Prosessin ohjaaminen	9
2.5.5	Käytännöt	10
2.5.6	Sertifiointi	10
3.	Scrum maailmalla	11
3.1	Tiedonlähteet, hakustrategia ja esikäsittely	11
3.2	Laadunarviointi	12
3.3	Tiedonkeruu	12
3.4	Tutkimustulokset	14
3.4.1	Scrumin hyödyt	14
3.4.2	Scrumin ongelmat ja haasteet	15
3.4.3	Tulosten yhtäläisyydet muiden ketterien menetelmien kanssa	16
3.4.4	Luotettavuus	17
4.	Scrum Pirkanmaalla	18
4.1	TTY:n Scrum-koulutukset	18
4.2	Tutkimuksen tavoitteet	19
4.3	Tutkimusympäristö	20
4.4	Tutkimusmenetelmä	20
4.5	Kyselyn sisältö	20
4.6	Nokia-testi	21
5.	Kyselytutkimuksen tulokset	22
5.1	Scrumin käyttö	22
5.1.1	Nokia-testin tulokset	23
5.1.2	Nokia-testin tuloksien vertailu	25
5.2	Scrumin soveltaminen	30
5.2.1	Scrum-koulutus	30
5.2.2	Asiakkaiden suhtautuminen	30
5.2.3	Organisaatiokulttuuri	31
5.2.4	Tiimit	31
5.2.5	Iteraatiot	32
5.2.6	Testaus	33

5.2.7	Määrittely	33
5.2.8	Tuoteomistaja	33
5.2.9	Tuotteen työlista	34
5.2.10	Työmääräarviointi	34
5.2.11	Burndown-kaavio	35
5.2.12	Häiriöt	35
5.2.13	Retrospektiivit	35
5.2.14	Tiimin työympäristö	36
5.2.15	Hajautus	36
5.2.16	Alihankinta	37
5.2.17	Dokumentointi	37
5.2.18	Scrum ja laatu järjestelmät	37
5.2.19	Valmiin tehtävän käsite	38
5.2.20	Käytetyt työkalut	38
5.3	Scrum-menetelmän hyödyt	39
5.4	Scrum-menetelmän ongelmat ja haasteet	40
5.5	Scrum-suosituksia	42
5.6	Tulosten luotettavuus	43
6.	Yhteenveto	44
	Lähteet	46
	Liite 1 Tiedonkeruukaavake artikkeleille	51
	Liite 2 Tutkitut artikkelit	52
	Liite 3 Haastattelukysymykset	54
	Liite 4 Nokia-testin kysymysten pisteytys	56

1. JOHDANTO

Ketterät ohjelmistotuotantomenetelmät ovat olleet jo pitkään käytössä ohjelmistoteollisuudessa. Näiden menetelmien rynnistys lähti liikenteeseen vuonna 2001 julkaistusta Agile-manifestista [1], joka pyrki vastaamaan kasvavan ohjelmistoteollisuuden kevyempien ja ketterämpien toimintamenetelmien tarpeeseen [2]. Ketterät menetelmät syntyivätkin vastauksena aiempien suunnittelupohjaisten menetelmien riittämättömyydestä kohdata ja käsitellä muuttuvia toimintaympäristöjä [3]. Sitten erilaisista ketteristä menetelmistä ja käytännöistä on kirjoitettu paljon kirjallisuutta ja artikkeleita. Ketterien menetelmien puolestapuhujat pitävät niiden keskeisinä etuina yksinkertaisuutta ja nopeutta [4]. Kehitystyössä tiimit voivat keskittyä vain ohjelmistojen olennaisiin toiminnallisuuksiin, jotka voidaan toteuttaa nopeasti, sekä palautteen keräämiseen ja liiketoiminta- ja teknologiaympäristön muutoksiin [5, 6]. Scrum on yksi suosituimmista iteratiivisista ketteristä menetelmistä. Scrum-menetelmässä tiimit työskentelevät iteratiivisesti ja inkrementaalisesti. Scrum-tiimissä jäsenillä on erilaisia rooleja, lisäksi olennaisia ovat myös erilaiset kokoukset, työlista ja työn jakaminen osiin. Luvussa 2 esitellään Scrumia ja muita ketteriä menetelmiä tarkemmin.

Tässä diplomityössä tavoitteena on selvittää ketterien menetelmien, ja erityisesti Scrumin, käytön hyötyjä sekä ongelmia ja haasteita. Tutkimustuloksia ketterien menetelmien käytöstä on saatu jonkin verran, mutta tulokset eivät ole kovin kattavia. Työssä lisätään yleistä tietoutta siitä, miten Scrumia käytetään yritys-elämässä, sekä tunnistetaan menetelmän hyötyjä ja heikkouksia. Työssä kootaan ensin systemaattisella artikkeliarvioinnilla Scrum-menetelmän käytöstä saadut aiemmat tutkimustulokset yhteen luvussa 3.

Diplomityössä kartoitetaan lisäksi haastattelututkimuksella millaisia ketteriä käytäntöjä Pirkanmaalaiset pienet ja keskisuuret ohjelmistoyritykset käyttävät. Lisäksi selvitetään miten paljon Scrumin käytäntöjä oli omaksuttu ja kuinka tarkasti sen menettelytapoja noudatetaan. Yksi tavoite on myös etsiä ja tunnistaa Pirkanmaalaisista ohjelmistoyrityksistä Scrum-menetelmällä saavutettuja hyötyjä sekä havainnoida menetelmän ongelmia ja haasteita. Haastattelututkimuksen taustat ja tarkempi kuvaus on luvussa 4. Tuloksia esittelevässä luvussa 5 kuvataan tarkempien tulosten lisäksi myös yritysten omia hyviä ketteriä käytänteitä.

2. KETTERÄT OHJELMISTOTUOTANTOMENETELMÄT

Ketterät menetelmät syntyivät vastauksena aiempien suunnittelupohjaisten menetelmien riittämättömyyteen kohdata ja käsitellä muuttuvia toimintaympäristöjä. Suunnittelupohjaisista menetelmistä tunnetuin on vesiputousmalli, jossa on selkeä jako määrittely-, suunnittelu-, toteutus-, testaus- ja ylläpitovaiheisiin [3]. Suunnittelupohjaiset menetelmät usein luottavat määrittelyn ja suunnittelun täydellisyyteen, ja monesti virheet ja väärinkäsitykset kertautuvat prosessin edetessä ja toimintaympäristön muuttuessa.

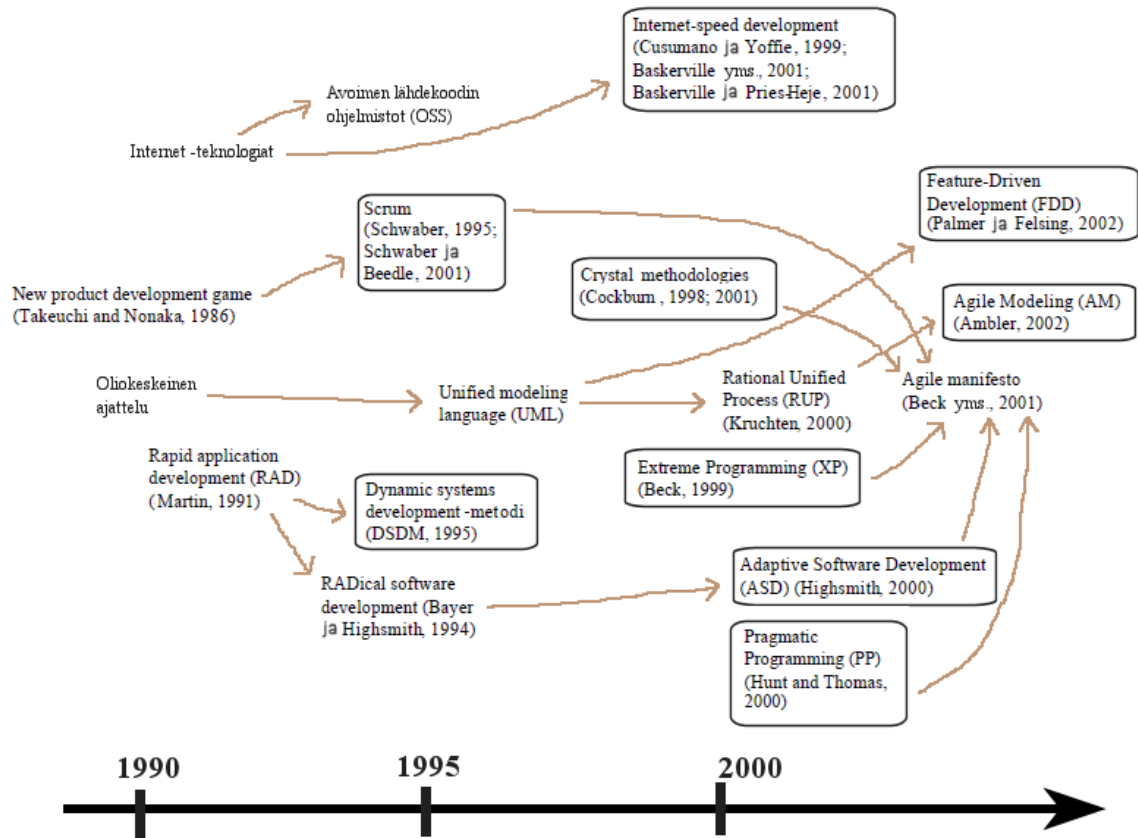
2.1 Ketterien menetelmien tausta

Monien ketterien menetelmien synty ajoittuu 1990-luvun taitteeseen, ja eräänlainen kehityksen huipentuma saavutettiin Agile-manifestin syntyessä vuonna 2001 [1]. Manifestin tarkoituksena oli luoda yhteistä pohjaa ketterille menetelmille ja siten edistää niiden leviämistä. Manifestin keskeinen sisältö oli nostaa yksilöiden, vuorovaikutuksen, toimivan sovelluksen, asiakasyhteistyön ja muutokseen reagoinnin arvostusta. Ketterät menetelmät pyrkivät tunnistamaan toimintaympäristön muutokset ja vastaamaan niihin saman tien. Tämä on mahdollista esimerkiksi iteratiivisen kehitysprosessin kautta. Ketterä kehitys on myös erityisen ihmiskeskeinen näkemys ohjelmistokehitykseen [7]. Sittenkin ketterät menetelmät ovat poikineet paljon suosiota ja kirjallisuutta, ja erilaisia menetelmiä on alettu aktiivisesti soveltaa ohjelmistoteollisuudessa. Ketterien menetelmien kehittymiseen ovat vaikuttaneet mm. Internetin kehitys, avoimen lähdekoodin ohjelmistojen suosio, oliokeskeisen ajattelun leviäminen ja UML:n kehitys [2]. Kuvassa 2.1. on esitetty ketterien ohjelmistotuotantomenetelmien kehittyminen ja menetelmien keskinäiset suhteet. Laatikoilla on merkattu varsinaiset menetelmät, ja lisäksi kaaviossa näkyvät myös muut keskeiset vaikuttimet.

Alistair Cockburn kuvaa ketterien menetelmien perusteoksessaan ohjelmistokehitystä ”yhteiseksi keksintöjen ja kommunikaation peliksi” ("cooperative game of invention and communication") [8]. Ketterä menetelmä tarkoittaa sekä kevyttä että riittävää menetelmää, jossa keveys on keino säilyä vauhdissa ja riittävyys tarkoittaa pelissä mukana pysymistä. Ketterä menetelmä pyrkii joka tilanteessa vastaamaan kysymykseen ”kuinka säilyttää ketteryys tässä tilanteessa?” [8].

Ketterät menetelmät ovat yleistyneet nopeasti. Kysely 4770 IT-osaajalle 91:ssä eri maassa vuonna 2010 paljasti että 92 % vastaajista oli tietoisia tai erittäin tietoisia ketteristä menetelmistä. Vastaajista 90 % työskenteli yrityksessä, jossa käytettiin

ketteriä menetelmiä jossain määrin ja lisäksi 40 % työskenteli yrityksessä, jossa ketteriä menetelmiä oli käytetty vähintään kahden vuoden ajan. Scrum-menetelmä tai sen variantit olivat suosituimpia yksittäisistä käytetyistä ketteristä menetelmistä [9].



Kuva 2.1. Agile-menetelmien kehityskaavio [2]

2.2 Erot perinteisiin menetelmiin

Ketteristä menetelmistä on kirjoitettu paljon teoksia, joissa korostetaan lisääntyneitä asiakastytyvyyttä, organisaation kehittymistä sekä kehittäjien moraalien ja lopputuotteen laadun paranemista. Nämä hyödyt kumpuavat perinteisten ohjelmistotuotantomenetelmien ja ketterien menetelmien perustavanlaatuisista eroavaisuuksista. Perinteinen ohjelmistotuotanto kuvaa ongelmia yleensä deterministisesti ja olettaa jollain tapaa stabiilin ympäristön. Ketterissä menetelmissä taas ongelmat nähdään kompleksisina ja ympäristö vaikeasti ennustettavana. Perinteisissä menetelmissä ongelmanratkaisun tavoitteena onkin ollut optimointi ja ketterissä menetelmissä taas sopeutuminen ja reagoitakyky. Lisäksi prosessissa tapahtuva oppiminen on ketterissä menetelmissä siirtynyt perinteisestä yksisilmukkaisesta adaptiivisuudesta kaksisilmukkaiseen produktiiviseen ajatteluun. Ketterää suunnitteluprosessia ei haluta formalisoida tiukkoihin sääntöihin ja askeliin perustuvaksi, vaan se nähdään iteratiivisena ja kokeilevana prosessina. Perinteisen ohjelmistotuotannon maailmassa johtajan käsissä on tiukka kontrolli ja ohjaus. Myös

konflikteja pyritään välttämään, ja tavoitteena on ensin suunnitella kunnolla ja toteutetaan vasta sitten. Ketterät menetelmät kannustavat yhteistyöhön sekä kommunikaatioon ja perustuvat konfliktitilanteiden käsittelyyn, jossa johtaja toimii tiimin fasilitoijana. Lisäksi suunnittelua ja toteutusta ei voi erottaa erillisiksi vaiheikseen vaan ne nähdään yhdessä kehittyviksi osiksi prosessia. Taulukossa 2.2. on kuvattu näitä ketterien menetelmien eroavaisuuksia perinteisiin ratkaisuihin nähden. [10]

Taulukko 2.2. Perinteinen ja ketterä näkökulma ohjelmistotuotantoon

	Perinteinen ohjelmistotuotanto	Ketterä ohjelmistotuotanto
Suunnitteluprosessi	Harkittua ja formaalia, peräkkäiset askeleet, erillinen määrittely ja toteutus, sääntöihin perustuva.	Iteratiivinen ja kokeileva, tietämys ja toiminta erottamattomia, ei tiukkoja formaaleja sääntöjä.
Tavoite	Optimointi.	Sopeutuminen, joustavuus, reagoitokyky.
Lähestymistapa ongelmanratkaisuun	Parhaiden keinojen valinta saavuttaakseen halutun lopputuloksen, hyvin suunniteltujen ja formalisoitujen toimenpiteiden avulla.	Kokeilun ja itsehavainnoinnin kautta oppiminen, jatkuva ongelman ja ratkaisun uudelleenmuotoilu.
Kuva toimintaympäristöstä	Vakaa, ennustettava.	Turbulentti, vaikea ennustaa.
Oppimistapa	Yksisilmukkainen / adaptiivinen oppiminen.	Kaksisilmukkainen / produktiivinen oppiminen.
Ominaispiirteet	Kontrolli ja ohjaus. Konfliktien välttäminen. Formalisoi innovaatiot. Johtaja kontrolloi. Suunnittelu edeltää implementointia.	Yhteistyö ja kommunikaatio – integroivat henkilöiden maailmankuvia. Ottaa vastaan konflikteja. Kannustaa tutkimiseen ja luovuuteen, opportunistinen. Johtaja on fasilitoija. Suunnittelu ja implementointi ovat erottamattomia ja kehittyvät yhdessä.
Logiikka	Tekninen, funktionaalinen	Olellaiseen keskittyminen.
Teoreettiset ja/tai filosofiset juuret	Looginen positivismi, tieteellinen suunnitelmallisuus.	Action learning–teoria, Deweyn pragmatismi, fenomenologia.

2.3 Ketterien menetelmien hyödyt

Ketterien menetelmien hyötyjä on myös tutkittu. Tore Dybån ja Torgeir Dingsøyryn ketterien menetelmien systemaattisessa tutkimuksessa tunnistettiin useita menetelmien hyviä puolia [11]. Kommunikaation sekä palautteenannon todettiin lisääntyneen ketteriä menetelmiä käyttämällä. Myös säännölliset tapaamiset tiimin kesken auttoivat yhteistyössä ja antoivat kaikille paremman kuvan työn etenemisestä. Lisäksi asiakkaat

nähtiin arvokkaina resursseina, ja asiakkaat itse kokivat pääsevänsä aktiivisesti osallistumaan prosessiin. Prosessin kontrolli, läpinäkyvyys ja laatu kasvoivat jatkuvan integroinnin ja hallittavan kokoisten tehtävien ansiosta. Menetelmät nähtiin myös hyvin teknislähtöisinä, joka motivoi insinöörikuntaa. Yksittäisistä käytännöistä pariohjelmoinnin todettiin johtavan tehokkaaseen oppimiseen, mikäli pareja vaihdetaan säännöllisesti. Työelämän laatu parani työympäristön kehittyessä mukavampaan ja vastuullisempaan suuntaan. Tulokset ketterien menetelmien hyvistä puolista on listattu taulukkoon 2.3. [12]. Dybån ja Dingsøyirin tutkimuksessa löytyi ja analysoitiin kuitenkin pääasiassa XP-menetelmään liittyvää tietoa. Lisäksi tiimit eivät olleet kovinkaan kokeneita ketterien menetelmien käyttäjiä ja projektien koot olivat pääsääntöisesti hyvin pieniä. Tutkimuksessa 73 % tuloksista saatiin projekteista, joissa työntekijöillä oli alle vuosi kokemusta ketterästä ohjelmistokehityksestä. Lisäksi suurimmassa osassa tutkimuksia tiimien koko oli alle kymmenen henkeä sekä projektin kesto oli vuoden tai alle.

Taulukko 2.3. Ketterien menetelmien havaittuja hyötyjä

Tunniste	Havaittu hyöty
H01	Parempi tietämyksen siirtyminen paremman kommunikaation ja säännöllisen palautteen ansiosta.
H02	Sovelluskehittäjät näkevät asiakkaat hyvin arvokkaina, koska kehittäjät voivat keskustella heidän kanssaan ja saada palautetta varhaisessa vaiheessa.
H03	Pariohjelmointi johtaa oppimiseen, jos pareja vaihdetaan säännöllisesti.
H04	Prosessin kontrolli, läpinäkyvyys ja laatu ovat kasvaneet jatkuvan integroinnin ja pienten hallittavissa olevien tehtävien ansiosta.
H05	Menetelmä on hyvin teknislähtöinen, joka saa insinöörit motivoitumaan paremmin.
H06	Pienet tiimit ja säännölliset tapaamiset kasvotusten auttavat yhteistyössä, sekä saamaan paremman kuvan kehitysprosessista.
H07	Sosiaalinen työympäristö nähdään rauhallisena, luotettavana, vastuullisena ja työelämän laatua edistävänä.
H08	Asiakkaat arvostavat aktiivista osallistumista prosessiin, koska näin he voivat kontrolloida projektia sekä kehitysprosessia, ja heidät pidetään ajan tasalla.
H09	Sovelluskehittäjät pitävät työympäristöä mukavana, ja he ovat mielestään tuottavampia käyttäessään pariohjelmointia.
H10	Opiskelijaohjelmoijat pitävät tuottamansa koodin laatua parempana pariohjelmoinnissa.

2.4 Ketterien menetelmien ongelmat

Ketterät menetelmät eivät kuitenkaan ole saaneet täysin varauksetonta vastaanottoa ohjelmistoteollisuudessa, ja niihin on väitetty liittyvän myös joitain isompia ja pienempiä ongelmia. Samaisessa Tore Dybån ja Torgeir Dingsøyirin ketterien menetelmien systemaattisessa tutkimuksessa tunnistettiin menetelmiin liittyviä ongelmia [11]. Ketterissä menetelmissä tapahtuvan jatkuvan testauksen todettiin vaativat paljon enemmän ponnisteluja ja integroidun testausympäristön luominen on

haastavaa. Myöskään aina ohjelmiston arkkitehtuurin suunnitteluun ei kiinnitetty tarpeeksi huomiota iteratiivisessa kehityksessä, joka johti huonoihin suunnitteluratkaisuihin. Ketterien menetelmien skaalautuvuudessa todettiin myös olevan isoja ongelmia. Eri tiimien välinen kommunikaatio saattaa olla heikkoa, vaikka tiimin sisäinen kommunikaatio toimisikin. Lisäksi tiimin jäsenten tulee olla päteviä moniosaajia pystyäkseen ketterään toimintaan. Johtajat eivät usein ymmärtäneet rooliaan tiimin työskentelyn mahdollistajana, vaan kokivat tiimin itsenäisyyden pelottavana. Asiakkaan suurempi rooli kehitystyössä saattaa olla myös hyvin stressaavaa. Tutkimustulokset on koottu taulukkoon 2.4. [12]. Tässäkin tapauksessa suurin osa tuloksista koski ainoastaan XP-menetelmää.[11]

Taulukko 2.4. Ketterien menetelmien havainnoituja ongelmia

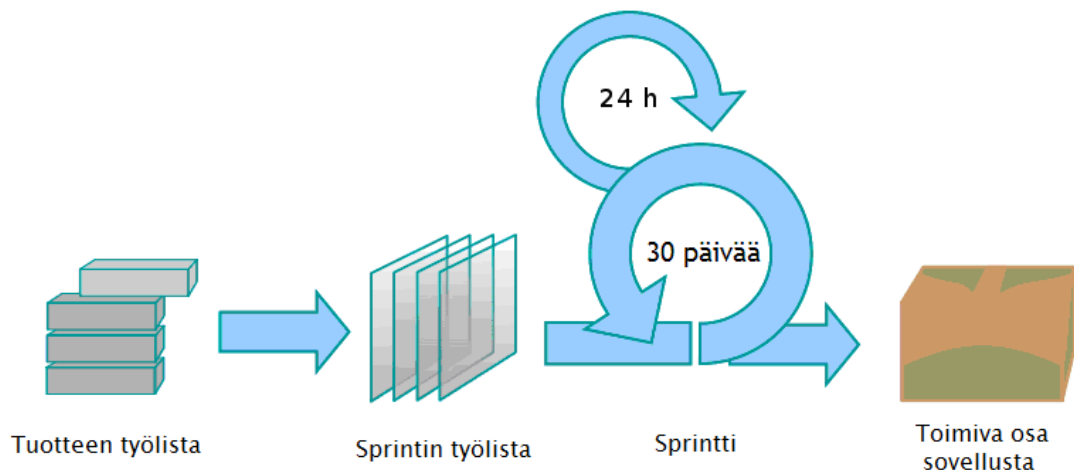
Tunniste	Havaittu ongelma
O01	Jatkuva testaus tarvitsee paljon enemmän ponnisteluja, koska integroidun testausympäristön luominen eri alustoille ja järjestelmien välille on hankalaa.
O02	Arkkitehtuuriseen suunnitteluun ei kiinnitetä tarpeeksi huomiota ketterässä kehityksessä, joka johtaa huonoihin suunnitteluratkaisuihin.
O03	Ketterä kehitys ei skaalaudu hyvin.
O04	Pariohjelmointi nähdään uuvuttavana ja tehottomana.
O05	Tiimin jäsenien tulee olla erittäin päteviä menestyäkseen ketterässä kehityksessä.
O06	Tiimit ovat hyvin koherentteja, mikä tarkoittaa että kommunikaatio tiimin sisällä toimii, mutta tiimien välillä kärsii.
O07	Johtajat tarvitsevat riittävää koulutusta, koska pitävät tiimin itsenäistä työskentelyä pelottavana.
O08	Implementointi alkaa varhain, ja tekniset ongelmat nousevat esiin liian aikaisin johdon näkökulmasta.
O09	Paikan päällä työskentelevän asiakkaan täytyy sitoutua koko kehitysprosessiin, joka stressaa heitä huomattavasti.
O10	Opiskelijan näkökulmasta pariohjelmointi ei toimi, jos toinen on paljon kokeneempi kuin toinen.

2.5 Scrum

Scrum on iteratiivinen ja inkrementaalinen ketterä toimintatapa, jossa itseohjautuva tiimi työskentelee yhteistä päämäärää kohti. Scrum-menetelmä sisältää monia erilaisia käsitteitä, kuten tietynlaisia rooleja tiimin jäsenille, erilaisia tapaamisia ja prosessin seurantaan sekä ohjaamiseen liittyviä tehtäviä, tuotoksia ja toimintatapoja. Scrum yleisenä projektinhallinnan menetelmänä ei rajoitu pelkästään ohjelmistotuotantoon, mutta tässä työssä käsitellään Scrumia ainoastaan ohjelmistotuotannon kontekstissa. Tämän kohdan ja alakohtien määritelmät perustuvat Schwaberin ja Beedlen kirjoittamaan Scrum-menetelmän perusteokseen *Agile Software Development with Scrum* [13].

2.5.1 Toimintaperiaate

Scrumissa tuotteen työlistasta (*en. Product Backlog*) otetaan aina osa toiminnallisuuksista ja tehtävistä, josta sitten muodostetaan sprintin työlista (*en. Sprint Backlog*). Yksi sprintti kestää yleensä 30 päivää, jonka aikana tiimi toteuttaa sprintin työlistassa olevat tehtävät. Sprintin sisällä edistystä seurataan päivittäin, ja 30 päivän kuluttua tuloksena on toimiva osa sovellusta. Tätä iteratiivista prosessia jatketaan kunnes tuotteen päätetään olevan riittävän valmis. Kuvassa 2.5. on esitetty Scrum-menetelmän perusrakenne ja käsitteet.



Kuva 2.5. Scrum-menetelmän iteratiivisuus ja keskeiset käsitteet [14]

Scrum-menetelmä perustuu empiriseen prosessikontrolliteoriaan. Tämän teorian mukaisen menetelmän keskeinen elementti on jatkuva palautteen anto ja käsittely, jota toteutetaan jokaisen iteraation päätteeksi. Tavoitteena on parantaa sekä itse tuotetta että prosessia. Tämän jatkuvan prosessin tarkastelun ja parantelun tuloksena mahdollisuudet uusin tilanteisiin sopeutumiseen kasvavat. [15]

2.5.2 Roolit

Scrum-tiimi koostuu henkilöistä, joilla on erilaiset roolit. **ScrumMaster** on rooli, joka on vastuussa Scrum-prosessista, sen noudattamisesta ja tiimin tuottavuudesta. ScrumMasterin tehtävä on poistaa tiimin työskentelyä haittaavia esteitä, auttaa heitä työskentelemään yhteisen päämäärän eteen ja estää muita häiritsemästä tiimin työskentelyä. Häiriöiden poistaminen on tärkeää, koska jatkuvat keskeytykset tiimin estävät tehokkaan ja tuottavan työskentelyn. ScrumMasterilla ei kuitenkaan ole suoraa määräysvaltaa tiimiin, vaan tiimi tekee päätöksensä itsenäisesti.

Tuoteomistaja (*en. Product Owner*) on henkilö, joka vastaa projektin kannattavuudesta priorisoimalla tuotteen työlistan. Hän toimii myös projektin sidosryhmän henkilöiden edustajana ja ohjaa iteraation alussa olevaa suunnittelupalaveria sekä hyväksyy edellisen iteraation tulokset. Sidoryhmällä

tarkoitetaan niitä henkilöitä, joilla on intressejä projektin suhteen. He voivat olla esimerkiksi projektin rahoittajia, käyttäjiä tai muuten vaan projektin vaikutuspiirissä.

Tiimi on itseohjautuva ryhmä ihmisiä, joka on vastuussa itsensä johtamisesta. Heidän tavoitteenaan on tuottaa toimiva sovellus jokaisen sprintin päätteeksi. Tiimin optimikoko on yleensä 5-9 henkeä. Itseohjautuvuus tarkoittaa sitä, että tiimin sisällä ei ole tiukkaa tehtäväjakoja, vaan kaikki ovat sitoutuneet tekemään parhaansa sprintin lopputuloksen eteen, ja voivat toimia missä tehtävässä tahansa joka liittyy projektiin. Joka tapauksessa yksittäisen tiimin jäsenen tehtävät ovat tiimin itsensä päätettävissä.

2.5.3 Tapaamiset

Scrum-menetelmän iteratiivinen toiminta rakentuu erilaisten palaverien ja tapaamisten varaan. **Päivittäinen Scrum** (*en. Daily Scrum*) on tiimin jokapäiväinen lyhyt, noin 15 minuutin mittainen, tilannepalaveri. Tavoitteena on kommunikoida koko tiimille ja ScrumMasterille työn edistystä ja tuoda julki mahdolliset esteet ja hankaluudet työssä. Näin tiimin jäsenet pystyvät parhaiten auttamaan toisiaan ja työtehtävät pysyvät paremmin samassa tahdissa. Nämä palaverit pidetään yleensä seisomapalavereina (*en. Stand-up Meeting*), niin että osallistujat eivät saa istuutua. Ideana on pitää palaverin kesto näin lyhyenä. Palaverin aikana jokainen tiimin jäsen vastaa kysymyksiin:

1. Mitä olet tehnyt edellisen Scrum-kokouksen jälkeen?
2. Mitä aiot tehdä seuraavaan Scrum-kokoukseen mennessä?
3. Mikä on hankaloittanut työtäsi?

Sprintin suunnittelutapaaminen (*en. Sprint Planning meeting*) on kahdeksan tunnin mittainen tapaaminen, joka aloittaa sprintin. Sprintin suunnittelu jakaantuu kahteen neljän tunnin osioon. Ensimmäisessä osiossa tuoteomistaja esittää priorisoidun työlistan tiimille. Sitten tiimi ja tuoteomistaja yhdessä selvittävät kuinka paljon toiminnallisuuksia tiimi voi seuraavan sprintin aikana toteuttaa. Tiimi sitoutuu näiden toiminnallisuuksien toteuttamiseen kokouksen päätteeksi. Toisessa osiossa tiimi suunnittelee yksityiskohtaisesti kuinka se aikoo toiminnallisuudet toteuttaa, eli suunnittelee sprintin työlistan.

Sprintin katselmointi (*en. Sprint Review meeting*) on neljän tunnin mittainen tapaaminen kunkin sprintin päätteeksi, jossa tiimi yhdessä demonstroi tuoteomistajalle ja sidosryhmäedustajille mitä se on saanut kyseisessä sprintissä aikaan. Tapaamisessa saa esitellä ainoastaan täysin valmiita toiminnallisuuksia.

Retrospektiivi (*en. Retrospective meeting*) on kolmen tunnin mittainen tapaaminen, jossa tiimi yhdessä ScrumMasterin kanssa keskustelee juuri päättyneestä sprintistä. Tarkoituksena on päättää asioista joita seuraavaan sprinttiin mennessä muutettaisiin, jotta työstä tulisi miellyttävämpää ja tuottavampaa.

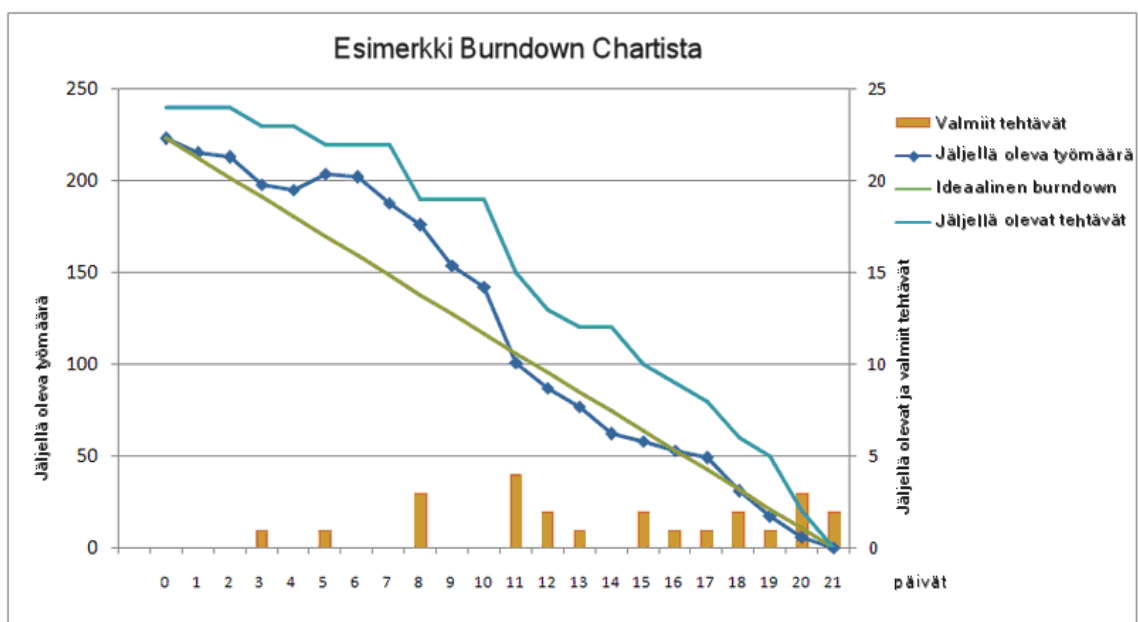
Scrum-of-Scrums on tärkeä palaveri Scrumin skaalaamisen kannalta. Jokaisesta Scrum-tiimistä yksi henkilö osallistuu palaveriin. Siellä vastataan samoihin Daily Scrumin kysymyksiin, mutta tällä kertaa koko oman tiimin osalta, ja näin ollen palaverissa saadaan isompi kokonaiskuva eri tiimien edistymisestä ja ongelmista.

2.5.4 Prosessin ohjaaminen

Scrum-prosessin ohjaamiseen käytetään erilaisia apuvälineitä. **Tuotteen työlista** (*en. Product Backlog*) on priorisoitu tuotteen työlista, joka sisältää kaikki projektin vaatimukset. Vaatimuksien työmäärä arvioidaan tiimin toimesta suhteessa muihin vaatimuksiin. Tuotteen työlista elää ja muuttuu jatkuvasti liiketoimintavaatimusten muuttuessa tai vaikkapa teknologian kehittyessä. Lista sisältää sekä toiminnallisuusvaatimuksia, ei-toiminnallisia vaatimuksia, sekä myös kaikkia eteen tulevia ongelmia ja muita töitä, joiden ratkaisuun kuluu aikaa.

Sprintin työlista (*en. Sprint Backlog*) on tiimille yhden sprintin ajaksi työstettäväksi valittu työlista. Tiimi on sitoutunut toteuttamaan sprintin työlistan asiat valmiiksi toiminnallisuuksiksi sprintin päätteeksi. Valmiin asian määritelmä voi vaihdella, ja tiimin täytyy itse sopia se yhdessä sidosryhmien kanssa. Määritelmä voi sisältää esimerkiksi ohjelmoinnin, kommentoinnin, yksikkötestauksen, integrointitestauksen ja dokumentoinnin.

Burndown-kaavio (*en. Burndown Chart*) kertoo sprintin, julkaisun tai tuotteen jäljellä olevan työmäärän. Kaavio saadaan muodostettua sprintin työlistasta tai tuotteen työlistasta; vaaka-akselilla on aika ja pystyakselilla jäljellä oleva työ. Esimerkki Burndown-kaaviosta on annettu kuvassa 2.6. Joka päivä päivitetävästä kaaviosta kaikki saavat helposti kuvan sprintin etenemisestä. Scrumissa tiimin **nopeus** (*en. Velocity*) muodostuu siitä, kuinka paljon työtä he pystyvät yhdessä sprintissä tekemään. Nopeus näkyy Burndown-kaaviosta, ja se tasaantuu yleensä sprinttien kuluessa. Nopeuden avulla tulevia sprinttejä voidaan suunnitella ja ennustaa tuotteen valmistumispäivämääriä.



Kuva 2.6. Esimerkki Burndown-kaaviosta [16]

2.5.5 Käytännöt

Pokerisuunnittelu (*en. Poker Planning*) on Scrumissa käytetty konsensushenkinen tapa tehtävien työmäärän arviointiin. Sprintin suunnittelutapaamisissa tapahtuva työmääräarviointi tapahtuu siten että jokainen tehtävä käydään yksitellen läpi. Sitten kukin tiimin jäsen esittää erikoispelikortilla oman arvionsa työmäärästä. Pelikortit sisältävät pisteitä (*en. Story Points*) jotka voidaan suhteuttaa toisiinsa yhden tiimin sisällä, mutta ne eivät suoraan kuvaa työtuntimääriä. Mikäli yhteisymmärrystä työmääristä ei näin löydy, pienimmän ja suurimman työmäärän esittänyt saavat perustella omat näkemyksensä, ja tämän jälkeen otetaan uusi korttikierros. Tavoitteena on nopeasti ja tehokkaasti tehdä suurpiirteisiä estimaatteja työmääristä Sprintin suunnittelua helpottamaan.

Käyttäjätarinat (*en. User Story*) on usein ketterissä menetelmissä käytetty tapa kuvata käyttäjän haluamaa toimintoa. Käyttäjätarina on yhden tai useamman lauseen kuvaus, joka kertoo joko selvällä suomella tai käyttäjän toimialan kielellä mitä käyttäjä haluaa aikaansaada. Käyttäjätarinat ovat nopea keino vaatimusmäärittelyyn ilman että tarvitsee kirjoittaa täydellisen formuloituja vaatimusdokumentteja. Tuotteen työlista voi koostua näistä käyttäjätarinoista.

Liiketoiminnallinen kannattavuus, ROI, (*en. Return on Investment*) on keskeinen käsite ja mittari Scrumissa. Tuoteomistaja on vastuussa tuotteen liiketoiminnallisesta kannattavuudesta. Työkalunaan hänellä on tuotteen työlista, jonka vaatimuksien prioriteettia muuttamalla hän maksimoi kannattavuutta organisaatiolle. Työlista muuttuu jatkuvasti liiketoimintaympäristön ja vaatimusten muuttuessa.

2.5.6 Sertifiointi

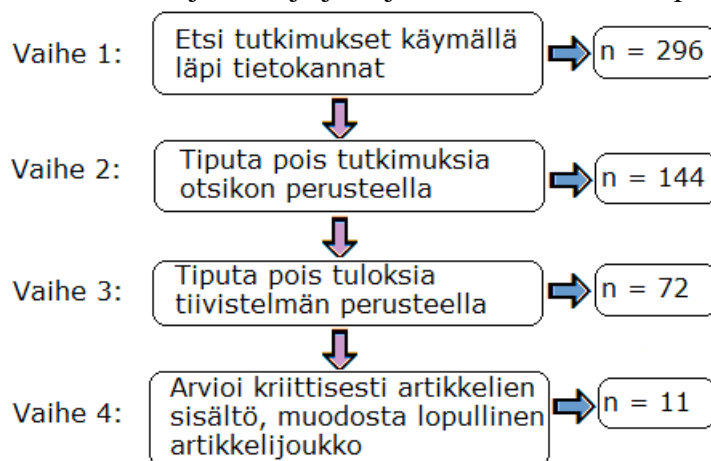
Scrum Alliance -yhdistys on organisoinut maailmanlaajuisen Scrum-koulutuksen. Scrum Alliance luotiin levittämään tietoutta ja ymmärrystä Scrumista, ja he tarjoavat virallisia Scrum-sertifiointeja kouluttajille ja käyttäjille [17]. Sertifioidut Scrum-kouluttajat järjestävät ScrumMaster-, tuoteomistaja- ja Scrum-kehittäjä -kursseja, joilla käydään läpi Scrumin peruseriaatteita ja opastetaan kyseisissä rooleissa toimimisessa. Kurssin käyneet voivat suorittaa yhdistyksen sivuilla testin, jonka perusteella saavat oikeuden kutsua itseään sertifioituksi Scrum-osaajiksi.

3. SCRUM MAAILMALLA

Tässä luvussa tehdään systemaattinen tutkimus- ja artikkeliarviointi Scrum-menetelmän käytöstä saatuihin tutkimustuloksiin. Kirjallisuustutkimuksen tarkoituksena on selvittää Scrumin käyttöä maailmalla ja siitä saatuja tuloksia. Näitä verrataan myöhemmin Pirkanmaan alueen yritysten Scrum-tuloksiin. Tämän systemaattisen arvioinnin pohjana on käytetty mukaillen Tore Dybån ja Torgeir Dingsøyryn ketterien menetelmien systemaattisessa tutkimuksessaan vuonna 2008 käyttämää menetelmää [11].

3.1 Tiedonlähteet, hakustrategia ja esikäsittely

Tutkimuksien ja artikkeleiden löytämistä varten käytiin läpi elektroniset tietokannat *IEEE Xplore* sekä *ScienceDirect* [18, 19]. Tietokannoista haettiin tuloksia hakusanoilla ”Scrum AND software”. Kaikki hakutulokset kerättiin RefWorks-työkalulla omaan tietokantaansa [20]. Tällä haulla saatiin tulokseksi yhteensä 296 erillistä artikkelia. Kuvassa 3.1. on kuvattuna koko artikkelien käsittelyprosessi. Seuraavassa vaiheessa 2 näiden artikkelien otsikot tarkastettiin, ja pudotettiin pois sellaiset, jotka eivät liittyneet Scrumin käyttöön. Mukana oli esimerkiksi lääketieteellisiä artikkeleita vereen liittyvän englanninkielisen scrum-termin vuoksi, rugbyyn vammoja käsitteleviä tutkimuksia ja merentutkimuksen scrum -mallia käsitteleviä artikkeleita. Nyt tutkittavia artikkeleita oli yhteensä 144. Tämän jälkeen vaiheessa 3 siirryttiin tutkimaan artikkelien tiivistelmiä, ja jälleen pudotettiin pois artikkelit, jotka eivät liittyneet Scrumin käytön tuloksiin. Tässä vaiheessa pudotettiin pois esimerkiksi Scrumin opettamiseen ja tulevaisuuden pohdintaan liittyvät artikkelit, uusiin Scrumiin esitettyihin lisämetodeihin ja tekniikoihin keskittyvät artikkelit, sekä muut artikkelit, jotka sisälsivät selvästi pelkkää tekijän omaa pohdintaa. Tämän jälkeen jäljelle jäi 72 artikkelia lähempää tarkastelua varten.



Kuva 3.1. Tutkimusten valintaprosessi

3.2 Laadunarviointi

Vaiheena 4 tutkimuksessa oli laadun arviointi. Arviointi suoritettiin neljän eri kriteerin perusteella. Ravoitteena oli löytää uskottavia, täsmällisiä ja ennen kaikkea relevantteja tuloksia Scrumin käytöstä. Käytetyt laatuksiteerit pohjautuvat osin Tore Dybån ja Torgeir Dingsøyryn systemaattisessa tutkimuksessaan käyttämään laadunarviointikriteereihin, joilla on pohja kvalitatiivisen tutkimuksen laadunarvioinnissa sekä empiirisen ohjelmistotuotannon tutkimuksen hyvissä periaatteissa [11, 21, 22]. Kriteereinä toimivat seuraavat neljä kysymystä, johon kuhunkin vastattiin ”kyllä” tai ”ei” jokaisen tutkimuksen kohdalla:

1. Onko kyseessä tutkimus, vai jonkun henkilökohtaiseen mielipiteeseen tai kertomukseen perustuva raportti?
2. Onko tutkimuksella selkeä tavoite?
3. Onko tutkimuskontekstista selkeää kuvausta?
4. Onko tutkimuksen tuloksena saatu tietoa Scrumin käytöstä?

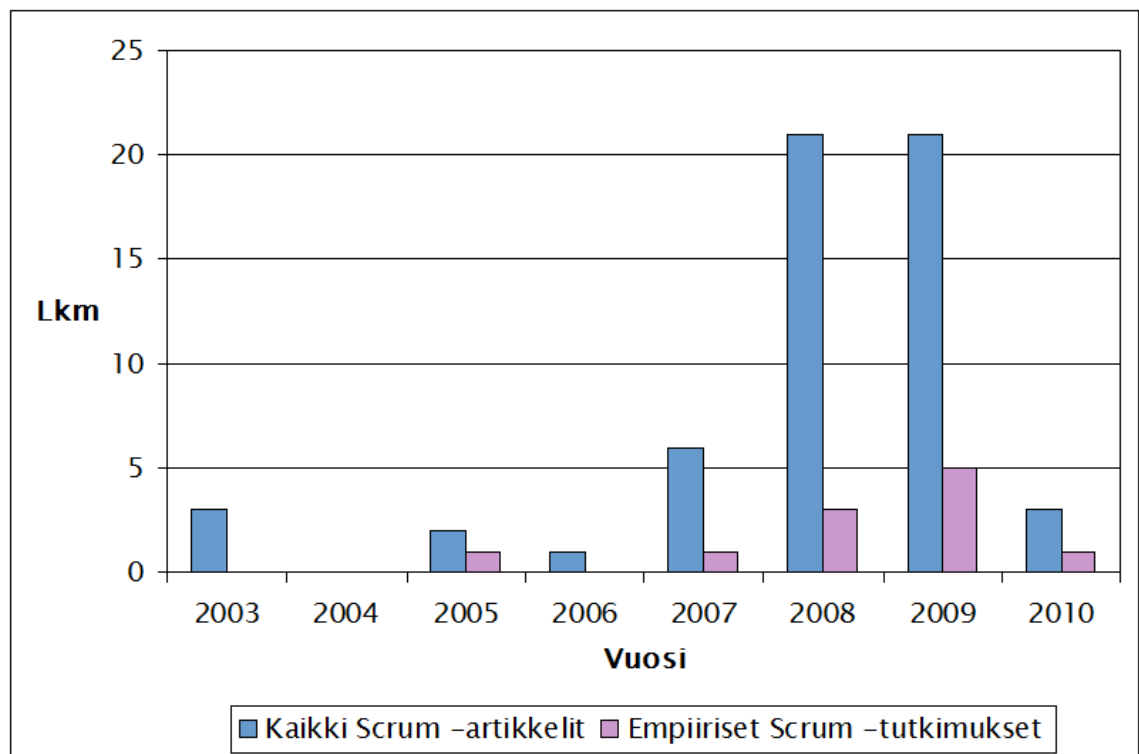
Mikäli johonkin näistä kysymyksistä vastaus oli negatiivinen, ei kyseistä artikkelia otettu jatkovaiheeseen. Vaiheen 4 laadunarvioinnin jälkeen jäljelle jäi 11 artikkelia, jotka toimivat lopullisena tutkimusjoukkona. Arvioinnissa putosi pois artikkeleita uusista konsepteista ja teorioista Scrumin suhteen, raportteja yksittäisten yritysten Scrumin käyttöönotosta omakohtaisina kuvauksina, sekä joitain raportteja, jotka esittelivät huimia tuottavuusparannuslukuja, kertomatta kuitenkaan mitään tutkimusmenetelmästä tai tavoitteista.

3.3 Tiedonkeruu

Kaikista yhdestätoista artikkelista kerättiin tiedot ylös ja taulukoitiin etukäteen määritellyn tiedonkeruukaavakkeen mukaisesti (*liite 1*). Tiedonkeruvaiheessa paljastui, että kolmessa artikkelissa (T07, T07a, T07b) käytettiin lähdemateriaalina samaa case-tutkimusta, joten kyseisistä tutkimuksista huomioitiin ainoastaan yksi. Lopullinen artikkelijoukko pieneni siten yhdeksään tutkimukseen, ja nämä tutkimukset on koottu liitteeseen 2. Taulukoon 3.2. on kerätty tutkimuksia koskevia yleisiä tietoja. Lisäksi kuvassa 3.3. on vertailtu artikkelien julkaisuvuosia, ja vertailun vuoksi mukana on myös vaiheen 3 kaikki Scrumia koskevat artikkelit.

Taulukko 3.2. Yleiskatsaus löydettyihin artikkeleihin

Tunniste	Julkaisu- vuosi	Artikkelin tyyppi	Tutkimus- tyyppi	Tutkittava ryhmä	Projektin kesto	Scrum kokemus
T01	2005	konferenssi- artikkeli	2v case- tutkimus	-	22kk	aloittelijoita
T02	2010	lehtiartikkeli	9kk case- tutkimus	8 henkeä	4800h	aloittelijoita
T03	2008	lehtiartikkeli	kysely- tutkimus	13 yritystä, 35 projektia	-	-
T04	2009	lehtiartikkeli	3v case- tutkimus	3 yritystä, 5 tiimiä	11-33kk	aloittelijoita
T05	2009	konferenssi- artikkeli	kirjallisuus- arviointi	20 artikkelia	-	-
T06	2009	konferenssi- artikkeli	case- tutkimus	4 henkeä	-	kokeneita
T07	2008	konferenssi- artikkeli	case- tutkimus	2 projektia, 18 henkeä	- / 4000h	- / aloittelijoita
T08	2007	konferenssi- artikkeli	kysely- tutkimus	492 henkeä	-	vaihteli
T09	2009	konferenssi- artikkeli	case- tutkimus	3 yritystä	2-10 vuotta	aloittelijoita

**Kuva 3.3.** Scrum-artikkelien julkaisuvuodet

Scrum-artikkelien julkaisuvuosista voi selvästi nähdä, että menetelmää on alettu vasta viime vuosina tutkia toden teolla. Myös kuluvana vuonna 2010 tullaan varmasti näkemään vielä lisää tutkimuksia aiheesta. Suurin osa tutkimuksista oli case-tutkimuksia, mutta mukaan mahtui myös kaksi kyselytutkimusta ja yksi systemaattinen kirjallisuusarviointi. Tutkimuksissa oli mukana monenkokoisia organisaatioita ja

tiimejä, ja projektien koko vaihteli myös paljon. Suurin osa tutkituista Scrum-projekteista käsitteli aloittelevia tiimejä, jotka olivat vasta saaneet Scrum-koulutuksen ja olivat toteuttamassa ensimmäistä tai toista projektiaan. Yleisenä huomiona voi todeta myös että tutkimuksien tavoitteet, kohderyhmät, tutkimusympäristöt ja -menetelmät eivät olleet kovin hyvin kuvattuja, vaan useassa artikkelissa oli näiltä osin suuria puutteita. Scrum-tutkimus on selvästi vielä hyvin aikaisessa vaiheessa, ja olisi tärkeää saada tuloksia myös pidemmän ajan Scrum-kokemuksista.

3.4 Tutkimustulokset

Jo aiemmissakin tutkimuksissa on todettu, että ketterien menetelmien tutkimukselle leimallista on hyvin erilaisista aihealueista tehty, hankalasti keskenään vertailtavissa olevat tutkimukset ja että empiirisellä tutkimuksella on hyvin pieni osuus [23]. Myös tutkimusten metodit perustuvat usein enemmän uskoon ketterän menetelmän toimivuudesta kuin empiiriseen dataan [24]. Nämä asiat olivat selvästi havaittavissa myös tässä selvityksessä. Empiirisiä tutkimuksia löytyi melko vähän, ja ne olivat hyvin sekalaisesti toteutettuja ja dokumentoituja. Löydetyistä artikkeleista pystyi kuitenkin syntetisoimaan tuloksia sekä Scrum-menetelmän hyödyistä että siihen liittyvistä ongelmista ja haasteista. Yksi tärkeä tulos onkin se, että Scrumin käyttöön liittyy sekä positiivisia että negatiivisia puolia. Lisäksi synteessissä tunnistettiin joitain ylemmän tason teemoja, jotka toistuivat tutkituissa artikkeleissa.

3.4.1 Scrumin hyödyt

Scrum-menetelmän käytössä yksi selkeimmistä esiin tulleista hyödyistä oli asiakastyytyväisyyden paraneminen [T01]. Asiakkaat voivat vaikuttaa prosessiin osallistumalla Daily Scrumiin ja sprintin katselmointiin. Menetelmä myös kannustaa tiimiä ja tuoteomistajaa olemaan aktiivisesti yhteydessä asiakkaisiin. Näin asiakkaita huomioidaan paremmin, ja sitä kautta asiakastyytyväisyys paranee. Asiakkaat tarvitsevat kuitenkin yhtä lailla tiimin jäsenien kanssa koulutusta osatakseen ja ymmärtääkseen prosessin.

Myös kehittäjien tyytyväisyys omaan työhönsä parani. Työmoraali parani joustavuuden myötä, ja ylityötuntien dokumentointiin vähenevän (19 % \rightarrow 7 %) [T01]. Scrumin iteratiivisuuden kautta kehittäjät pääsivät paremmin perille siitä mihin pitää keskittyä ja milloin. Tehtävien selkeytyminen ja ylityötuntien väheneminen liittyvät todennäköisesti siihen, että sprintin työlista lyödään lukkoon kunkin sprintin alussa, eikä siihen tehdä enää muutoksia. Kehittäjistä 77 % koki Scrumin hyödylliseksi, ja Scrumia käyttävien tiimien jäsenet raportoivat selkeästi omia odotuksiaan paremmista kokemuksista [T03].

Tiimityöskentelyn suurimpana parannuksena on lisääntynyt kommunikaatio ja yhteinen päätöksenteko verrattuna perinteisiin ohjelmistotuotantomenetelmiin. Yhteistyö asiakkaan kanssa sekä tiimin sisällä jo aikaisemmassa vaiheessa edistää sovelluksen uudelleenkäytettävyyttä ja integroitavuutta. Lisääntynyt kommunikaatio toi

taas mahdolliset ongelmat esiin jo aikaisemmassa vaiheessa [T09]. Palautetta saatiin ja annettiin enemmän, mutta palautteen antamiseen oli myös varattava erillistä aikaa, esimerkiksi pitämällä retrospektiivitilaisuuksia.

Tuotteen työlista nähtiin yksittäisistä Scrumin käytännöistä positiivisimpana, ja sen oikeaoppinen käyttö selkeytti projektia. Myös Daily Scrum keräsi kiitosta. Muita Scrum-käytäntöjä ei artikkeleissa erikseen useinkaan mainittu, ja yhdessä tutkimuksessa todettiin, että muut Scrum-käytännöt ovat hieman tuntemattomia [T03]. Tämä voisi viitata siihen, että osa tutkimuksissa mukana olleista Scrum-projekteista ei kuitenkaan lopulta noudattanutkaan Scrum-prosessia kovin kirjaimellisesti oppien mukaan.

3.4.2 Scrumin ongelmat ja haasteet

Yhtenä keskeisenä ongelmana Scrum-menettelyn käytössä oli kehittäjien pitkälle viety erikoistuminen tiettyihin osa-alueisiin, joka johti siihen, että tiimin sisällä tehtävät olivat hyvin eriytettyjä. Näin ollen tiimityöskentely oli hyvin näennäistä, eikä toiminut ketterällä tavalla. Yksittäiset tiimin jäsenet sitoutuivat omien osa-alueidensa tavoitteisiin, eivätkä tiimin yhteisiin tavoitteisiin. Tämä on taas selkeästi vastoin Scrumin periaatteita. Kehittäjät kokivat monenlaisia resurssiongelmia, esimerkiksi aikaa ei annettu riittävästi tai heidän piti olla osallisia useammassa tiimissä yhtä aikaa. [T04, T07]

Scrum-projektin johtaminen nähtiin haastavaksi. Joissain tutkimuksissa kerrottiin ScrumMasterin välillä käskytävän kehittäjiä entiseen perinteisten ohjelmistotuotantomenetelmien tapaan. Välillä yksittäiset tiimin jäsenet kaappasivat yhteisen päätöksentekovallan sekä tekivät omia ratkaisujaan kertomatta niitä muille tiimin jäsenille [T07].

Yrityskulttuurilla ja johdon sitoutumisella on yksi keskeisimmistä rooleista Scrum-projektin onnistumisen kannalta [T02]. Mikäli yrityskulttuuri ei kannusta tiimityöskentelyyn ja johto ei noudata Scrumin pelisääntöjä, kehittäjien työmoraaali häviää nopeasti, eikä kehitystyö muistuta enää ketterää menetelmää. Esimerkiksi sprintin työlistaa täytyy suojella muutoksilta, joita siihen yritettiin kesken sprintin tehdä tuoteomistajan tai muiden toimesta [T07].

Kuvatut tiimityöskentelyn, johtamisen ja johdon tuen puutteen ongelmat viittaavat siihen, että pystyäkseen soveltamaan Scrumia yrityksen täytyy läpikäydä laaja muutosprosessi [T02]. Yrityksen kannustimet sekä yrityskulttuuri on muokattava siihen suuntaan, että ne kannustavat yhteistyöhön ja avoimuuteen. Johdon tuen on oltava kohdallaan, niin että myös tiimit ja jokainen yksilö pystyy luottamaan Scrumin pelisääntöihin.

Globaalin hajautetun ohjelmistokehityksen kanssa Scrumilla todettiin usein olevan ongelmia. Menettelyn skaalautuvuus ei tuntunut suoraan toimivan, vaan prosessia oli aina muokattava jotenkin, jotta kulttuuri- ja aikaeroja, sekä eri toimipaikkojen tiimien koordinaatiota saatiin jotenkin hallittua. Tähän olivat usein apuna monenlaiset modernit kommunikaatiovälineet ja menetelmät sekä toistuva paikan päällä vieraileminen [T05, T09].

3.4.3 Tulosten yhtäläisyydet muiden ketterien menetelmien kanssa

Edellä kuvatut Scrum-menetelmän havaitut hyödyt sekä ongelmat ja haasteet on koottu taulukoihin 3.4. ja 3.5. Verrattaessa näitä löydöksiä yleisesti ketterien menetelmien vastaaviin taulukoihin 2.3. ja 2.4. havaitaan, että niistä on löydettävissä selviä yhtäläisyyksiä. Nämä yhtäläisyydet on listattuna taulukoiden kolmanteen sarakkeeseen. Esimerkiksi asiakastytyväisyyden paraneminen kommunikaation ja huomioimisen kautta on havaittu myös aiemmissa ketterien menetelmien tutkimuksessa. Myös yleinen työtyytyväisyys lisääntyi, ja aiemmissa tutkimuksissa tämä on liitetty nimenomaan työympäristön paranemiseen. Tiimityöskentelyn paraneminen palautteenannon ja lisääntyneen kommunikaation myötä oli myös tuttua aiemmista ketterien menetelmien tutkimuksesta. Tyytyväisyys tuotteen työlistaan vastaa aiemmin havaittua laadun paranemista pienten hallittavissa olevien tehtävien ansiosta. Scrum-projektin johtamisen haasteet ovat tuttuja myös muista ketteristä menetelmistä. Johtajat tarvitsevat paljon koulutusta kyetäkseen hyväksymään itseorganisoituvat tiimit ja kollektiivisen päätöksenteon. Skaalautuvuus näyttäisi olevan ongelma myös Scrumille niin kuin muillekin ketterille menetelmille.

Taulukko 3.4. Scrum-menetelmän hyödyt

Tunniste	Havaittu hyöty Scrumissa	Ketterät menetelmät yleisesti
HS01	Asiakastytyväisyyden paraneminen lisääntyneen kommunikaation ja asiakkaan huomioimisen kautta.	H08
HS02	Kehittäjien työmoraalin ja työtytyväisyyden paraneminen joustavuuden myötä.	H07
HS03	Ylityötuntien väheneminen.	-
HS04	Tiimityöskentelyn paraneminen yhteisen päätöksenteon, lisääntyneen kommunikaation ja palautteenannon myötä.	H06, H01
HS05	Tuotteen työlistan käyttö selkeytti projektia.	H04

Taulukko 3.5. Scrum-menetelmän ongelmat ja haasteet

Tunniste	Havaittu ongelma / haaste Scrumissa	Ketterät menetelmät yleisesti
OS01	Tiimityöskentely ei toiminut kehittäjien pitkälle viedyn erikoistumisen takia.	-
OS02	Scrum-projektin johtaminen ja kollektiivinen päätöksenteko vaikeaa.	O07
OS03	Yrityskulttuuri ei tue Scrum-käytänteitä.	-
OS04	Sprintin työlistaa täytyi jatkuvasti suojella muutosyrityksiltä kesken sprintin.	-
OS05	Scrum ei suoraan skaalaudu globaaliin hajautettuun ohjelmistokehitykseen.	O03

3.4.4 Luotettavuus

Vaikka Scrumista tehdyt tutkimukset ovatkin hyvin hajanaisia ja erilaisista aiheista, tutkimusten tuloksista on kuitenkin löydettävissä selkeitä yhtäläisyyksiä. Tulokset ovat myös hyvin samansuuntaisia kuin mitä yleisesti ketterien menetelmien hyödyistä ja ongelmista on aiemmin saatu. Näin ollen nyt saatuja tuloksia voi pitää melko luotettavina. Saatujen tulosten tilastollinen luotettavuus kärsii kuitenkin siitä, että löydetty ja tutkittu artikkelimäärä oli niin pieni. Ongelmallista on myös se, että lähes kaikkien tutkimusten kohteena oli aloitteleva Scrum-tiimi. Pitkän aikavälin tuloksia ja laajempia kokemuksia Scrumin käytöstä joudutaan todennäköisesti odottelemaan vielä vuosia. Tutkimuskulttuuri tämän tyyppisissä asioissa on vielä melko nuorta, kuten nähdään myös Scrum-artikkelien julkaisuvuosista (kuva 3.3.). Mahdollista on myös se, että nyt tehdyssä selvityksessä on jäänyt joitain olennaisia artikkeleita huomaamatta ja tarkastelematta, koska monien tutkimusten kuvaukset tutkimuksen tavoitteista, sisällöstä, kohderyhmästä tai tekotavasta olivat puutteellisia. Tämä voi osaltaan vääristää tuloksia.

4. SCRUM PIRKANMAALLA

Tässä työssä toteutettiin haastattelututkimus, jolla kartoitettiin Scrum-menetelmän käyttöä suomalaisissa pienissä ja keskisuurissa ohjelmistoyrityksissä. Lisäksi kerättiin tietoa menetelmän hyödyistä sekä ongelmista ja haasteista. Tarkoituksena on selvittää kuinka orgaanisesti yrityksissä noudatetaan Scrum-menetelmää, mitä osia menetelmästä hyödynnetään, onko Scrumia jouduttu muokkaamaan jollain tapaa, ja mitkä ovat yritysten kokemukset menetelmästä. Tutkimuksen kohderyhmäksi valikoitiin Tampereen teknillisen yliopiston Scrum-koulutuksiin osallistuneiden yritysten ohjelmistoammattilaisia.

4.1 TTY:n Scrum-koulutukset

Tampereen teknillisen yliopiston ohjelmistotekniikan laitoksella on maaliskuusta 2009 lähtien järjestetty Certified ScrumMaster sekä Certified Scrum Product Owner -kurseja [25, 26]. Kurseja on vetänyt James O. "Jim" Coplien, ja niihin on osallistunut työntekijöitä ohjelmistoteollisuudesta sekä TTY:n ohjelmistotekniikan laitokselta. Elokuuhun 2010 mennessä oli järjestetty kaksitoista Certified ScrumMaster -kurssia ja kolme Certified Scrum Product Owner -kurssia. Jokaisella kurssilla oli mukana lähes kaksikymmentä ohjelmistoammattilaista ja opiskelijaa. Kurseihin sisältyi oikeus käyttää Scrum Alliancen myöntämiä Certified ScrumMaster (CSM) tai Certified Scrum Product Owner (CSPO) -nimikettä [17].

Scrum-kurseista saatiin keskimäärin loistavaa palautetta. Palautteisiin vastasi yhteensä 202 henkilöä, joista 184, eli 91 %, suosittelee tai oli jo suositellut koulutusta kollegoilleen. Kokonaisuutena koulutukset saivat erittäin hyvän arvosanan, 4,6 asteikolla 1-5. Myös koulutuksessa käsitellyt asiat koettiin hyödyllisiksi (4,2) ja tiedot uusiksi (3,8). Hieman heikommin arvioitiin mahdollisuuksia soveltaa käsiteltyjä tietoja käytäntöön omissa työtehtävissä (3,7) tai muualla yrityksessä (3,6). Tiedot Scrum-koulutuksien palautteista on koottu taulukkoon 4.1.

Taulukko 4.1. Scrum-koulutuksien palaute

Kysymys	Arvosteluasteikko	Keskiarvo
Arviointi koulutuksesta kokonaisuutena	1-huono, 5-hyvä	4,6
Koulutuksessa käsitellyt tiedot olivat kannaltani	1-vanhoja, 5-uusia	3,8
Koulutuksessa käsitellyt tiedot olivat kannaltani	1-hyödyttömiä, 5-hyödyllisiä	4,2
Käsitellyt tiedot ovat sovellettavissa käytäntöön omassa tehtävässäni	1-ei lainkaan, 5-pääosin	3,7
Käsitellyt tiedot ovat sovellettavissa käytäntöön ja/tai muuten yrityksessäni	1-ei lainkaan, 5-pääosin	3,6
Kestoltaan koulutus oli antiinsa nähden	1-liian pitkä, 5-liian lyhyt	3,2
Koulutus oli esitystavaltaan	1-sekava, 5-selkeä	4,5
Koulutus oli osanottajien aktivoinnin suhteen	1-huono, 5-hyvä	4,5
Jaettu luentomateriaali oli	1-huonoa, 5-hyvää	3,9
Luennointia oli	1-liian vähän, 5-liian paljon	3,1
Ryhmätöiden toimeksiannot olivat	1-epäselviä, 5-selviä	4,2
Harjoitukset olivat	1-tarpeettomia, 5-tarpeellisia	4,4
Keskusteluun varattu aika oli	1-liian lyhyt, 5-liian pitkä	2,8
Kysymys	Vastausvaihtoehdot	Tulos
Suosittelisitko koulutusta kollegoillesi	kyllä / ei	184 / 202 suosittelisi

4.2 Tutkimuksen tavoitteet

Tutkimuksen tavoitteet olivat seuraavat:

- Selvittää millaisia ketteriä käytäntöjä yritykset käyttävät.
- Selvittää miten paljon Scrumin käytäntöjä oli omaksuttu ja kuinka tarkasti sen menettelytapoja noudatetaan.
- Etsiä ja tunnistaa Scrum-menetelmällä saavutettuja hyötyjä.
- Etsiä ja tunnistaa ongelmia ja haasteita ketterissä menetelmissä sekä löytää niihin vastauksia.
- Etsiä, tunnistaa ja levittää hyviä ketteriä käytäntöjä.

Aiemmat tutkimustulokset Scrumin käytöstä eivät ole kovin kattavia. Tämän tutkimuksen tavoitteet voisi siis tiivistää siihen, että lisätään yleistä tietoutta siitä, miten Scrumia käytetään oikeassa yritys-elämässä, sekä tunnistaa menetelmän hyötyjä ja heikkouksia. Osallistuvien yritysten kannalta erityisen tärkeää oli myös julkaista ja levittää hyviä käytäntöjä yrityksistä toisiin.

4.3 Tutkimusympäristö

Tutkimusjoukkona toimi yksitoista Pirkanmaalaista ohjelmistoyritystä. Yritykset valikoitiin Tampereen teknillisen yliopiston Scrum-koulutuksiin osallistuneiden yritysten keskuudesta. Mukaan haluttiin pieniä ja keskisuuria yrityksiä, koska oletettiin että niiden käytännöt saadaan helpommin selvitettyä. Yli kahdensadan hengen yrityksissä tiimejä ja käytäntöjä alkaa olla niin paljon, että yhteenvedon tekeminen käy mahdottomaksi. Keskittymällä pieniin ja keskisuuriin yrityksiin haluttiin myös tukea tutkimuksen yhtä keskeisistä tavoitteista, hyvien käytänteiden leviämistä yrityksistä toisiin.

4.4 Tutkimusmenetelmä

Tutkimusmenetelmäksi valittiin haastattelututkimus. Ennalta valittujen yritysten edustajia lähestyttiin sähköpostitse, jossa kerrottiin tutkimuksen taustasta sekä lähtökohdista ja kysyttiin mahdollisuutta tulla paikan päälle haastattelemaan yrityksen edustajia. Kaikilta yrityksiltä saatiin vastaus, tosin osa ohjasi ottamaan yhteyttä eri henkilöihin yrityksen sisällä. Kaikkien kanssa saatiin kuitenkin haastattelut sovittua, ja osa yritysten edustajista lähetti viestiä jopa eteenpäin uusille yrityksille.

Yhteensä haastateltuja yrityksiä oli yksitoista, ja haastateltuja henkilöitä yhteensä kahdeksantoista. Tyypillinen haastateltava oli nimikkeeltään kehityspäällikkö, laatu- tai projektipäällikkö, ja joissain haastatteluissa oli mukana lisäksi myös sovelluskehittäjä. Haastateltavien valinnalla pyrittiin varmistamaan, että henkilöllä olisi mahdollisimman laaja ja kattava tietämys oman yrityksensä ohjelmistokehitysmenetelmistä ja tiimien toimintatavoista.

Haastattelut suoritettiin yrityksen tiloissa ennalta määritellyn kysymyspatteriston mukaisesti. Ennen varsinaisten haastattelujen aloitusta suoritettiin yksi koehaastattelu, jonka pohjalta joitain kysymyksiä tarkennettiin hieman ja järjestystä muutettiin. Puolistrukturoidut haastattelut etenivät niin, että kaikille haastateltaville esitettiin samat tai lähes samat kysymykset samassa järjestyksessä. Haastattelujen kesto vaihteli yhdestä tunnista kahteen, riippuen siitä kuinka paljon yrityksen ketteristä käytännöistä riitti kerrottavaa. Haastateltavien vastaukset kirjattiin muistiinpanoiksi käsin, ja osa haastatteluista myös nauhoitettiin. Haastattelututkimuksen etuna oli selvästi suhteellisen pieneen haastattelujoukkoon liittyvä monipuolinen aineisto ja syvä informaatio.

4.5 Kyselyn sisältö

Haastattelututkimuksen kysymykset jakautuivat neljään eri osioon:

- *Yleistä*-osiossa kyseltiin yrityksen taustatietoja, kokoa, rakennetta ja kokemusta.
- *Scrum-menetelmän noudattaminen*-osiossa selvitettiin kuinka tarkasti yritys noudattaa Scrum-menetelmää. Osion kysymykset noudattivat Nokia-testiä, joka on kuvattu seuraavassa kohdassa [27].

- *Scrumin implementointi*-osiossa selvitetiin tarkemmin hyviä ja huonoja puolia yksittäisistä Scrumin käytännöistä, ja näiden käyttöä yrityksessä.
- *Yhteenvedo*-osiossa kyseltiin saavutettuja hyötyjä ja pyrittiin nostamaan esiin hyviä käytäntöjä sekä toivottiin annettavan neuvoja ja ohjeita muilla yrityksille.

Haastattelukysymyksiä oli yhteensä 46 kappaletta, ja ne löytyvät kokonaisuudessaan liitteestä 3. Kysymykset pyrittiin rakentamaan siten, että niiden kautta voisi saada vastaukset kaikkiin tutkimuksen tavoitteisiin (kohta 4.2.).

4.6 Nokia-testi

Haastattelututkimuksen toisessa osiossa käytettiin yhdeksää Nokia-testin kysymystä. Nokia-testi on peräisin vuodelta 2005, kun Bas Vodde aloitti koulutukset ja valmennukset suomen Nokia Networksilla. Ensimmäinen Nokia-testi keskittyi ketteriin käytäntöihin. Vuonna 2007 Bas Vodde muutti kiinaan kouluttamaan Scrumia yhdistyneen Nokia Siemens Networksin henkilökunnalle kiinassa ja päivitti samalla Nokia-testin sisältämään Scrum-käytänteitä. Vuosina 2007 ja 2008 Jeff Sutherland kehitti testiä eteenpäin Scrum-sertifiointikursseille ja kehitti testiin myös pisteytyksen. [27]

Nokia-testi pitää sisällään yhdeksän kysymystä Scrumin eri osa-alueilta:

- iteraatiot,
- testaus,
- ketterä määrittely,
- tuoteomistaja,
- tuotteen työlista,
- työmääräarviot,
- Burndown-kaavio,
- tiimin häiriötekijät,
- tiimi.

Jokainen kysymys pisteytetään asteikolla 0-10, ja vastausten keskiarvona saadaan arvo kuvaamaan yrityksen ketteryyttä Scrumissa. Nokia-testi kysymyksineen, vastausvaihtoehtoineen ja pisteytyksineen löytyy kokonaisuudessaan liitteestä 4. Testiin on kohdistunut myös paljon kritiikkiä, eikä sen voi sanoa olevan universaali ketteryyden mittauskeino. Jonkinlaisen arvion se kuitenkin pystyy Scrum-prosessin ketteryydestä antamaan.

5. KYSELYTUTKIMUKSEN TULOKSET

Diplomityössä haastateltiin kahdeksatoista henkilöä yhdessätoista eri yrityksessä. Haastatteluissa oli mukana projekti-, kehitys-, ja laatuapäällikköjä sekä sovelluskehittäjiä. Välillä haastatteluissa syntyi herkullisia tilanteita, kun mukana oli sekä tuoteomistaja että ScrumMaster, joiden näkemykset käytetyistä toimintatavoista erosivat toisistaan. Scrum-toimintatavoista olisi selkeästi tarpeen keskustella monessa yrityksessä laajemminkin.

Yritykset toimivat myös hyvin erityyppisillä sovellusalueilla. Viiden yrityksen voi sanoa erikoistuneen sulautettuun ohjelmistotuotantoon, kaksi toimi mobiilisovelluksien parissa ja loput tekivät kaikkea mahdollista aina web-sovelluksista räätälöityihin työpöytäsovelluksiin ja SOA:n kautta tietovarastoihin.

Haastateltujen yritysten IT-henkilöstömäärä vaihteli välillä 15-175. Haastatteluissa kysyttiin siis ainoastaan IT-henkilöstön määrää, joten mukana on myös yrityksiä, joilla paljon muutakin henkilöstöä kuin pelkkä IT-puoli. Yritysten henkilöstön kokemus ohjelmistokehityksestä vaihteli huomattavasti. Tiimeissä oli mukana niin 40 vuoden IT-kokemuksen omaavia kuin myös suoraan koulun penkiltä palkattuja henkilöitä. Eräässä yrityksessä todettiin kuitenkin osuvasti kysyttäessä henkilöstön kokemusta alalta, että kokemusvuodet eivät välttämättä kerro mitään osaamisen tasosta.

5.1 Scrumin käyttö

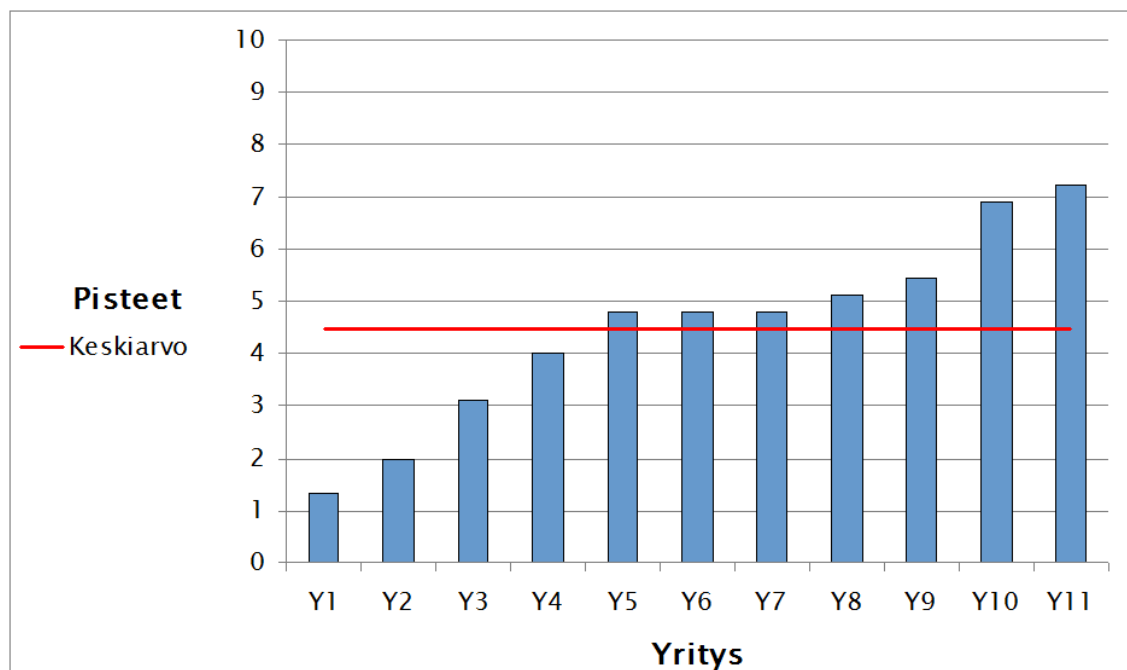
Kymmenessä yrityksessä oli omien sanojensa mukaan käytössä Scrum-menetelmä, ja yhdessä oli kehitetty RUP:n ja XP:n pohjalta oma ketterä menetelmä, jota oli käytetty vuodesta 2007 lähtien. Scrumia käyttävien yritysten toimintatavoissa oli kuitenkin huomattavia eroja siinä kuinka tarkasti ne noudattivat Scrumin periaatteita.

Scrumin käytöstä yrityksillä oli myös vaihtelevasti kokemusta. Joillain Scrumia oli käytetty vasta alle vuoden, ja joillain oli menetelmästä jo viiden vuoden kokemus. Suurimmalla osalla Scrum-kokemusta oli kuitenkin 1,5-3 vuotta. Scrum-kokemusvuosien tasaisuus luultavasti kertoo enemmän siitä, että monet Pirkanmaalaisista yrityksistä alkoivat käyttää menetelmää samoihin aikoihin. Moni yrityksistä kertoi lähtösyyksiksi Scrumin käyttöön Tampereen teknillisellä yliopistolla järjestetyt Oliopäivät vuosina 2006 - 2008, joissa ketterät menetelmät ovat olleet isossa roolissa [28, 29]. Scrumin oli monesti tuonut yritykseen yksittäinen ketterästä kehityksestä innostunut projekti- tai kehityspäällikkö, joka oli vähitellen saanut muuta henkilöstöä mukaansa.

Moni yrityksistä kertoi, että Scrum on heille ensimmäinen kunnan menetelmä ohjelmistotuotantoon. Aikaisemmin heillä ei ollut käytössä mitään erityistä menetelmää, ainoastaan löyhästi omaksuttuja oppeja vesiputousmallisesta kehityksestä. Scrumin käyttöön oli usein lähdetty projektinhallinnan näkökulma edellä, ja oppeja oli omaksuttu pikkuhiljaa.

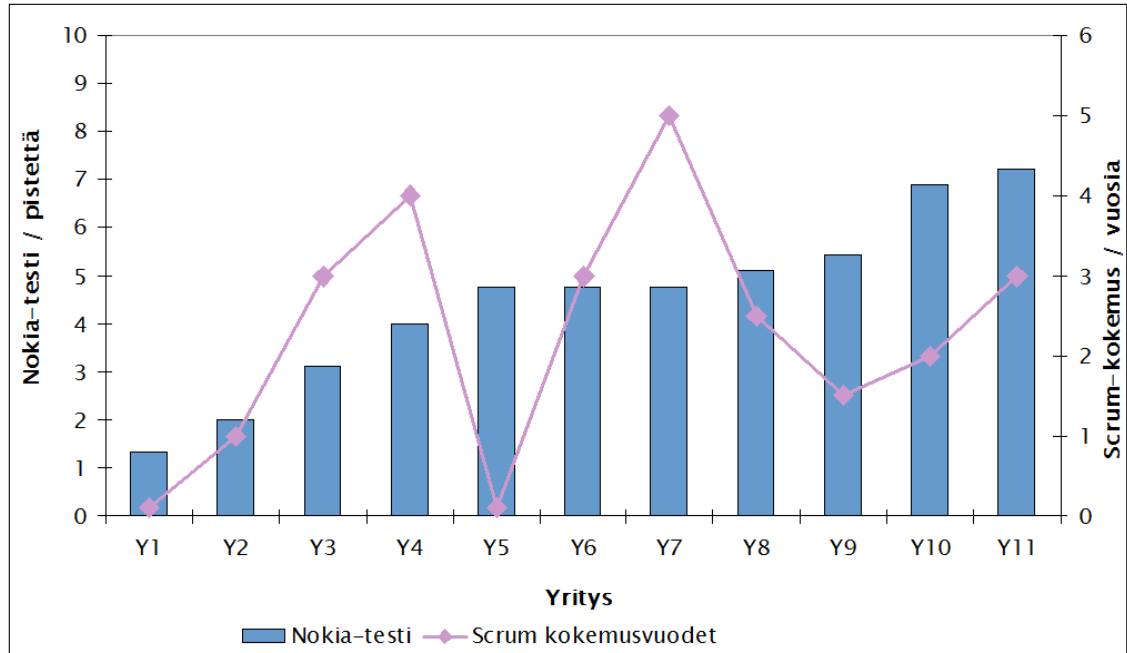
5.1.1 Nokia-testin tulokset

Nokia-testissä tyypillinen tulos Scrum-kursseilla on Jeff Sutherlandin mukaan 4 / 10 [27]. Tutkituissa yrityksissä Nokia-testin keskiarvoksi muodostui 4,5, joka on lähellä tätä Sutherlandin havaintoa. Tulosten hajonta oli 1,3 pisteestä 7,2 pisteeseen. Kuvassa 5.1. on tarkemmin esitettyä yritysten pistejakauma testissä. Joukossa oli mukana yksi yritys, joka oli kehittänyt täysin oman ketterän menetelmänsä, mutta silti sai tässä testissä keskiarvon tuloksen 4,8. Tästä voidaan päätellä, että testin tulos pystyy myös jollain tapaa kuvaamaan yleistä ketteryyttä, vaikka keskittyikin Scrumin prosessin noudattamiseen. Joukosta erottui myös selkeästi muutama yritys, joka kertoi itse noudattavansa Scrum-prosessia, mutta testin tulosten perusteella he eivät juuri käyttäneet Scrumia. Nämä yritykset olivat ehkä ottaneet ohjelmistotuotantoprosessiinsa yhden tai kaksi käytäntöä Scrumista, mutta muuten eivät noudattaneet ketterien menetelmien käytäntöjä. Nämä yritykset saivat testissä kaksi pistettä tai vähemmän. Toinen joukosta erottunut ryhmä oli kaksi keskivertoa paremmin testissä pärjännyttä, joiden pisteet olivat kuusi tai enemmän. Kaiken kaikkiaan testin perusteella vaikutti kuitenkin siltä, että näissä haastatelluissa yrityksissä Scrum-prosessia ei noudatettu kovin orjallisesti. Lisäksi tämän testin perusteella voisi todeta, että yritysten toimintatavoissa olisi paljon parantamisen varaa.

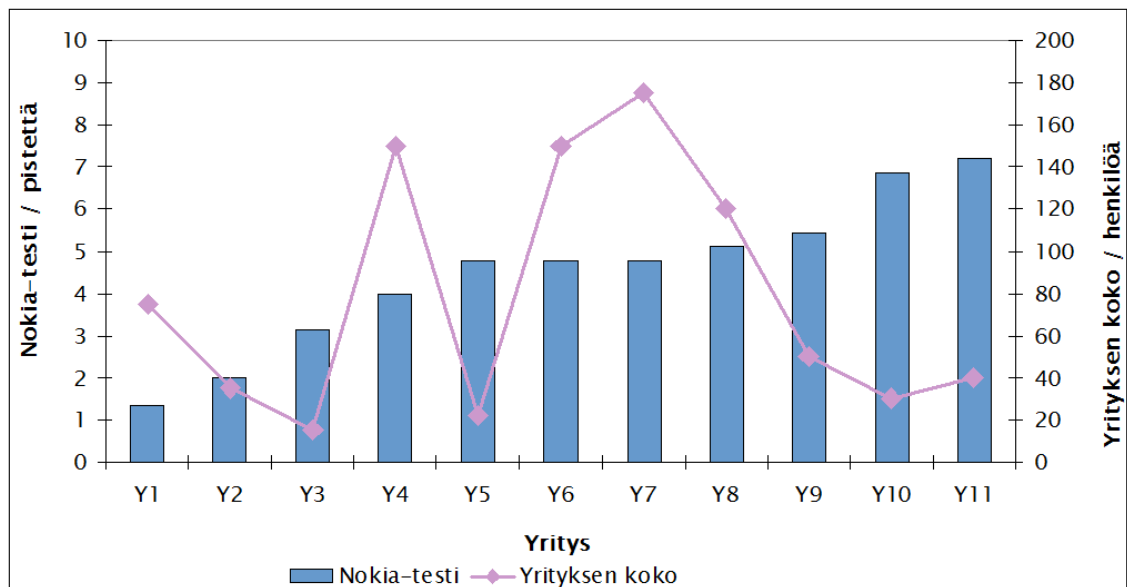


Kuva 5.1. Tutkittujen yritysten tulokset Nokia-testissä

Kun tarkastellaan erikseen yrityksen Scrum-kokemuksen ja Nokia-testin tuloksien yhteyttä (kuva 5.2.) sekä yrityksen koon ja Nokia-testin tuloksien yhteyttä (kuva 5.3.), havaitaan, että mitään suurempia johtopäätöksiä näiden välisestä yhteydestä ei voida tehdä. Testissä hyvin pärjänneillä yrityksillä oli jotakuinkin sama 1,5-3 vuoden kokemus menetelmästä kuin huonomminkin pärjänneillä. Poikkeuksena joukossa oli yksi yritys, jolla oli jo viiden vuoden kokemus Scrumin käytöstä, mutta sekin pärjäsi testissä keskiaverrosta.



Kuva 5.2. Scrum-kokemuksen ja Nokia-testin tuloksien yhteys



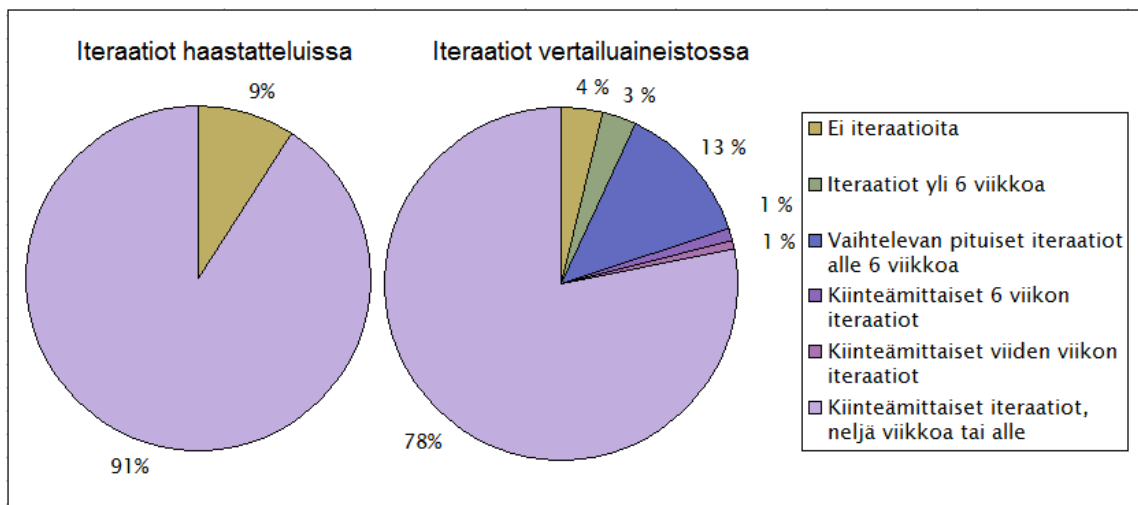
Kuva 5.3. Yrityksen koon ja Nokia-testin tuloksien yhteys

Myöskään yrityksen koolla ja Nokia-testin tuloksilla ei näytä olevan selvää yhteyttä (kuva 5.3.). Yrityksen kokoon ei tässä tapauksessa laskettu mukaan kuin IT-henkilöstön määrä, joten mukana on myös yrityksiä, joilla paljon muutakin henkilöstöä kuin pelkkä IT-puoli. Parhaiten testissä pärjänneillä kolmella yrityksellä oli alle 50 henkeä, mutta toisaalta pieniä yrityksiä löytyi myös huonosti pärjänneiden joukosta. Isommat, yli 120 hengen yritykset sijoittuivat Nokia-testissä keskivaiheille, joka kertoo osaltaan sitä että myös suurempi IT-yritys voi olla ketterä.

5.1.2 Nokia-testin tuloksien vertailu

Antoine Vernois on koonnut verkkosivustolleen tietokantaa Nokia-testin tuloksista. Sivustolle saa kuka tahansa käydä kirjaamassa omat tuloksensa testin kysymyksistä. Tavoitteena on helpottaa testin tuloksien vertailua muiden saamiin tuloksiin, sekä kerätä tietoa Scrumin tilastollista tutkimusta varten. Testin tuloksia on verkkosivulle kertynyt jo lähes 770 kappaletta. Tämä muodostaa hyvän verrokkiaineiston haastattelu-tutkimuksessa kerätylle Nokia-testin aineistolle. [30]

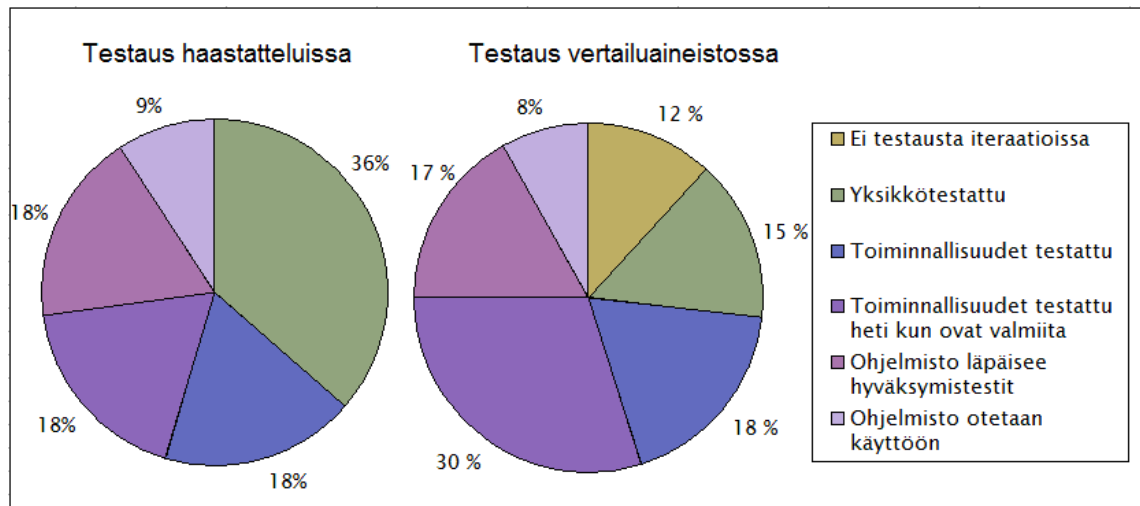
Iteraatiokysymyksen jakauma on hyvin samansuuntainen verrokkiaineiston kanssa (kuva 5.4.). Lähes kaikilla haastatelluilla iteraatioiden pituus on neljä viikkoa tai alle neljä viikkoa (91 % vastauksista) kuten myös verrokkiaineistossa (78 %).



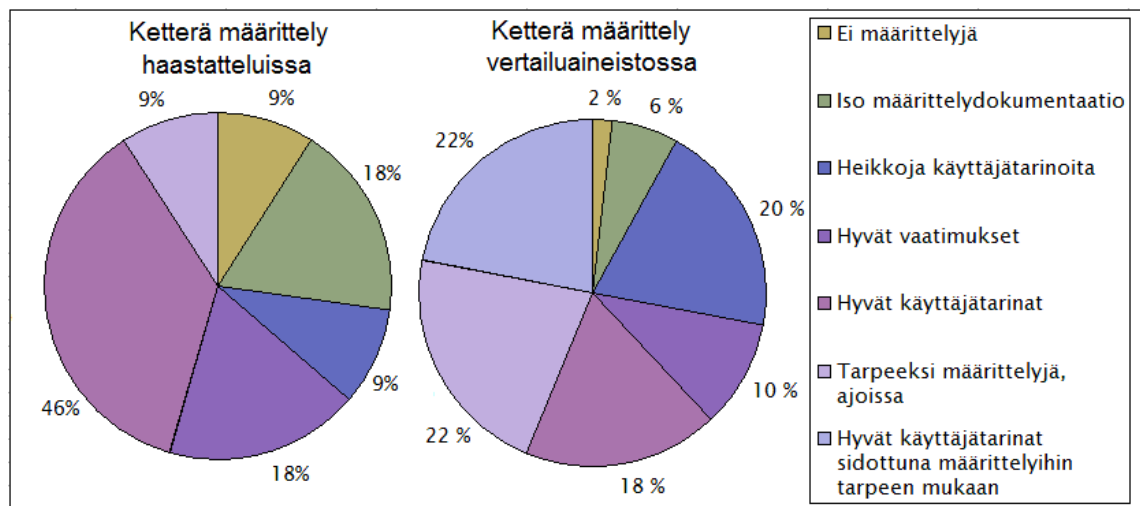
Kuva 5.4. Nokia-testin iteraatiokysymysten vastausjakauma

Testauksen osalta yleisin vastaus oli yksikkötestien tekeminen (37 %), kun taas verrokkiaineistossa näiden osuus pienempi (12 %). Verrokkiaineistossa yleisintä oli toiminnallisuuden testaaminen heti niiden valmistuttua (30 %), kun haastatteluissa tällaisia tuli vastaan vain 18 % (kuva 5.5.).

Haastateltujen yritysten ketterä määrittely tuntui keskittyvän hyviin käyttäjätarinoihin (37 %) ja vaatimuksiin (18 %). Verrokkiaineistossa vastaajat saivat taas tarpeeksi määrittelyä ajoissa (22 %), tai hyvät käyttäjätarinat olivat sidottuna määrittelyihin tarpeen mukaan (22 %) (kuva 5.6.).



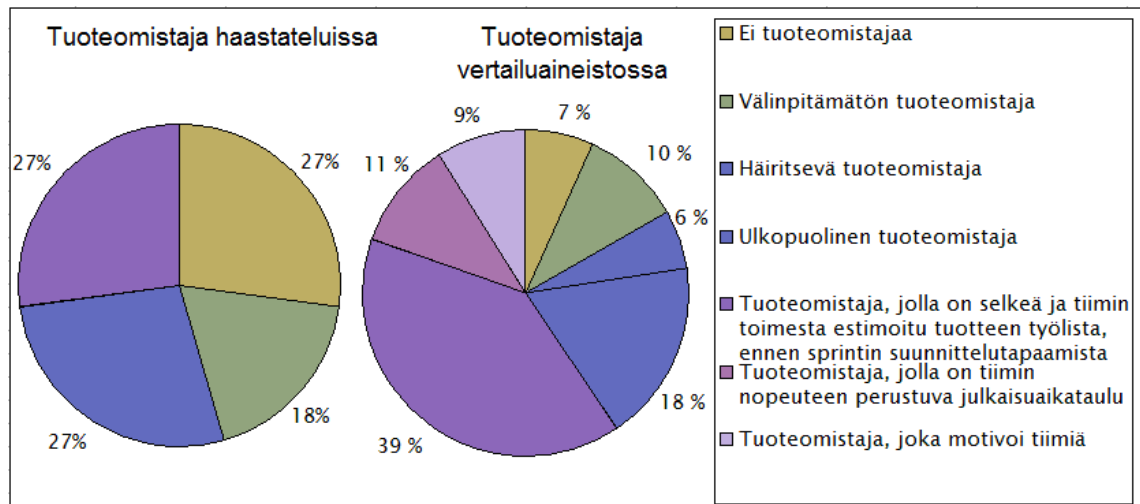
Kuva 5.5. Nokia-testin testauskysymysten vastausjakauma



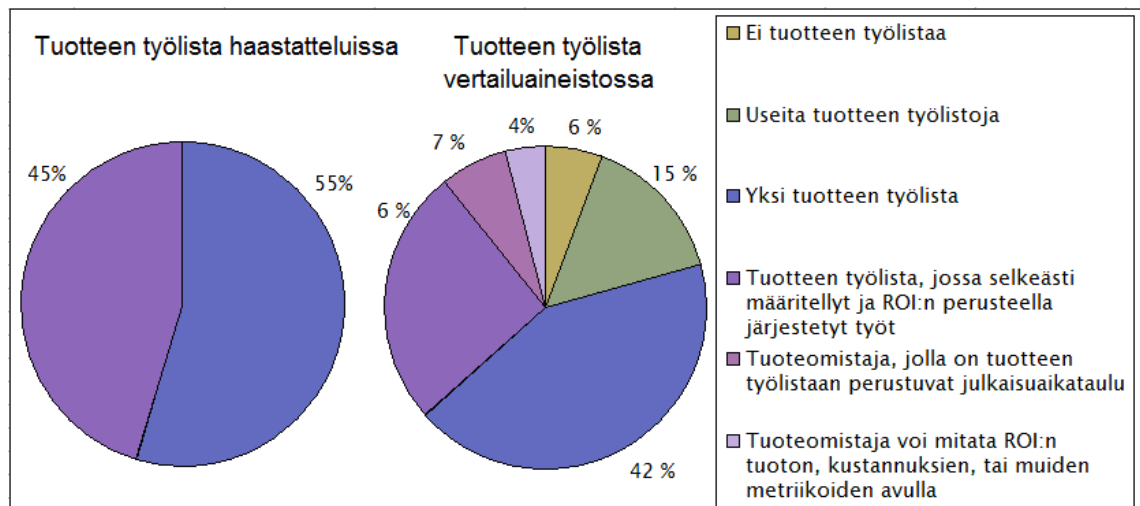
Kuva 5.6. Nokia-testin määrittelykysymysten vastausjakauma

Haastatelluissa yrityksissä yllättävän moni oli ilman selkeästi tehtävään omistautunutta ja motivoivaa tuoteomistajaa (28 %), kun verrokkiaineistossa näiden osuus oli vain 7 %. Yleistä tuntui myös olevan että tuoteomistaja oli välinpitämätön (18 %) ja häiritsevä tai ulkopuolinen (27 %). Verrokkiaineistossa suosituin vastaus oli tuoteomistaja, jolla on selkeä tuotteen työlista (40 %). Haastatteluaineistossa tällaisten vastausten osuus oli vain 27 % (kuva 5.7.).

Haastatelluissa yrityksissä tuotteen työlista oli käytössä lähes kaikilla, eikä käytössä ollut useita työlistoja. Verrokkiaineistossa oli taas 6 % yrityksiä, joissa tuotteen työlistaa ei käytetty ollenkaan ja 15 % käytti useampaa kuin yhtä työlistaa. Verrokkiaineistossa tuli kuitenkin vastaan myös yrityksiä, joiden julkaisuaikataulu perustui tuotteen työlistaan (7 %) tai joissa voitiin mitata tehdyn työn liiketoiminnallista kannattavuutta (4 %). Haastatelluissa yrityksissä näiden tilalla oli lähinnä selkeästi määritellyn ja järjestetyn tuotteen työlistan käyttö (45 %) (kuva 5.8.).

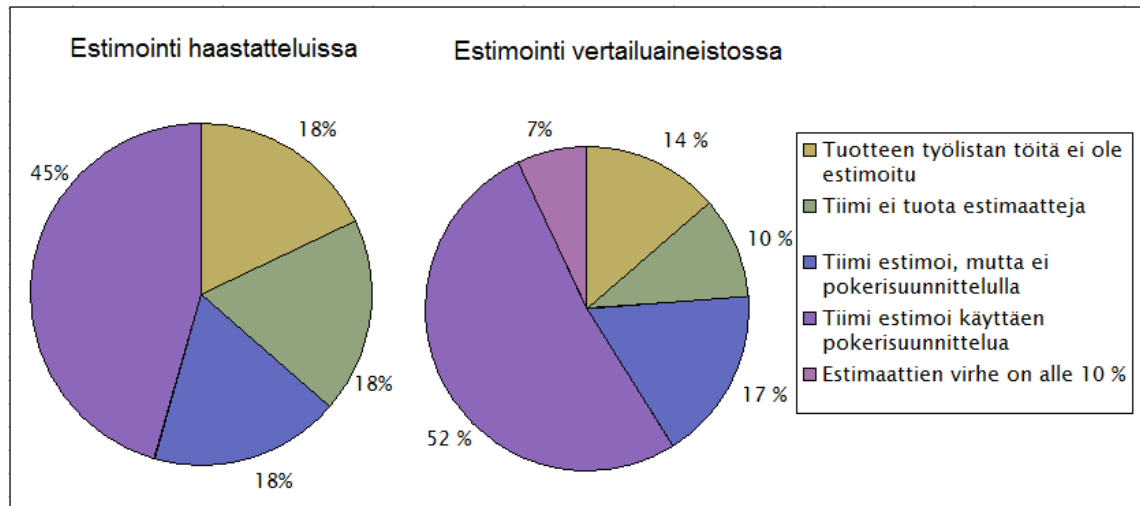


Kuva 5.7. Nokia-testin tuoteomistajakysymysten vastausjakauma

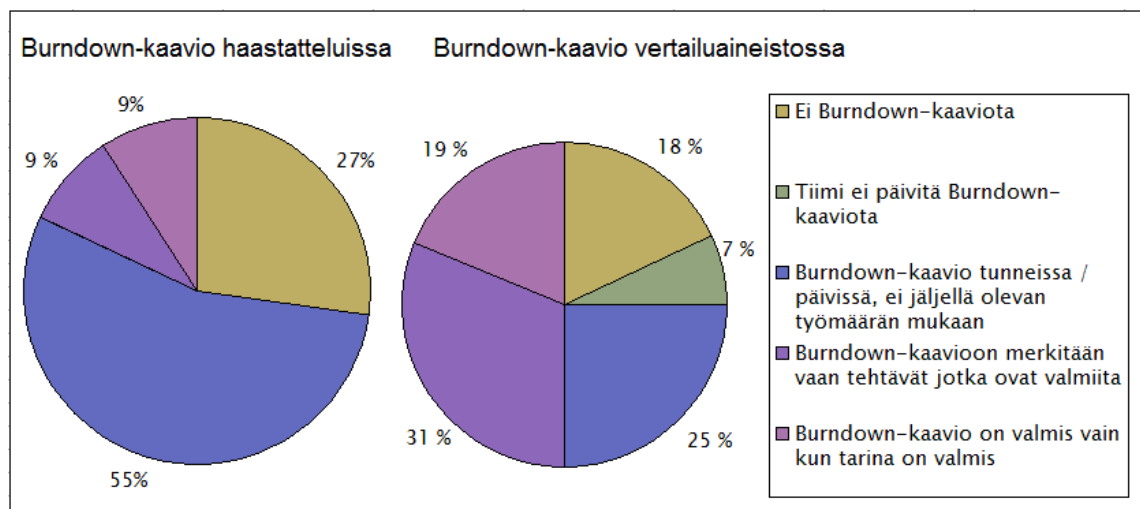


Kuva 5.8. Nokia-testin tuotteen työlista -kysymyksen vastausjakauma

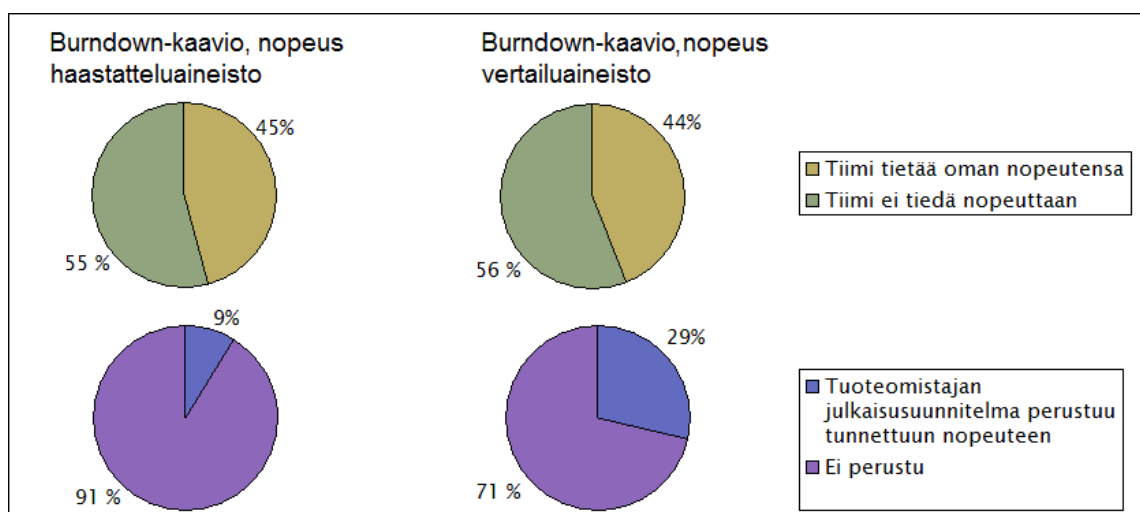
Työmäärän arviointi -kysymyksessä haastatteluvastaukset olivat hyvin lähellä verrokkiaineiston vastauksia (kuva 5.9.). Burndown-kaavion osalta verrokkiaineistossa oli enemmän vastauksia, joissa tehty työ merkattiin vasta täysin valmiit tarinat (19 %) tai tehtävät (31 %). Haastatteluissa yrityksissä Burndown-kaavioon merkattiin taas heti tehdyt tunnit (55 %), eikä tarinoita tai tehtäviä (kuva 5.10.). Vain hieman alle puolet tiimeistä tiesi oman nopeutensa (45 %), joka vastaa täysin verrokkiaineistoa (44 %). Haastatteluissa yrityksissä huomattavasti pienempi osuus (9 %) oli niitä, joissa tuoteomistajan julkaisusuunnitelma perustui tiimin tunnettuun nopeuteen. Verrokkiaineistossa tällaisten vastausten osuus oli 29 % (kuva 5.11.).



Kuva 5.9. Nokia-testin tuotteen työmäärän arviointi -kysymyksen vastausjakauma

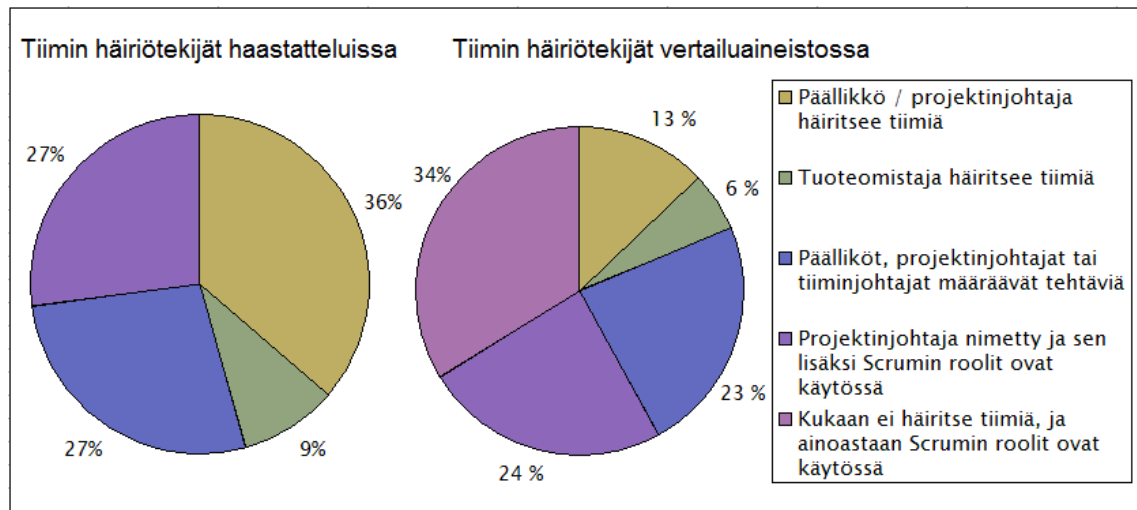


Kuva 5.10. Nokia-testin Burndown-kaaviokysymyksen vastausjakauma

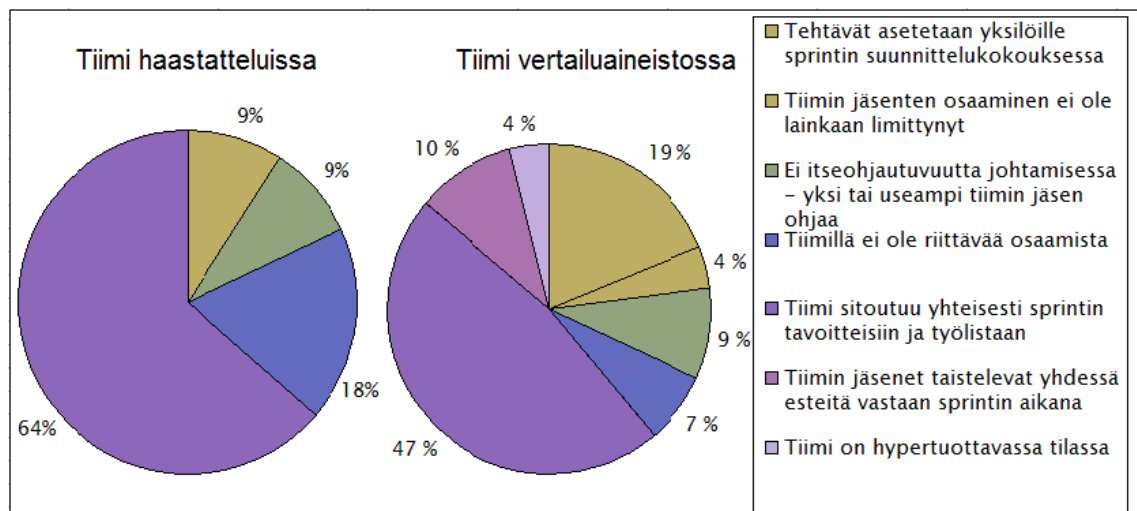


Kuva 5.11. Nokia-testin nopeuskysymyksen vastausjakauma

Tiimin häiriötekijät keskittyivät haastatteluaineistossa siihen että päällikkö tai projektinjohtaja häiritsee tiimiä (37 %). Verrokkiaineistossa selkeästi isompi osuus niitä tapauksia, joissa kukaan ei häiritse tiimiä ja käytössä on ainoastaan Scrumin roolit (34 %) (kuva 5.12.). Häiriötekijäkysymys oli kuitenkin hyvin hankala pisteyttää ja arvioida, koska siinä yritettiin yhdistää kahta täysin eri kysymystä: millaisia häiriöitä tiimille tulee, ja onko tiimillä käytössä Scrumin mukaiset roolit. Tiimien toiminnassa sekä haastatteluissa (64 %) että verrokkiaineistossa (47 %) selvästi yleisintä oli, että tiimi sitoutuu yhteisesti sprintin tavoitteisiin ja työlistaan (kuva 5.13.).



Kuva 5.12. Nokia-testin tiimin häiriötekijäkysymyksen vastausjakauma



Kuva 5.13. Nokia-testin tiimikysymyksen vastausjakauma

Yhteenvetona voisi todeta, että ketterässä määrittelyssä olisi haastatteluissa yrityksissä vielä paljon parannettavaa. Myös tuoteomistajan käyttäytymisessä ja hyödyntämisessä jäätin selvästi jälkeen kansainvälistä verrokkiaineistoa. Tuotteen työlistan hyödyntämisessä oltiin kuitenkin paljon pidemmällä kuin verrokkiaineiston vastaajat. Häiriöitä tuntuu kuitenkin tulevan varsinkin päälliköiden tai projektinjohtajien tahoilta enemmän kuin muualla, joka voi hyvin olla seurausta suomalaisesta autoritäärisestä johtamiskulttuurista.

5.2 Scrumin soveltaminen

Haastattelukysymyksien kautta yrityksistä saatiin monenlaista tietoa Scrumin soveltamisesta käytännössä. Käytäntö erosi usein melko paljonkin siitä, mitä Scrum-kirjallisuudessa on kuvattu esimerkkiprosessiksi tai -tapauksiksi. Seuraavissa alakohdissa on kattavasti kirjattu yhdentoista yrityksen edustajien haastattelujen kautta saatuja havaintoja Scrumin soveltamisesta kyseisissä yrityksissä.

5.2.1 Scrum-koulutus

Scrum-koulutuksista haastateltavat antoivat hyvää palautetta. Koulutuksien myötä tietoisuus, innostuneisuus ja hyvät käytännöt lisääntyivät. Todettiin myös, että Scrumin käytäntöjen hyödyntäminen tehostui koulutuksien jälkeen. Monet kehittivät aluksi kokeilemaan menetelmää ensin käytännön kautta ja sitten hakemaan koulutuksilla sertifikaatit. Koulutuksia käytettiin myös herättämään ajatuksia ja sitouttamaan ihmisiä Scrumiin. Kohdassa 4.1. tarkasteltiin jo TTY:n Scrum-koulutuksia ja heti koulutuksien jälkeen saatua positiivista palautetta. Nämä positiiviset mielikuvat olivat edelleen havaittavissa haastatteluissa. Hyvän kouluttajan kurssi koettiin suositeltavana, ja kouluttajalta pitää myös löytyä näkemystä ja kokemusta. Muutamalle yritykselle asiakas oli jopa tarjonnut ensimmäiset Scrum-koulutukset.

Scrum-koulutuksiin lähetettyjen ihmisten asema ja lukumäärä vaihteli paljon, riippuen yrityksen rakenteesta ja hierarkiasta. Joissain yrityksissä kaikki työntekijät oli lähetetty ScrumMaster-koulutuksiin, toisissa ainoastaan tiiminvetäjät. Projektipäälliköt tai osastonjohtajat oli useimmiten lähetetty tuoteomistaja-koulutuksiin. Kahdessa haastatellussa yrityksessä myös ylempi johto tai maajohtajat olivat käyneet tuoteomistaja-kurssilla.

Muuta koulutusmateriaalia oli myös käytetty vaihtelevasti. Suurimmalla osalla yrityksistä oli oma käsikirjasto, josta löytyi ketteristä menetelmistä ja Scrumista kertovaa kirjallisuutta. Yhdessä yrityksessä oli jopa teetetty diplomityö yrityksen käyttämästä ketterästä menetelmästä. Monet olivat myös käyttäneet muita koulutuksia kuin TTY:llä järjestetyn koulutuksen. Jotkut arvelivat myös, että voi olla hyvä idea ottaa ulkopuolinen konsultti Scrumin käynnistämiseen. Kahdessa yrityksessä oli myös panostettu omaan koulutusmateriaaliin. Heillä oli Scrumiin oma sisäinen koulutuspakettinsa, joka sisälsi menetelmän perusteet, työtapaohjeet, hyviä toimintatapoja ja joka toimi sisäisenä ohjenuorana kehitykseen.

5.2.2 Asiakkaiden suhtautuminen

Asiakkaiden suhtautuminen Scrum-menetelmään oli hyvin vaihtelevaa. Monet yritykset totesivat, etteivät asiakkaat ymmärrä menetelmää. Jotkut asiakkaat olivat jopa todenneet, etteivät haluakaan ymmärtää. Heitä kiinnosti lähinnä lopputulos, eivätkä halunneet kuulla mitään käytetyistä menetelmistä tai keskeneräisistä tuotoksista. Joissain yrityksissä ei ollut edes uskallettu kertoa asiakkaille, että käytetään Scrumia.

Toisaalta monet asiakkaat kiittelivät avoimempaa ja säännöllistä kommunikaatiota sekä joustavuutta. Asiakkaat osallistuivat usein sprintin suunnittelu- ja katselmointipalavereihin. Välillä ongelmaksi muodostui tosin se, että asiakkaan kontaktihenkilöille ei ollut varattu työaikaa yhteisiin projekteihin. Muutamat asiakkaat olivat kuitenkin osanneet jopa vaatia Scrumin käyttöä sekä tarjonneet siihen koulutusta.

Asiakkaat vaativat usein kiinteähintaisia projekteja, jotka nähtiin taas toimittajien puolelta ongelmallisina ketterien menetelmien kanssa. Myös muutostenhallintaprosessit voivat olla hyvinkin raskaita. Monet yritykset toivoivatkin asiakkaiden maksavan ennemmin työstä kuin kiinteän työmäärän ja hinnan projekteista. Ketterät sopimukset vaativat kuitenkin pitkää luottamuksen rakentamista, koska monia asiakkaita pelotti vaihtuva sisältö projektissa. Monessa yrityksessä olikin projektit myyty kiinteällä työmäärällä ja hinnalla, mutta sisäinen toimintapa noudatteli Scrumin mallia.

5.2.3 Organisaatiokulttuuri

Noin puolet haastatelluista henkilöistä kertoi, että Scrum sopii yhteen heidän organisaatiokulttuurinsa kanssa. Joissain yrityksissä kaikki muut toiminnot etenivät tiukan ylhäältä johdetun prosessin mukaan, mutta IT-puolella Scrumia käytettiin johdolta salassa. Joissain yrityksissä oli käytössä vanhaa prosessimallia ja Scrumia sekaisin. Tällöin esimerkiksi päätöksenteko hajautui tuoteomistajille ja tuotepäälliköille, ja monet muutkin roolit sekoittuivat vanhojen roolien kanssa. Myös työtuntien tarkka seuranta koettiin jäykän organisaatiomallin taakkana. Muutosvastarintaa Scrumiin siirtyessä esiintyi myös. Kauemmin yrityksessä olleilla oli välillä sopeutumisvaikeuksia uuteen toimintatapaan, ja sen takia oli jopa lähdetty yrityksestä pois. Muutosvastarinta näkyi myös esimerkiksi niin, ettei kaikkiin yhteisiin palavereihin saavuttu paikalle. Haastateltujen yrityksiä joukossa oli toisaalta kaksi yritystä, joiden koko toiminta perustui Scrumin periaatteisiin. Näillä organisaatio rakentui Scrum-of-Scrums - tyyppiselle toiminnalle.

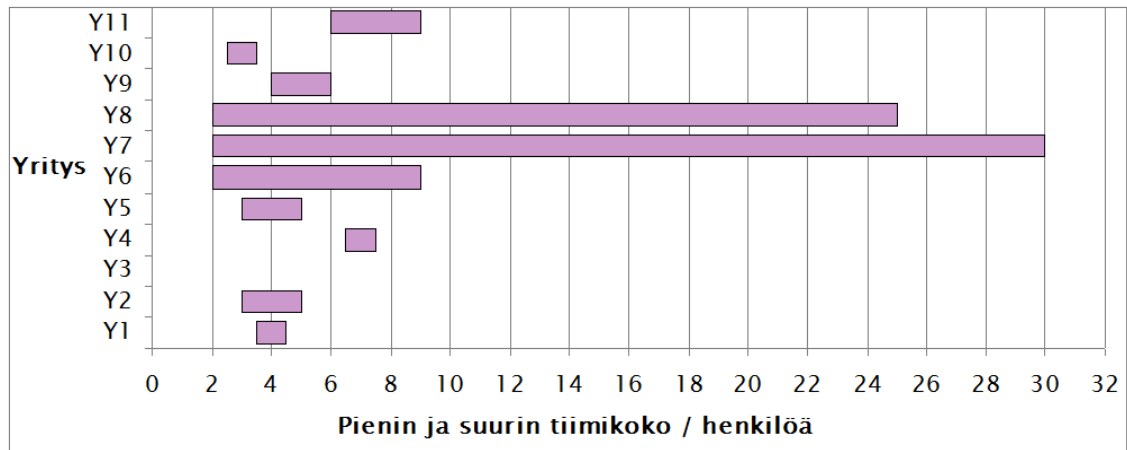
Johdon tuki Scrumille vaihteli haastatelluissa yrityksissä. Muutamassa oli johdon täysi tuki takana, ja suurin osa ylemmästä johdostakin oli käynyt tuoteomistaja-koulutuksissa. Koulutuksia oli myös ajettu yritykseen johdon toimesta, ja asetettu tavoitteeksi tietty osuus sertifioituja Scrum-osaajia. Monesti johto oli innoissaan tuloksista, vaikkeivät ehkä täysin ymmärtäneet menetelmää. Mukana oli myös yrityksiä, joissa johto ei ollut aktiivisesti edistänyt menetelmän käyttöä, vaan se oli lähinnä muutaman tiimin omaa puuhastelua. Haastatelluissa myös todettiin pariin otteeseen, että sitoutuminen Scrumiin pitää lähteä sekä tiimistä että johdosta käsin.

5.2.4 Tiimit

Scrum-tiimien koot erosivat paljon toisistaan eri yrityksissä ja myös yksittäisen yrityksen sisällä. Tiimikoko vaihteli kahdesta henkilöstä kolmeenkymmeneen, riippuen projektista (kuva 5.14.). Suurimmalla osalla yrityksistä oli kuitenkin tavoitteena 4-6

hengen tiimit. Esimerkiksi kymmenen hengen tiimi oli erässä yrityksessä koettu liian isoksi ja jaettu kahtia.

Tiimit olivat pääsääntöisesti itseohjautuvia, ainoastaan muutamassa yrityksessä oli määrätty rooleja, kuten testaaja, käyttöliittymäsuunnittelija, arkkitehti. Itseohjautuvuus koettiin toimivana, päällekkäinen osaaminen oli kasvanut ja tuoteomistaja on voinut jättää monia asioita tiimin itsensä ratkaistavaksi. Jopa yhdessä yrityksessä, jossa projektipäällikkö erikseen jakoi tehtävät, todettiin, että motivoitumisen kannalta olisi kuitenkin tärkeää antaa tiimille vapautta päättää omat tehtävänsä.



Kuva 5.14. Tutkittujen yritysten pienin ja suurin tiimikoko henkilöissä

5.2.5 Iteraatiot

Vaikka iteratiivisuus kuuluu hyvin keskeisesti kaikkiin ketteriin menetelmiin, niin silti haastattelussa paljastui yrityksiä, joissa Scrumin yhteydessä ei ollut käytössä sprinttejä. Näissä yrityksissä toiminta perustui enemmän tai vähemmän johonkin ylhäältä määrättyyn isompaan suunnitelmaan, jota noudatettiin sovelluskehityksessä.

Suurimmalla osalla sprintit olivat kuitenkin käytössä, ja niiden kesto vaihteli kahdesta viikosta neljään. Kahden tai kolmen viikon mittainen sprintti todettiin useimmiten tehokkaimmaksi, koska silloin vauhti oli tasainen eikä kulu liikaa aikaa suunnitteluun. Myös lyhyt palautteenantosykli koettiin eduksi kahden viikon sprinteissä. Kehittäjien ei myöskään tarvitse miettiä tulevaisuutta kuin muutama viikko eteenpäin, jos eivät pidemmälle halua.

Alle kahden viikon sprinteissä ongelmaksi muodostui liiallinen palaverien määrä suhteessa varsinaiseen työaikaan. Eräs yritys oli myös havainnut että yli kolmen viikon sprintit eivät toimi, koska niin kauas tulevaisuuteen ei pystytä ennustamaan. Työmääräarviot heittävät pitkällä sprinteillä, ja häiriöitä ehti tulla enemmän. Yrityksessä arvioitiin että kuukauden mittaisista sprinteistä 40–60 % onnistuu, ja kahden viikon sprinteissä 80 % onnistuu odotusten mukaisesti. Joskus kuitenkin yritykset oli pakotettu toimimaan neljän viikon sprinteissä, koska silloin toiminta synkronissa yrityksen muiden toimintatapojen, kuten työaikakirjanpidon, kanssa.

5.2.6 Testaus

Yrityksien testausautomaatiossa oli selkeästi paljon parannettavaa. Jatkuva automaattinen yksikkötestaus oli käytössä vain kolmessa yrityksessä, ja kahdessa siitä haaveiltiin. Testausautomaation kehittäminen nähtiin jatkuvana parantamisena. Perinteistä yksikkötestausta kuitenkin käytettiin kaikkialla. Erillisiä hyväksymiskriteerejä tai hyväksymistestausta käytettiin neljässä yrityksessä. Testaus koettiin usein hankalana, haastavana tai ongelmallisena. Jotkut totesivat että testaus ei nivoudu Scrumiin, kukaan ei nauti testauksesta, tai että kehittäjät eivät testaa tarpeeksi. Toisaalta jotkut olivat todenneet testivetoisen kehityksen (*en. Test-Driven Development, TDD*) hyväksi, ja monesti virheet korjattiin saman tien havaitsemisen jälkeen [31].

5.2.7 Määrittely

Määrittelyjä tehtiin hyvin eri lailla eri yrityksissä. Jotkut tekivät kaikki määrittelyt ennen ensimmäistäkään iteraatiota, joka ei ole kovin ketterää tai Scrumin mukaista. Määrittelyjen muoto ja formaatti riippui paljon myös projektista ja asiakkaasta. Monellakaan yrityksellä ei ollut käytössä formaaleja määrittelyjä. Niiden sijaan käytössä oli käyttötapauksia tai käyttäjätarinoita. Moni myös korosti, että ketteryys ei tarkoita kokonaan määrittelyistä luopumista.

5.2.8 Tuoteomistaja

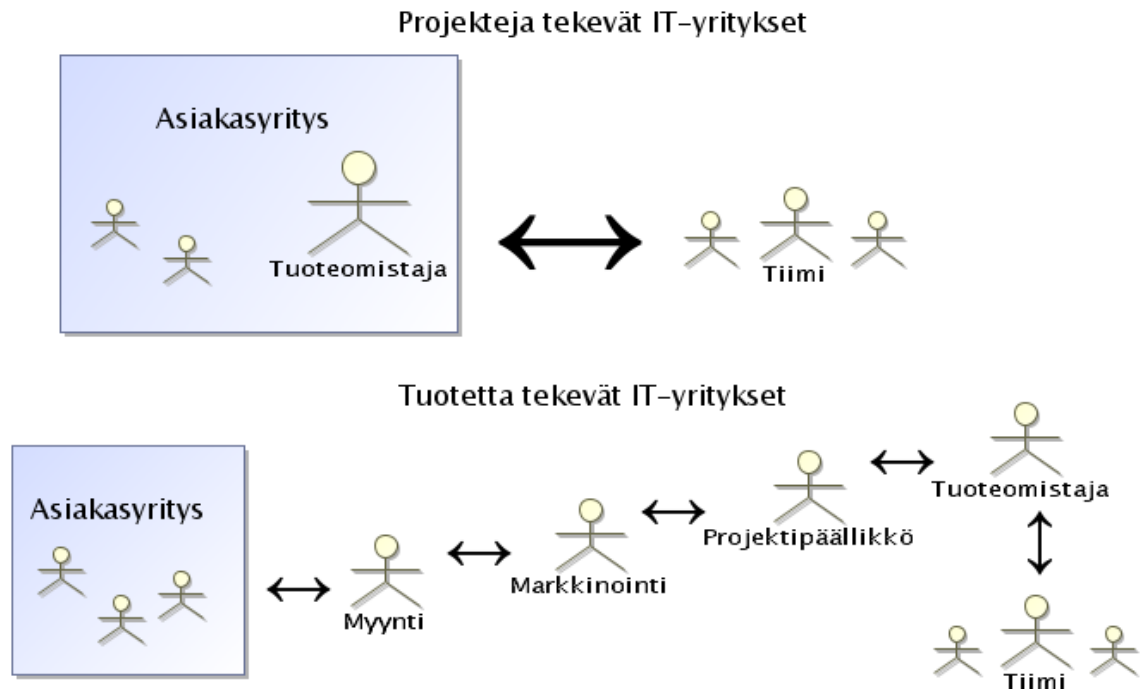
Tuoteomistajan rooli koettiin usein haastavana. Haastateltujen yritysten joukossa oli kahta eri koulukuntaa, niitä joissa tuoteomistaja tuli yrityksen sisältä, ja niitä joissa tuoteomistaja oli asiakkaan puolelta. Tuoteomistajana toimiminen koettiin haastavaksi, koska täytyy hallita niin paljon eri asioita; määrittely, asiakkaan ongelmien analysointi, asiakasrajapintayhteys, hyväksymistestaus, ongelmien selvittely, eri palavereissa toimiminen. Kahdessa haastatellusta yrityksistä ei projekteissa käytetty tuoteomistajia lainkaan.

Seitsemässä yrityksessä tuoteomistaja tuli yrityksen sisältä. Tuoteomistajat saattoivat usein kävellä tiimin yli ja sekaantua liikaa käytännön työhön. Tuoteomistaja saattoi olla myös virtuaalihahmo, niin että joku tiimistä ottaa aina kyseisen roolin.

Kahdessa yrityksessä tuoteomistaja oli asiakkaan puolella. Silloin ongelmaksi muodostui se, että aika tai ymmärrys ei riittänyt tuoteomistajana toimimiseen ja tiimi jätettiin usein yksin priorisointien, ongelmien ja palaverien kanssa. Yleinen ongelma oli, että tuoteomistajia oli yrityksessä liian vähän, tai että he eivät ehtineet panostamaan kaikkiin projekteihin.

Isona ongelmaa aiheutti myös Scrumin tuoteomistaja-käsitteen sovittaminen muuhun yrityskulttuuriin. Monissa tuotetta tekevissä IT-yrityksissä tuoteomistajan rooli oli liimattu muun organisaation päälle. Tällöin ketju tiimiltä tai tuoteomistajalta asiakkaalle muodostui kohtuuttoman pitkäksi. Tällöin käytännössä tuoteomistaja ei hallinnoinut tuotteen kehitystä, vaan se oli hajautettu monille eri tahoille. Toinen ääripää oli taas

projekteja tekevät IT-yritykset, joissa kommunikaatioketju oli välillä liiankin lyhyt. Tällöin esimerkiksi asiakkaan puolella oleva tuoteomistaja saattoi häiritä sprinttien suorittamista. Kuvassa 5.15. on kuvattu näitä tuoteomistajan roolin haasteellisia tilanteita.



Kuva 5.15. Tuoteomistajan haasteellinen rooli

5.2.9 Tuotteen työlista

Tuotteen työlista oli käytössä kaikissa haastatelluissa yrityksissä. Osalla työlista tuli isommasta julkaisusuunnitelmasta, mutta useimmilla yrityksistä oli tuoteomistaja tai joku muu yksittäinen henkilö joka listaa hallinnoi. Työlistan priorisointia tehtiin hyvin erilaisten periaatteiden mukaan, esimerkiksi liiketoiminnallisen kannattavuuden (ROI), määräaikaisten, riskipitoisuuden tai johdon asettamien korkeampien tavoitteiden mukaisesti. Neljässä yrityksessä nähtiin erityisen tärkeänä se, että priorisointia tehdään aina kiinteässä yhteistyössä asiakkaan kanssa. Työlistan käyttäminen ja hallinnointi vaati tuoteomistajalta paneutumista sekä ammattitaitoa. On esimerkiksi otettava huomioon monimutkaiset riippuvuudet työlistalla olevien asioiden välillä.

5.2.10 Työmääräarviointi

Työmäärien arviointi oli pääsääntöisesti annettu tiimin tehtäväksi. Ainoastaan kahdessa yrityksessä jokin muu taho kuin tiimi teki tehtävien työmääräarvioinnin. Tiimin itsensä tekemät työmääräarviot nähtiin suurena parannuksena. Todettiin, että aiemmin kun projektipäällikkö oli tehnyt työmääräarviot, ne olivat paljon enemmän pielessä. Aiemmin myös kehittäjät saattoivat ajatella pelkästään ohjelmointityön työmäärää, mutta Scrumin myötä tiimi huomioi paremmin myös kaiken muun tarvittavan työn.

Arvioiden tekeminen itsessään opettaa kaikille, mitä työtä valmiiseen ominaisuuteen oikeasti sisältyy.

Pokerisuunnittelua käytettiin viidessä yrityksessä. Pokerisuunnittelun pisteiden käyttö oli usein jouduttu selittämään tiimeille juurta jaksuen, kunnes se oli lopulta ymmärretty kompleksisuuden mittariksi. Pokerisuunnittelua kuvattiin usein hauskaksi käytännöksi.

5.2.11 Burndown-kaavio

Burndown-kaavio oli käytössä suurimmassa osassa yrityksiä, ainoastaan kolmessa yrityksessä sitä ei nähty hyödylliseksi. Burndown-kaavion avulla tiimin nopeutta seurattiin, ja nopeus oli tiedossa neljässä yrityksessä.

Nopeuden käsite todettiin melko ongelmalliseksi. Jotkut totesivat, ettei nopeuden mittaaminen toimi kuin todella pitkäkestoisissa projekteissa, koska projektit päättyvät usein ennen nopeuden mahdollista tasaantumista. Tiimit eivät myöskään pitäneet nopeuden vaihtelusta, mutta toisaalta usein tuoteomistaja oli ollut kuitenkin tyytyväinen nopeuteen. Edistyneimmissä yrityksissä Burndown-kaavio generoitiin automaattisesti sprintin tyolistasta, tiimi tiesi oman nopeutensa ja historiadataa käytettiin suunnittelussa apuna.

5.2.12 Häiriöt

Häiriöitä ja keskeytyksiä tiimin työhön tuntui tulevan useista eri lähteistä. Ylläpitotehtävät muista projekteista, tukipyynnöt, asiakkaan pyynnöt sekä vaatimukset ja omat projektipäälliköt häiritsivät, ja hyvin harvoin nämä ylimääräiset tehtävät näkyivät kuitenkaan tuotteen tyolistassa. Asiakkaat eivät usein tuntuneet osaavat rauhoittaa sprinttiä, vaan yrittivät saada siihen jatkuvasti muutoksia. Ylipäätään todettiin, että asennemuutos ja tietoisuuden lisääminen häiriöiden käsittelyn osalta tuntui vaativan paljon aikaa ja panostusta.

Haastatteluissa todettiin myös, että ScrumMaster on epäonnistunut, jos häiriöitä tulee. Asiakkaan ja tiimin väliin tarvitaan tarpeeksi vahva ScrumMaster kilveksi, joka osaa ohjata muutospyynnöt seuraavaan sprinttiin. Scrumin käytön myötä olikin alettu enemmän määrätä työtehtäviä tiimeille tai tuoteomistajille, kun aiemmin määräyksiä tuli yksittäisille kehittäjille.

5.2.13 Retrospektiivit

Retrospektiiveistä tuli positiivista palautetta siksi, että ne lisäsivät avoimuutta ja keskustelua projekteissa. Niistä saadaan paljon kehitysideoita, jotka sitten toteutuvat vaihtelevalla menestyksellä. Palautteenantocyklin lyhentäminen nähtiin tärkeänä tavoitteena. Joillain yrityksillä parannukset jalkautuivat nopeasti, mutta niiden levittäminen muihin tiimeihin nähtiin haastavana tai jopa surkeana. Hyviä käytänteitä pyrittiin levittämään esimerkiksi Wikin avulla. Eräässä yrityksessä retrospektiivit kuuluivat ehdottoman pakollisena sprinttiin, ja retrospektiivien pitäjät olivat tehtävään

koulutettuja projektin ulkopuolisia henkilöitä. Yrityksessä nähtiin, että ulkopuoliset ammattilaiset osaavat parhaiten kysyä tiimiltä oikeat kysymykset ja saada näin tilaisuudesta eniten irti.

Toisaalta jotkut firmat olivat luopuneet retrospektiivi-käytännöstä kokonaan, koska ihmisiä oli vaikea saada avoimesti keskustelemaan keskenään.

5.2.14 Tiimin työympäristö

Scrum-kirjallisuudessa usein mainitaan ideaaliksi kommunikoinnin kannalta se, että tiimi istuu samassa työhuoneessa lähekkäin toisiaan. Haastatteluissa selvisi, että lähes kaikissa yrityksissä oli pyrkimys järjestää työympäristöä tähän suuntaan. Eräässä yrityksessä oli jopa kaadettu seiniä, jotta saadaan tiimi istumaan lähemmäs toisiaan. Avokonttoriympäristö oli myös suosittu, siinä kuitenkin tiimin jäsenet joutuivat usein huutelemaan sermien yli toisilleen. Monissa yrityksissä oli myös erillisiä Scrum-huoneita palavereita varten.

Muutamassa yrityksessä, joissa tiimi oli hajautettuna eri työpisteisiin tai eri kerroksiin, todettiin suoraan asian olevan suuri epäkohta, ja että kaikki pitäisi saada samaan tilaan työskentelemään.

5.2.15 Hajautus

Hajautettu ohjelmistotuotanto on tämän päivän todellisuutta ohjelmistoyrityksissä. Ainoastaan yhdellä yrityksellä yhdestätoista ei ollut kokemuksia hajautetusta ohjelmistotuotannosta. Hajautusta oli harrastettu monella eri tasolla, yksittäisiä tiimin jäseniä saattoi olla eri paikkakunnilla tai eri paikkakunnilla sijaitsevat tiimit saattoivat tehdä yhteistyötä.

Hajautuksesta tuntui olevan monenlaista päänvaivaa. Siitä seurasi lähes aina kommunikaatio-ongelmia, ja myös erilainen toimintakulttuuri oli ongelmana ulkomaille toimintaa hajauttaessa. Ongelmia tuli myös isoista aika-eroista ja kunnollisten etäyhteyksien puutteesta, jotka hankaloittivat päivittäistä yhteydenpitoa. Työkalut eivät tuntuneet korvaavan kasvotusten kommunikointia, mutta silti tilannetta yritettiin monesti helpottaa videoneuvotteluilla tai pitämällä Scrum-palavereita pikaviestimillä tai sähköpostilistoilla. Ainoastaan yksi yritys haastatelluista tuntui olevan tyytyväinen hajautettuun toimintaansa muiden valitellessa järjestelyjen hankaluutta. Ongelmat saattoivat tulla usein esiin sprintin katselmointitilaisuudessa, josta syystä myös nähtiin että sprinttien olisi hyvä olla suhteellisen lyhyitä.

Hyväksi käytännöksi oli monesti todettu ihmisten tapaaminen kasvotusten tarpeeksi usein, sillä näin päivittäinen etätoiminta helpottuu huomattavasti. Eräässä yrityksessä järjestettiin myös kaksi kertaa vuodessa samanmuotoinen kysely hajautetusta kehityksestä, josta havainnointiin trendiä henkilöstön tyytyväisyydestä.

5.2.16 Alihankinta

Alihankinta oli myös mukana monessa projektissa tavalla tai toisella. Osa haastatelluista yrityksistä oli itse alihankittuina mukana projekteissa, ja toiset ostivat itse töitä alihankintana muilta yrityksiltä. Monesti alihankitut työntekijät ovat paikanpäällä samoissa tiloissa yrityksen muiden työntekijöiden kanssa, jolla estetään hajautuksen tuomia kommunikaatio-ongelmia. Monesti kuitenkin tuntui olevan vaikeuksia ottaa mukaan ja sopeuttaa alihankittuja henkilöitä muuhun toimintaan, esimerkiksi heitä ei monesti ollut koulutettu Scrumiin ja oli ainoastaan osoitettu tapa millä toimitaan.

Toisaalta muutamassa yrityksessä oli päästy siihen tilanteeseen, että alihankitut työntekijät eivät juuri eronneet muista työntekijöistä ja yhteistyö toimi. Työntekijöiden vaihtuvuus, tiukka työtuntiseuranta, kommunikaatio ja sopimustekniset asiat vaikuttivat kuitenkin monesti haastavilta. Alihankitut työntekijät saattoivat esimerkiksi olla liian varovaisia tai nöyriä ilmoittamaan projektin ongelmakohdista eteenpäin. Toisaalta itse alihankkijoina toimineet valittelivat, että asiakkaat eivät ymmärrä ketteriä toimintatapoja.

5.2.17 Dokumentointi

Projekteissa tehtävään dokumentaatioon suhtauduttiin yrityksissä hyvin pragmaattisesti. Monesti mietittiin ensin asiakkaan kanssa yhdessä tarkasti mitä dokumentaatiota tarvitaan, ja ylimääräisiä ja turhia dokumentteja ei prosessissa tarvinnut tuottaa. Dokumentaation eläessä jatkuvasti monissa yrityksissä oli siirrytty käyttämään wikipohjaista dokumentointia. Wiki sai kuitenkin myös moitteita sekavuudesta.

Yleisimmin oli dokumentoitu arkkitehtuuritaso sekä rajapinnat, ja muuta dokumentaatiota tehtiin asiakkaan toiveen mukaisesti. Toisaalta eräässä yrityksessä todettiin dokumentaatiota syntyvän sopiva määrä, mutta sen olevan vääränlaista. Osassa yrityksistä oli todettu projektin aikainen dokumentointi tärkeäksi, josta osa sitten siirrettiin ylläpitovaiheen dokumentaatioksi. Muita tavoitteita dokumentaatiolle oli mahdollistaa kommunikointi dokumenttien avulla niin asiakkaan kuin muidenkin sidosryhmien kanssa.

5.2.18 Scrum ja laatu järjestelmät

Kuudessa yrityksessä oli käytössä ISO 9001-laatustandardin mukainen laatu järjestelmä [32]. Näistä yrityksistä neljä kertoi, ettei laatu järjestelmä vaikuta Scrumiin millään lailla, ja tyytyväisiä laatu järjestelmäänsä oli vain kaksi yritystä. Kaksi yritystä ei halunnutkaan laatustandardeja käyttöön, vaan näkivät niiden sotivan lähtökohdiltaan ketteryyttä vastaan ja että niiden hyödyt olisivat pieniä haittoihin nähden. Nähtiin kuitenkin laatustandardeista olevan etua julkisissa kilpailutuksissa.

Toisaalta eräässä yrityksessä todettiin Scrumin keventävän laatu järjestelmää, sillä enää ei tarvitse päivittää raskaita yksityiskohtaisia määrittelydokumentteja. Ainoastaan tuotteen työlistan tehtävät on käytävä läpi ja varsinainen työ tehdään aina niiden

tehtävien määrittelyjen mukaan. Kahdessa yrityksessä käytettiin tuotekehityksen kypsyysmallia, CMMI:tä [33]. Näissä yrityksissä CMMI-kypsyys oli tasolla kolme.

5.2.19 Valmiin tehtävän käsite

Milloin yksittäinen sovelluksen ominaisuus tai muu työlistan tehtävä on valmis? Scrum kehottaa määrittelemään asian selkeästi, mutta ei anna selvää vastausta kysymykseen. Haastatellut yritykset käyttivät kriteereinä mm. hyväksymistestausta, koodin ja dokumentoinnin valmistumista, yksikkötestien läpimenoa, läpi mennyttä koodin katselmointia, lokalisointien valmistumista sekä tiettyä rajaa yksikkötestikattavuudessa. Kuitenkaan kovin monella yrityksellä kriteereitä ei oltu määritelty selkeästi ja niin että ne olisivat kaikkien tiedossa, joten asiaan kaivattiin selkeytystä. Todettiin myös sen olevan tärkeintä, että tiimi tietää milloin tehtävää voi pitää valmiina. Lisäksi tärkeänä pidettiin, että kriteerit on sovittu yhdessä asiakkaan kanssa. Välillä tosin tiimi joutui itse keksimään määritelmät. Joillain yrityksillä kuului heti projektin käynnistymisen yhteydessä sopia mitkä ovat kriteerit valmiille työlle. Yhdellä yrityksellä oli jopa tehty yrityksen sisäinen listaus, josta pystyi hakemaan ja poimimaan erilaisia määritelmiä sille, mitä valmis tarkoittaa.

5.2.20 Käytetyt työkalut

Scrum-käyttöön kaivattiin yhtä kevyitä työkaluja kuin Post-it-laput seinällä, mutta jotka mahdollistaisivat paremmin hajautetun työskentelyn ja varmuuskopioinnin. Suurin osa käytti Scrum-työkaluinaan kuitenkin Post-it-lappuja, tussitauluja ja Excel-taulukkoita. Jotkut käyttivät myös erityisiä ohjelmistoja, kuten *ScrumWorks* tai *Agilefant* [34, 35]. Eräässä yrityksessä oli käytössä täysin itse tehty ohjelmisto Scrumia varten. Raskaiden työkalujen todettiin olevan toimimattomia pienillä organisaatioilla, esimerkiksi *Microsoft Project* koettiin olevan kaukana ketteryydestä [36]. Alla on listattuna yritysten käyttämiä ohjelmistoja:

- Scrum-työkaluina: *ScrumWorks*, *Agilefant* [34, 35].
- Kommunikointiin: *Office Communicator* [37].
- Vaatimuksienhallintaan: *CaliberRM*, *Polarion Requirements* [38, 39].
- Dokumentointiin: Wiki (*Confluence*, *Trac*), *Doxygen*, *Google Apps* [40, 41, 42, 43].
- Rautalankamallinnukseen: *Balsamiq Mockups* [44].
- Versionhallintaan: *Subversion* + *FishEye* [45, 46].
- Jatkuvaan integrointiin: *Hudson*, *CruiseControl* [47, 48].
- Build-automaatioon: *Maven* [49].
- Testaukseen: *JUnit*, *Jmeter* [50, 51].
- Ongelmien ja virheiden hallintaan: *JIRA*, *Trac*, *HP Quality Center* [52, 41, 53].
- Staattiseen analyysiin: *Sonar*, *StyleCop* [54, 55].
- Katselmointeihin: *Crucible* [56].

5.3 Scrum-menetelmän hyödyt

Scrum-menetelmän suurimpana hyötynä vaikutti olevan se, että tiimi koki ohjelmistoprojekteissa työskentelyn mielekkäämmäksi. Menetelmää kehitettiin tiimeille motivoivaksi, koska ihmiset pääsevät itse vaikuttamaan tekemiseensä, oma-aloitteisuus lisääntyi ja stressiä oli vähemmän. Scrumia kehitettiin tehokkaimmaksi tavaksi saada jotain aikaiseksi. Ylitöitä ei ollut enää niin paljon, mutta muuten kyllä työskentely oli hektistä. Iteratiivisuuden etuna oli se, että jokainen sprintti on uusi mahdollisuus, ja henkinen nollaus tapahtuu aina sprintin vaihtuessa. Kehittäjiä ei myöskään tarvitse miettiä tulevaisuutta kuin sprintin verran eteenpäin, jos eivät halua. Kehittäjät kokivat usein myös palkitsevana nähdä kehitystä joka sprintin jälkeen. Asiantuntijaorganisaatiossa Scrumilla saavutettavaa vapautta on vaikea nähdä noudattamalla tiukkoja rutiineja ja kurinalaisuutta.

Scrum oli tarjonnut monille yritykselle ensimmäisen kunnan kehityksen sovelluskehitykseen. Mikäli aikaisemmin ei ollut kunnan toimintatapoja, aikatauluarviot pettivät usein, sillä tiimin toimintaa ei välttämättä seurattu lainkaan. Scrum koettiin systemaattisena mutta yksinkertaisena mallina moneen yritykseen. Sillä saatiin ryhtiä etenkin tiedonkulkuun.

Scrum pakottaa kommunikointiin ja palautteenantoon, mikä nähtiin hyvänä. Lukuisat yhteiset palaverit, Daily Scrum ja lähekkäin istuminen lisäävät tiimin keskinäistä kommunikaatiota. Avoimuus kommunikaatiossa ja menetelmän joustavuus pitivät useimmat asiakkaat myös tyytyväisinä. Scrum tarjoaa selkeän rakenteen asiakkaan kanssa kommunikointiin ja asiakkaiden osallistumiseen prosessiin.

Projektipäälliköt ja tuoteomistajat kertoivat Scrumin myötä tietävänsä paremmin missä vaiheessa projektia ollaan menossa. Koska kaikki tehtävät ovat näkyvissä, lisää se varmuutta myös tiimille itselleen, ja kaikki menevät samaan suuntaan yhdessä. Scrumin myötä riskit ovat entistä paremmin ja aikaisemmassa vaiheessa näkyvissä. Tämä mahdollistaa aikaisemman puuttumisen ongelmakohtiin. Myös asiakkaat näkevät paremmin mitä on työn alla ja suunnitteilla, sekä pääsevät vaikuttamaan tulevien ominaisuuksien priorisointiin. Priorisointien avulla tekeminen kohdistuu oikeisiin asioihin. Scrumin myötä ohjelmakoodin refaktorointi ei ole enää mikään iso ongelma, vaan osa normaalia prosessia. Taulukkoon 5.16. on koottu Pirkanmaalaisten yritysten haastatteluista havaittuja Scrum-menetelmän hyötyjä, sekä verrattu niitä aiempiin taulukoihin 3.4. ja 2.3.

Harvalla yrityksellä oli esittää mitään konkreettista näyttöä siitä, että menetelmä todella toimii. Useimmilla oli kuitenkin jonkinlainen tuntuma siitä, että Scrumin myötä on tullut lisää tehokkuutta ja projektit olivat sujuneet paremmin. Eräällä yrityksellä oli kokemuksena, että edellinen tuote oli epäonnistunut täysin, kun taas Scrumin myötä uusi tuote oli saatu toimivana ulos. Toinen yritys kertoi, että epäonnistumisillaan ollut projekti saatiin takaisin raiteilleen Scrumiin siirtymällä. Eräs yritys totesi, että aiemmin työmääräarviot olivat enemmän pielessä ja henkilöstöllä oli enemmän aikataulustressiä, mutta Scrumin myötä ajanhallinta oli parantunut. Yksi yritys kertoi mitatun

asiakastyytyväisyyden parantuneen. Asiakastoimitukset saatiin useammin ajallaan ja dynaamisesti. Yksi yritys kertoi heillä raportoitujen ohjelmistovirheiden määrän pudonneen puoleen samalla kun projektin kesto lyhentyi viidellä kuukaudella.

Taulukko 5.16. Scrum-menetelmän hyödyt Pirkanmaalla

Tunniste	Haastatteluissa havaittu hyöty Scrumissa	Scrum-menetelmän hyödyt	Ketterät menetelmät yleisesti
HP01	Tiimi kokee työskentelyn mielekkäämmäksi ja on motivoitunut.	HS02, HS04	H07
HP02	Sprintit tarjoavat mahdollisuuden nähdä kehitystä säännöllisesti.	–	–
HP03	Scrum tarjoaa selkeän ja riittävän yksinkertaisen toimintamallin.	–	–
HP04	Scrum tehostaa avointa kommunikaatiota tiimin sisällä sekä asiakkaan kanssa.	HS01	H08
HP05	Scrum lisää avoimuutta, koska tiedetään miten projekti sujuu ja riskit ovat näkyvissä.	HS05	H04, H06
HP06	Työlistojen priorisointien avulla tekeminen kohdistuu oikeisiin asioihin.	HS05	–

5.4 Scrum-menetelmän ongelmat ja haasteet

Scrumin käyttöön liittyi myös monia ongelmakohtia ja haasteita, joita tuli hyvin esiin haastatteluissa. Taulukkoon 5.17. on koottu näitä ongelmia ja haasteita, sekä verrattu niitä aiempiin taulukoihin 3.5. ja 2.4. Yritykset kokivat monesti asiakkaiden (sisäisten tai ulkoisten) toimintatapojen ja odotusten sotivan Scrum-menetelmää vastaan. Asiakkaat eivät tuntuneet ymmärtävän ketterää ohjelmistokehitystä tai omaa rooliaan siinä. Tämä näkyi esimerkiksi niin, että saneltiin kiinteitä aikarajoja, joita ei sitten kyetty saavuttamaan edes Scrumilla. Tiimi ei pystynyt millään sitoutumaan tavoitteisiin, koska eivät olleet päässeet mukaan arvioimaan työmääriä. Asiakas saattoi myös jättää priorisoinnin täysin tekemättä, tai priorisoida aivan kaiken tärkeimmäksi. Näin ollen tiimi joutui itse pohtimaan priorisoinnit. Ohjelmistoyritykset eivät myöskään usein saaneet itse valita käyttämäänsä menetelmää, vaan asiakas saneli ne. Jotkut olivat ratkaisseet tämän tarjoamalla asiakkaalle vesiputousmallista kehitystä, vaikka oikeasti työ yritettiin sisäisesti tehdä Scrumilla.

Scrumin sovittaminen yrityksen omiin vanhoihin toimintatapoihin ja organisaatiokulttuuriin vaikutti myös haastavalta. Tämä saattoi johtaa siihen, että yhdellä ihmisellä oli useita päällekkäisiä rooleja, kuten projektipäällikkö, linjaesimies, ScrumMaster tai tuoteomistaja. Tämä johti sekaviin tilanteisiin, joissa vastuut olivat epäselviä. Myös sprintit jouduttiin usein sovittamaan esimerkiksi yrityksen kuukausiraportointijärjestelmään. Yrityksillä oli myös vaikeuksia sovittaa esimerkiksi muiden projektien kautta tulevia ylläpitotehtäviä tai virheiden korjauksia Scrum-prosessiin. Ne eivät näkyneet Scrumin työlistoissa, vaan tulivat useimmiten kaiken

muun työn päälle. Yritykset eivät myöskään aina osanneet tarjota sopivia toimitiloja tai työkaluja Scrum-tiimien käyttöön.

Johdon huono sitoutuminen Scrumiin nähtiin myös ongelmana. Scrumin käyttöä jouduttiin piilottelemaan tai peittelemään, koska ei uskallettu kertoa käytetyistä toimintatavoista. Perinteinen suomalainen autoritääriinen johtamiskulttuuri piti usein Scrumia liian epämääräisenä, ja vaikeasti johdettavana menetelmänä. Myös jotkut kehittäjät olivat halunneet jäykempää kontrollia, eivätkä sopeutuneet uusiin toimintatapoihin. Ääritapauksissa oli jopa vaihdettu yritystä tämän takia.

Taulukko 5.17. Scrum-menetelmän ongelmat ja haasteet Pirkanmaalla

Tunniste	Haastatteluissa havaittu ongelma / haaste Scrumissa	Ongelma / haaste Scrumissa	Ketterät menetelmät yleisesti
OP01	Asiakkaat eivät ymmärrä Scrum-menetelmää tai osaa sitoutua sen toimintatapoihin.	-	-
OP02	Scrumia on haastava sovittaa yrityksen jähmeään organisaatiokulttuuriin.	OS03	-
OP03	Johto ei ole sitoutunut Scrumiin tai pitää sitä epämääräisenä.	OS03	O07
OP04	Scrumin käyttöönotto on haastavaa, ja siihen täytyy varata paljon aikaa.	-	-
OP05	Tuoteomistajan rooli on haastava. Hän ei ehdi panostamaan projektiin tai sekaantuu liikaa tiimin tekemiseen.	OS02	-
OP06	Sprinttien rauhoittaminen on hankalaa, koska ylimääräiset ylläpitotehtävät ja virhekorjaukset häiritsevät	OS04	-

Scrumin käyttöönotto nähtiin myös todella haasteellisena. Yritykset kertoivat, että käyttöönotossa kannattaa varautua vastarintaan, eikä odottaa välittömiä tuloksia. Scrum ei toimi hetkessä, ja sitä on hyvin hankala ottaa käyttöön vakiintuneessa firmassa. Nuoremmat työntekijät adaptoituivat kuitenkin helposti. Yrityksissä saatettiin myös ymmärtää ketterät menetelmät väärin, esimerkiksi niin, että dokumentoinnista tai arkkitehtuurisuunnittelusta voitiin luopua. Myös liian joustavat työajat haittasivat Scrum-menetelmän käyttöä, sillä jotkut ihmiset eivät kyenneet osallistumaan yhteisiin palaveriin liian epäsäännöllisten työaikojen takia.

Tuoteomistajan rooli osoittautui monesti hankalaksi. Jos tuoteomistaja tuli asiakkaan puolelta, isona ongelmana oli tuoteomistajan työpanoksen vähäisyys. Heillä ei ollut aikaa panostaa projektiin, ja usein joku tiimin sisältä joutui ottamaan tuoteomistajan roolia, ja tekemään priorisointeja sekä ylläpitämään tuotteen työlistaa. Tuoteomistajan heikko kommunikaatio ja ohjaus tiimin suuntaan näkyivät yleensä suoraan sprintin heikkona tuloksena. Toisaalta jos tuoteomistaja tuli oman yrityksen sisältä, ongelmana saattoi olla se, että hän sekaantui liikaa tiimin tekemiseen. Ei osattu antaa tiimille tarpeeksi vapauksia ja vastuuta, vaan puututtiin herkästi sovelluksen

yksityiskohtiin. Toisaalta myöskään aina tiimi ei tuntunut tajuavan omaa vapauttaan päättää sprintin sisäisistä asioista.

Ongelmana oli myös se, että tiimin jäseniä määrättiin tekemään kiireellisiä korjauksia tai hoitamaan muita projekteja kesken sprintin. Sprintin rauhoittaminen koettiin hankalaksi, joka kertoo aika paljon sitoutumattomuudesta Scrumin periaatteisiin. Joissain yrityksissä laskettiin aina mukaan 15 % ylimääräisiä työtehtäviä jokaiseen sprinttiin yllättäviä ylläpitotehtäviä varten.

5.5 Scrum-suosituksia

Haastattelujen pohjalta pystyi keräämään oheisen listan hyvistä käytänteistä ja Scrum-suosituksista:

- Käytä omaa harkintaa Scrumin soveltamisessa. Pilkuntarkkaan ohjeiden mukaan meneminen ei ole tarpeellista. Scrum saa elää ja mukautua.
- Mene tarkemmin Scrum-sääntöjen mukaan, jos kyseessä on täysin uusi tiimi tai tuntematon sovellusalue.
- Luo Scrumiin oma sisäinen koulutuspaketti, joka sisältää perusteet, työtapaohjeet, toimii sisäisenä ohjenuorana kehitykseen, ja josta löytyy kootusti hyviä käytänteitä.
- Luo yrityksen sisäinen listaus, josta pystyy hakemaan ja poimimaan erilaisia määritelmiä sille, mitä valmis tehtävä Scrumissa tarkoittaa.
- Järjestä kuukausittaisia tietoiskuja henkilöstölle ketterän kehityksen aiheista.
- Aseta tiimien kooksi 5-9 henkeä.
- Mahdollista tiimien itseorganisoituminen. Se lisää motivaatiota ja mahdollistaa limittäisen osaamisen syntymisen.
- Osaavat ja noviisit tasapainottavat toisiaan Scrum-tiimeissä.
- Linjaesimiestä ei kannata asettaa ScrumMasteriksi, vaan valita joku tiimin sisältä.
- Lyhennä palautteenantocykliä käyttämällä kahden viikon mittaisia sprinttejä.
- Ota käyttöön tiimin oma sprintti, jossa he saavat välillä täysin itse päättää sprintin työlistan sisällön. Tiimin oman sprintti voi olla esimerkiksi joka kuudes sprintti, ja se on tapa viedä eteenpäin tiimin omia kehitysideoita.
- Luo dokumentaatiota projektissa sen verran, että mahdollistat kommunikaation dokumentaation avulla. Määrittele loppukäyttäjän kanssa tarkkaan haluttu dokumentaatio.
- Älä unohda arkkitehtuurisuunnittelua tai riittäviä määrittelyjä Scrumin myötä.
- Järjestä retrospektiivejä, ja ota projektin ulkopuolisia henkilöitä vetämään niitä. Jalkauta niistä saatuja parannusehdotuksia myös muille tiimeille.
- Automatisoi testausta ja buildien tekemistä. Jatkuvalle automaattisella yksikkötestauksella, muulla testausautomaatiolla ja automaattisilla buildeilla mahdollistat lyhyet sprintit.

- Pidä yläosa tuotteen työlistä aina valmiiksi analysoituna, ja palastele siitä osia sprintteihin toteutettavaksi.
- Pokerisuunnittelu on tiimille hauska tapa hoitaa työmääräarviointi. Käytä abstrakteja pisteitä työtuntien sijaan. Panosta pisteiden käytön kouluttamiseen ja harjoitteluun.
- Älä hajauta sovelluskehitystä.
- Jos kuitenkin hajautat sovelluskehitystä, käytä etänä työskenteleviä henkilöitä säännöllisesti paikan päällä. Tämä helpottaa kommunikointia jatkossa.
- Kerää henkilöstöltä säännöllisesti palautetta käytetyistä työtavoista, ja havainnoi trendiä.
- Mittaa asiakastyytyvää säännöllisesti, se kertoo onko työssä onnistuttu.
- Opasta sidosryhmiä Scrumin toiminnasta. Käytä selkeitä ja ymmärrettäviä termejä, äläkä hämää vaikeilla Scrum-termeillä.
- Asiakslähtöisyys ei ole keneltäkään pois. Koko toimialan etu olisi että ketterän kehityksen oppeja sovellettaisiin. Usein projekteissa on useita eri toimittajia mukana, joten olisi hyvä saada yhteisiä ketteriä toimintatapoja.

5.6 Tulosten luotettavuus

Haastatellut yritykset olivat keskenään hyvin erilaisia, joka luonnollisesti näkyi vastauksissa. Eroja oli esimerkiksi sulautettuja ohjelmistoja tekevien ja ainoastaan ohjelmistoprojekteja myyvien yritysten välillä. Näin ollen saatuja tuloksia ei voi suoraan yleistää koko toimialaa koskeviksi, vaan niistä on poimittava kunkin yrityksen toimintaympäristöön sovellettavissa olevat havainnot ja suositukset. Haastattelutuloksista löytyi kuitenkin helposti samansuuntaisia tai toisiaan tukevia havaintoja, jotka pystyttiin kokoamaan yhden otsikon alle. Tuloksia voi siis pitää melko luotettavina. Kuitenkin tässäkin tapauksessa saatujen tulosten tilastollinen luotettavuus kärsii siitä, että tutkittujen yritysten ja henkilöiden määrä oli niin pieni. Lisäksi joissain yrityksissä haastateltavat olivat niin kaukana ohjelmistoprojekteista, etteivät suoraan tieneet kaikkea yrityksessä olevasta käytännön ohjelmistotyöstä. Toisaalta taas tiiviisti projekteissa työskentelevät eivät aina osanneet vastata laajempia toimintatapoja koskeviin kysymyksiin. Todennäköistä on myös se, että tiedusteltaessa oman yrityksen toimintatapoja haastateltavat saattoivat kaunistella todellisuutta ja pyrkiä antamaan todellisuutta ruusuisemman kuvan yrityksen käytännöistä.

6. YHTEENVETO

Scrum ja muut ketterät menetelmät ovat tutkimuskohteena vielä uusi asia. Tässä työssä tuotettiin ennen kaikkea lisää tietoa Scrumin käytännön soveltamisesta yrityksissä. Lisäksi tunnistettiin Scrum-menetelmällä saavutettuja hyötyjä, kuten myös sen käytön ongelmakohtia ja haasteita. Tuloksena kertyi myös useita hyviä ketteriä käytäntöjä ja suosituksia käytännön ohjelmistoprojekteihin.

Työssä löydettiin ja analysoitiin systemaattisen artikkeliselvityksen avulla yhdeksän eri Scrum-tutkimusta, joista suurin osa käsitteli aloittelevia Scrum-tiimejä. Scrumin tutkimus vaikuttaa lisääntyvän vuosi vuodelta, ja odotettavissa onkin että jatkossa saadaan myös laajempia tuloksia käyttökokemuksista. Haastattelututkimuksessa selvitettiin yhdentoista Pirkanmaalaisen ohjelmistoyrityksen ketteriä käytäntöjä ja Scrum-menetelmän käyttöä tarkemmin. Löydetyistä Scrum-tutkimuksista sekä haastatteluista pystyi löytämään tuloksia sekä Scrum-menetelmän hyödyistä että siihen liittyvistä ongelmista ja haasteista. Yksi tärkeä tulos onkin se, että Scrumin käyttöön liittyy sekä positiivisia että negatiivisia puolia.

Haastatelluilla yhdellätoista yrityksellä oli kokemusta Scrumin käytöstä keskimäärin 1,5-3 vuotta. Moni yrityksistä kertoi että Scrum on heille ensimmäinen kunnan menetelmä ohjelmistotuotantoon. Haastatteluissa yrityksille suoritettiin myös Nokia-testi, jonka tuloksena saadaan arvo väliltä 1-10 kuvaamaan yrityksen ketteryyttä Scrumissa. Tutkituissa yrityksissä Nokia-testin keskiarvoksi muodostui 4,5 tulosten hajonnan ollessa 1,3 pisteestä 7,2 pisteeseen. Tämä kertoo yritysten omaksuneen Scrumin käytäntöjä hyvin vaihtelevasti. Vertailtaessa tarkemmin haastateltujen yritysten testin tuloksia kansainväliseen vertailuaineistoon huomattiin muutamia eroavaisuuksia. Havaittiin, että ketterässä määrittelyssä olisi haastatelluissa yrityksissä vielä paljon parannettavaa. Myös tuoteomistajan käyttäytymisessä ja hyödyntämisessä jäätiin selvästi jälkeen kansainvälistä vertailuaineistoa. Tuotteen työlistan hyödyntämisessä oltiin kuitenkin paljon pidemmällä kuin vertailuaineiston vastaajat. Häiriöitä tiimin työskentelyyn tuntuu kuitenkin haastatelluissa yrityksissä tulevan varsinkin päälliköiden tai projektinjohtajien tahoilta enemmän kuin vertailuaineistossa.

Aiempien ketterien menetelmien ja Scrum-tutkimusten, sekä tässä työssä tehdyn haastattelututkimuksen perusteella tunnistettiin useita Scrumin käyttöön liittyviä hyötyjä. Tiimeissä työskentelevät ihmiset kokevat työskentelyn mielekkäämmäksi sekä työtyytyväisyyden ja motivaation paranevan menetelmän joustavuuden myötä. Sosiaalinen työympäristö, yhteinen päätöksenteko ja palautteen anto lisäävät tiimin työelämän laatua. Toinen ehdottoman positiivinen seikka oli asiakastyytyväisyyden paraneminen avoimen kommunikaation kautta. Asiakkaat itse arvostavat aktiivista

osallistumista kehitysprosessiin, asiakas huomioidaan ja pidetään ajan tasalla sekä he voivat kontrolloida paremmin projektia. Lisäksi Scrum lisäsi avoimuutta ja läpinäkyvyyttä sekä auttoi saamaan paremman kokonaiskuvan kehitysprosessista. Tuotteen työlistan käyttö selkeytti projektia, yksittäiset tehtävät olivat hallittavissa olevan kokoisia ja riskit olivat paremmin näkyvissä. Työlistojen priorisointien avulla tekeminen myös kohdistuu oikeisiin asioihin. Yhteenvedona voisi sanoa Scrumin siis lisäävän työelämän, kommunikaation ja kehitysprosessin laatua.

Toisaalta työssä tunnistettiin myös useita Scrumin käyttöön liittyviä ongelmakohtia ja haasteita. Jähmeään yrityskulttuuriin ja organisaation vanhoihin toimintatapoihin on hankala sovittaa Scrumia tai muita ketteriä ohjelmistotuotantomenetelmiä. Myös yrityksen johdon huono sitoutuminen Scrumiin nähtiin myös ongelmana. Johtajat pitivät usein Scrumia liian epämääräisenä, ja vaikeasti johdettavana menetelmänä. Tiimien itsenäiseen työskentelyyn ei luotettu. Tuoteomistajan rooli havaittiin myös haastavaksi. Joko hän ei ehtinyt panostamaan projektiin tai sitten hän sekaantuu liikaa tiimin käytännön tekemiseen. Tiimin näkökulmasta sprinttejä oli hankala rauhoittaa työskentelylle. Sprintin työlistaa täytyi usein suojella muutosyrityksiltä kesken sprintin, koska ylimääräiset ylläpitotehtävät ja virhekorjaukset häiritsevät.

Haastatteluissa virinnyt vilkas keskustelu yrityksen omista toimintatavoista kertoo siitä, että yrityksiä pitäisi käydä enemmän ja useammin dialogia omista toimintatavoista ja niiden kehittämistä. Tässä diplomityössä tarjotaankin useita konkreettisia Scrum-suosituksia ja hyviä käytänteitä yrityksiä ketterien toimintatapojen kehittämiseksi.

LÄHTEET

- [1] Manifesto for Agile Software Development [WWW]. [viitattu 23.6.2010]. Saatavissa: <http://agilemanifesto.org/>
- [2] Abrahamsson, P., Warsta, J., Siponen, M. & Ronkainen J. New Directions on Agile Methods: A Comparative Analysis. 2003. 25th International Conference on Software Engineering (ICSE'03) pp. 244
- [3] Highsmith, J. 2002. Agile Software Development Ecosystems. Addison-Wesley, Boston. 404 p.
- [4] Highsmith, J. & Cockburn, A. 2001. Agile Software Development: The Business of Innovation. Computer. 34(2001), pp. 120-122.
- [5] Boehm B. 2002. Get Ready For The Agile Methods, With Care. Computer, 35(2002), pp. 64-69.
- [6] Fowler, M. & Highsmith, J. 2001. Agile methodologists agree on something. Software Development, 9(2001), pp. 28-32.
- [7] Abrahamsson, P., Salo, O., Ronkainen, J., Warsta, J. 2002. Agile software development methods. Review and analysis. VTT Publications, VTT Electronic, Espoo. 107 p.
- [8] Cockburn, A. 2002. Agile software development. Addison-Wesley. 278 p.
- [9] State of Agile Development Survey Results 2010 [WWW]. [viitattu 15.1.2011]. Saatavissa: http://www.versionone.com/state_of_agile_development_survey/10/default.asp
- [10] Nerur, S., Balijepally, V. 2007. Theoretical Reflections on Agile Development Methodologies, Comm. ACM, 50(2007)3, pp. 79–83.
- [11] Dybå, T. & Dingsøy, T. 2008. Empirical studies of agile software development: A systematic review. Information and Software Technology. 50(2008)9-10, pp. 833-859.
- [12] Petersen, K., Wohlin C. 2009. A comparison of issues and advantages in agile and incremental development between state of the art and an industrial case. Journal of Systems and Software, 82(2009)9, pp. 1479-1490

- [13] Schwaber, K. & Beedle, M. 2002. Agile Software Development with Scrum. Prentice Hall. 158 p.
- [14] Scrum (development), Wikipedia [WWW]. [viitattu 28.6.2010]. Saatavissa: http://en.wikipedia.org/wiki/Scrum_%28development%29
- [15] Schwaber K. 2004. Agile project management with Scrum. Microsoft Press. 192 p.
- [16] Burn down chart, Wikipedia [WWW]. [viitattu 5.7.2010]. Saatavissa: http://en.wikipedia.org/wiki/Burn_down_chart
- [17] Scrum Alliance [WWW]. [viitattu 28.8.2010]. Saatavissa: <http://www.scrumalliance.org/>
- [18] IEEE Xplore [WWW]. [viitattu 20.12.2010]. Saatavissa: <http://ieeexplore.ieee.org/>
- [19] ScienceDirect [WWW]. [viitattu 20.12.2010]. Saatavissa: <http://www.sciencedirect.com/>
- [20] RefWorks Home Page [WWW]. [viitattu 20.12.2010]. Saatavissa: <http://www.refworks.com/>
- [21] Greenhalgh, T. 2001. How to Read a Paper. 3. painos, Wiley-Blackwell. 229 p.
- [22] Kitchenham, B., Pfleeger, S., Pickard, L., Jones, P., Hoaglin, D., El Emam, K., Rosenberg, J. Preliminary guidelines for empirical research in software engineering. IEEE Transactions on Software Engineering, 28(2002)8, pp. 721–734.
- [23] Dingsoyr, T., Dyba, T., Abrahamsson, P. 2008. A Preliminary Roadmap for Empirical Research on Agile Software Development. Agile, 2008. AGILE '08. Conference, pp. 83-94.
- [24] Fenton, N. 2001. Viewpoint Article: Conducting and Presenting Empirical Software Engineering, Empirical Software Engineering, 6(2001)3, pp. 195-200.
- [25] Certified ScrumMaster [WWW]. [viitattu 19.9.2010]. Saatavissa: <http://www.scrumalliance.org/pages/CSM>
- [26] Certified Scrum Product Owner [WWW]. [viitattu 19.9.2010]. Saatavissa: http://www.scrumalliance.org/pages/certified_scrum_product_owner

- [27] ScrumButt Test – aka ”The Nokia Test” [WWW]. [viitattu 13.10.2010]. Saatavissa: <http://jeffsutherland.com/scrumbutttest.pdf>
- [28] Oliopäivä TTY:llä marraskuussa 2006 [WWW]. [viitattu 14.11. 2010]. Saatavissa: <http://www.cs.tut.fi/tapahtumat/olio2006/>
- [29] OO Days at Tampere University of Technology in November 2008 [WWW]. [viitattu 14.11.2010]. Saatavissa: <http://www.cs.tut.fi/tapahtumat/olio2008/>
- [30] The Scrum but Test [WWW]. [viitattu 7.11.2010]. Saatavissa: <http://antoine.vernois.net/scrumbut/?page=graph2&lang=en>
- [31] Beck, K. 2003. Test-Driven Development by Example. Addison-Wesley. 220 p.
- [32] ISO 9000 – Quality management [WWW]. [viitattu 20.12.2010]. Saatavissa: http://www.iso.org/iso/iso_catalogue/management_and_leadership_standards/quality_management.htm
- [33] CMMI [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.sei.cmu.edu/cmami/>
- [34] Scrum Tools - ScrumWorks Pro & ScrumWorks Basic [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.danube.com/scrumworks>
- [35] Agilefant - The simplest solution that actually works [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.agilefant.org/>
- [36] Microsoft Project | Project Professional 2010 | Project 2010 Server [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.microsoft.com/project/>
- [37] Microsoft Office Communicator 2007 R2 Product Overview [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://office.microsoft.com/en-us/communicator-help/microsoft-office-communicator-2007-r2-product-overview-HA010203715.aspx>
- [38] CaliberRM™ Enterprise Software Requirements Management System [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.borland.com/us/products/caliber/index.html>
- [39] Polarion Requirements : Integrated Web-based Requirements Management [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.polarion.com/products/requirements/index.php>

- [40] Confluence - Enterprise Collaboration and Wiki Software [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.atlassian.com/software/confluence/>
- [41] The Trac Project [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://trac.edgewall.org/>
- [42] Doxygen [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.doxygen.org/>
- [43] Google Apps for Business [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.google.com/apps/intl/fi/business/index.html>
- [44] Balsamiq Mockups [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://balsamiq.com/products/mockups>
- [45] Apache Subversion [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://subversion.apache.org/>
- [46] FishEye - Subversion, Git, Mercurial, CVS, Perforce & ClearCase in your browser [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.atlassian.com/software/fisheye/>
- [47] Hudson CI [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://hudson-ci.org/>
- [48] CruiseControl Home [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://cruisecontrol.sourceforge.net/>
- [49] Maven - Welcome to Apache Maven [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://maven.apache.org/>
- [50] JUnit.org Resources for Test Driven Development [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.junit.org/>
- [51] JMeter - Apache JMeter [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://jakarta.apache.org/jmeter/>
- [52] JIRA - Bug, Issue and Project Tracking for Software Development [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.atlassian.com/software/jira/>
- [53] HP Quality Center software [WWW]. [viitattu 12.12.2010]. Saatavissa: https://h10078.www1.hp.com/cda/hpms/display/main/hpms_content.jsp?zn=bto&cp=1-11-127-24^1131_4000_100__
- [54] Sonar [WWW]. [viitattu 12.12.2010]. Saatavissa: <http://www.sonarsource.org/>

[55] StyleCop [WWW]. [viitattu 12.12.2010]. Saatavissa:
<http://code.msdn.microsoft.com/sourceanalysis>

[56] Crucible - Code Review Tool [WWW]. [viitattu 12.12.2010]. Saatavissa:
<http://www.atlassian.com/software/crucible/>

LIITE 1 TIEDONKERUUKAAVAKE ARTIKKELEILLE

Tutkimusta koskevat tiedot	Selite
Tutkimuksen tunniste:	<i>Yksilöllinen id</i>
Tutkimuksen julkaisuvuosi:	<i>Vuosiluku</i>
Bibliografinen viite:	<i>Tekijät, vuosi, otsikko, lähdetieto</i>
Artikkelin tyyppi:	<i>Lehtiartikkeli / konferenssipaperi / muu</i>
Tutkimuksen tavoitteet:	<i>Kuvaus</i>
Tutkimustyyppi:	<i>kyselytutkimus / case-tutkimus / kirjallisuusarvio</i>
Tutkittavan ryhmän koko:	<i>Organisaatioiden / tiimien / henkilöiden lukumäärä / artikkelien määrä</i>
Tutkimuskohde:	<i>Minkä alan yritys, maa</i>
Projektin kesto:	<i>Kuinka laaja projekti, kesto / työtunnit</i>
Tutkimuskohteen Scrum kokemus:	<i>Kuinka kokeneita Scrum-kehittäjiä on tarkasteltu</i>
Tutkimuksen löydökset	
Hyvät kokemukset / positiiviset tulokset Scrumista:	<i>Tutkimuksen tuloksena tunnistetut positiiviset kokemukset Scrumista</i>
Ongelmia / haasteita Scrumissa:	<i>Tutkimuksen tuloksena tunnistetut negatiiviset kokemukset Scrumista</i>
Muut olennaiset löydökset ja johtopäätökset:	<i>Muut olennaiset löydökset ja havainnot Scrumiin liittyen</i>

LIITE 2 TUTKITUT ARTIKKELIT

[T01] Mann, C., Maurer, F. 2005. A Case Study on the Impact of Scrum on Overtime and Customer Satisfaction. Agile Development Conference (ADC'05), pp.70-79

[T02] Moe, N., Dingsøyr, T., Dybå, T. 2010. A teamwork model for understanding an agile team: A case study of a Scrum project. Information and Software Technology, 52(2010)5, pp. 480-491

[T03] Salo, O., Abrahamsson, P. 2008. Agile methods in European embedded software development organisations: a survey on the actual use and usefulness of Extreme Programming and Scrum. Software, IET, 2(2008)1, pp. 58-64

[T04] Moe, N., Dingsøyr, T., Dyba, T. 2009. Overcoming barriers to self-management in software teams. Software, IEEE, 26(2009)6, pp. 20-26

[T05] Hossain, E., Babar, M.A., Hye-young Paik. 2009. Using Scrum in Global Software Development: A Systematic Literature Review. Global Software Engineering, 2009. ICGSE 2009. Fourth IEEE International Conference on, pp. 175-184

[T06] Vlaanderen, K., Brinkkemper, S., Jansen, S., Jaspers, E. 2009. The Agile Requirements Refinery: Applying SCRUM Principles to Software Product Management. Software Product Management (IWSPM), 2009 Third International Workshop on, pp. 1-10

[T07] Moe, N.B., Aurum, A. 2008. Understanding Decision-Making in Agile Software Development a Case-study. Software Engineering and Advanced Applications, 2008. SEAA '08. 34th Euromicro Conference, pp. 216-223

[T07a] Moe, N.B., Dingsøyr, T., Dyba, T. 2008. Understanding Self-Organizing Teams in Agile Software Development. Software Engineering, 2008. ASWEC 2008. 19th Australian Conference on, pp. 76-85

[T07b] Moe, N.B., Dingsøyr, T., Kvangardsnes, O. 2009. Understanding Shared Leadership in Agile Development: A Case Study. System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on, pp. 1-10

[T08] Begel, A., Nagappan, N. 2007. Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study. Empirical Software Engineering and Measurement, 2007. ESEM 2007. First International Symposium on, pp. 255-264

[T09] Paasivaara, M., Durasiewicz, S., Lassenius, C. 2009. Using Scrum in Distributed Agile Development: A Multiple Case Study. Global Software Engineering, 2009. ICGSE 2009. Fourth IEEE International Conference on, pp. 195-204

LIITE 3 HAASTATTELUKYSYMYKSET

Yleistä

- Haastateltavan nimi ja yritys
- Yrityksen koko, yksikön nimi ja haastateltavan asema yksikössä?
- Onko yrityksessänne ennen TTY:n kurssia käytetty Scrumia tai muita ketteriä menetelmiä?
- Onko yrityksellänne ohjelmistokehitykseen jotain sertifikaatteja, kuten ISO 9001 tai CMMI jollain tasolla?
- Otitteko Scrumin käyttöön TTY:n koulutuksien jälkeen?
- Kuinka monta ihmistä tiimissänne on?
- Kuinka kokeneita ohjelmistokehittäjiä tiimissäsi on? Mikä on keskiarvo, minimi ja maksimi kokemusvuosista? Onko osa-aikaisia työntekijöitä?
- Minkä tyyppistä ohjelmistokehitystä teette (web, tietokanta, hajautettu, sulautettu jne.)?
- Kuinka monen kuukauden kokemus tiimilläsi on Scrumista?
- Ketkä tiimistänne osallistuivat TTY:n Scrum-koulutuksiin ja mihin koulutuksiin?
- Oletteko osallistunut joihinkin muihin Scrum-koulutuksiin, tai opiskelleet Scrumia muun materiaalin avulla?

Scrum-menetelmän noudattaminen (Nokia-testi)

- Onko iteraation pituus alle kuusi viikkoa? / Alkavatko ja päättyvätkö sprintit sovittuina päivämäärinä?
- Onko ohjelmisto täysin testattu ja toimiva iteraation päätteeksi?
- Voiko iteraatio alkaa ennen kuin määrittely on valmis?
- Tietääkö tiimi kuka on tuoteomistaja?
- Onko tuotteen työlista priorisoituna bisnesarvon perusteella?
- Estimoiko tiimi tuotteen työlistan?
- Tuottaako tiimi itse Burndown-kaavion ja tietääkö se nopeutensa?
- Onko tiimillä ulkopuolisia henkilöitä häiritsemässä tiimin työtä sprintin aikana?
- Onko tiimi itse-organisoituva?

Scrumin implementointi

- Millaisia erityisiä työkaluja yrityksessänne käytetään tukemaan Scrumia?
- Miten hoidatte testauksen Scrum-projekteissa? Onko testauksessa ollut ongelmia?
- Miten hoidatte dokumentoinnin Scrum-projekteissa? Onko dokumentointi riittävää? Millaisia dokumentteja teillä syntyy?

- Onko työssänne valmiin koodin ja dokumentoinnin välillä tasapaino?
- Onko tiimin työympäristö kunnossa? Työskenteleekö tiimi niin että voivat nähdä ja kuulla muut tiimin jäsenet?
- Onko teillä ollut tarvetta hajauttaa Scrum-tiimiä? Onko eri tiimien välillä vai yhden tiimin sisällä hajautusta? Millaisia kokemuksia tästä on?
- Onko teillä ollut alihankintaa mukana Scrum-projekteissa? Millaisia kokemuksia tästä on?
- Miten asiakkaat (sisäisten tai ulkoisten) ovat suhtautuneet Scrumiin?
- Onko asiakkaiden kanssa avointa ja yhteistyökykyistä kommunikaatiota?
- Onko asiakkaiden kanssa ollut sopimusongelmia Scrumin myötä?
- Onko käsite "done" määritelty sprintin työlistan vaatimuksille? Mitä se teidän projektissanne tarkoittaa?
- Onko Scrum ja ketterät menetelmät mielestäsi yhteensopiva nykyisen organisaatiokulttuurinne kanssa?
- Onko teillä täysi johdon tuki Scrumin käyttöön? Onko johto aktiivisesti edistänyt Scrumin käyttöönottoa?
- Onko johto asettanut tavoitteet Scrumin käyttöönotolle? Onko näitä tavoitteita seurattu?
- Onko Scrum-projektien johtaminen eronnut jollain tapaa muiden projektien johtamisesta? Onko ollut ongelmia?
- Järjestetäänkö retrospektiivejä iteraatioiden jälkeen ja mikä niiden vaikutus on?
- Onko työntekijöiltä kerätty palautetta Scrumin käytöstä? Millaista se on ollut?
- Näetkö että kehitystiimillä on tarpeeksi valtuuksia toteuttaa Scrum-kehitystä?
- Miten koet kommunikaation muuttuneen Scrumin myötä?

Yhteenveto

- Onko käyttämänne Scrum-metodi tehokas ja tuloksellinen tapa sovelluskehitykseen?
- Onko teillä kerätty jotain konkreettista näyttöä Scrumin toimivuudesta yrityksessänne?
- Mikä on ollut palkitsevinta Scrum -kehityksessä? Missä olette onnistuneet parhaiten?
- Missä olette epäonnistuneet?
- Koitko että Scrum on yritykselle sopiva tapa sovelluskehitykseen?
- Millaisia terveisiä / ohjeita / neuvoja teillä on Scrumin käyttöönottoa harkitseville?
- Suositteletko Scrum-kurssin käymistä muillekin? Miksi / miksi ei?

LIITE 4 NOKIA-TESTIN KYSYMYSTEN PISTEYTYS

Kysymysnro	Kysymys / Pisteytys	Pisteet
12.	Iteraatiot	
	Ei iteraatioita	0
	Iteraatiot yli 6 viikkoa	1
	Vaihtelevan pituiset iteraatiot alle 6 viikkoa	2
	Kiinteämittaiset 6 viikon iteraatiot	3
	Kiinteämittaiset viiden viikon iteraatiot	4
	Kiinteämittaiset iteraatiot, neljä viikkoa tai alle	10
13.	Testaus	
	Ei testausta iteraatioissa	0
	Yksikkötestattu	1
	Toiminnallisuudet testattu	5
	Toiminnallisuudet testattu heti kun ovat valmiita	7
	Ohjelmisto läpäisee hyväksymistestit	8
	Ohjelmisto otetaan käyttöön	10
14.	Ketterä määrittely	
	Ei määrittelyjä	0
	Iso määrittelydokumentaatio	1
	Heikkoja käyttäjätarinoita	4
	Hyvät vaatimukset	5
	Hyvät käyttäjätarinat	7
	Tarpeeksi määrittelyjä, ajoissa	8
	Hyvät käyttäjätarinat sidottuna määrittelyihin tarpeen mukaan	10
15.	Tuoteomistaja	
	Ei tuoteomistajaa	0
	Välinpitämätön tuoteomistaja	1
	Häiritsevää tuoteomistajaa	2
	Ulkopuolinen tuoteomistaja	2
	Tuoteomistaja, jolla on selkeä tuotteen työlista, joka on estimoitu tiimin toimesta ennen sprintin suunnittelutapaamista	5
	Tuoteomistaja, jolla on tiimin nopeuteen perustuva julkaisuaikataulu	8
	Tuoteomistaja, joka motivoi tiimiä	10
16.	Tuotteen työlista	
	Ei tuotteen työlistaa	0
	Useita tuotteen työlistoja	1
	Yksi tuotteen työlista	3
	Tuotteen työlista, jossa selkeästi määritellyt ja ROI:n perusteella järjestetyt työt ennen sprintin suunnittelutapaamista	5
	Tuoteomistaja, jolla on tuotteen työlistaan perustuvat julkaisuaikataulu	7
	Tuoteomistaja voi mitata ROI:n tuoton, kustannuksien, tai muiden metriikoiden avulla	10

17.	Työmääräarviot	
	Tuotteen työlistan töitä ei ole estimoitu	0
	Tiimi ei tuota estimaatteja	1
	Tiimi estimoit, mutta ei pokerisuunnittelulla	5
	Tiimi estimoit käyttäen pokerisuunnittelua	8
	Estimaattien virhe on alle 10 %	10
18.	Burndown-kaaviot	
	Ei Burndown-kaaviota	0
	Tiimi ei päivitä Burndown-kaaviota	1
	Burndown-kaavio tunneissa / päivissä, ei jäljellä olevan työmäärän mukaan	2
	Burndown-kaavioon merkitään vaan tehtävät jotka ovat valmiita	4
	Burndown-kaavio on valmis vain kun tarina on valmis	5
	Tiimi tietää oman nopeutensa	+3
	Tuoteomistajan julkaisusuunnitelma perustuu tunnettuun nopeuteen	+2
19.	Tiimin häiriötekijät	
	Päällikkö / projektinjohtaja häiritsee tiimiä	0
	Tuoteomistaja häiritsee tiimiä	1
	Päälliköt, projektinjohtajat tai tiiminjohtajat määräävät tehtäviä	3
	Projektinjohtaja nimetty ja sen lisäksi Scrumin roolit ovat käytössä	5
	Kukaan ei häiritse tiimiä, ja ainoastaan Scrumin roolit ovat käytössä	10
20.	Tiimi	
	Tehtävät asetetaan yksilöille sprintin suunnittelukokouksessa	0
	Tiimin jäsenten osaaminen ei ole lainkaan limittynyt	0
	Ei itseohjautuvuutta johtamisessa – yksi tai useampi tiimin jäsen osoitettu ohjaavaksi auktoriteetiksi	1
	Tiimillä ei ole riittävä osaamista	2
	Tiimi sitoutuu yhteisesti sprintin tavoitteisiin ja työlistaan	7
	Tiimin jäsenet taistelevat yhdessä esteitä vastaan sprintin aikana	9
	Tiimi on hypertuottavassa tilassa	10