

Yhteisen tietopohjan ylläpidon välineet ketterissä ohjelmistoprojekteissa

Ville Hämäläinen

Tampereen yliopisto
Viestintätieteiden tiedekunta
HTI-maisteriohjelma
Pro gradu -tutkielma
Ohjaaja: Saira Ovaska
Kesäkuu 2017

Tampereen yliopisto
Viestintätieteiden tiedekunta
HTI-maisteriohjelma
Ville Hämäläinen: Yhteisen tietopohjan ylläpidon välineet ketterissä
ohjelmistoprojekteissa
Pro gradu -tutkielma, 63 sivua, 2 liitesivua
Kesäkuu 2017

Ketterä ohjelmistokehitys syntyi vastauksena yritysmaailman nopeasti muuttuviin tarpeisiin. Ketterät menetelmät painottavat tehokasta epämuodollista viestintää ja yhteistyötä asiakkaan kanssa. Tutkielmassa tarkastellaan, millä välineillä ketterissä projekteissa ylläpidetään asiakkaan ja kehitystiimin välillä yhteistä tietopohjaa, eli yhteisesti jaettua käsitystä tarpeista ja projektin tilanteesta, ja miten ne soveltuvat tarkoitukseensa.

Tutkielma on usean tapauksen tapaustutkimus, jonka aineisto on kerätty kolmesta ohjelmistoyrityksestä puolistrukturoiduilla haastatteluilla. Haastatteluihin saatiin rekrytoitua kuusi ohjelmistoprojekteissa eri tehtävissä toiminutta henkilöä. Heitä pyydettiin miettimään yhtä esimerkkiprojektia haastattelua varten.

Haastateltujen esittelemät projektit olivat hyvin erilaisia jopa saman yrityksen sisällä, joten niitä oli mielekästä tarkastella omina tapauksinaan. Aineiston analyysivaiheessa haastatteluissa esiin nousseiden viestintämedioiden soveltuvuutta viestintätarpeisiin arvioitiin median synkronisaation teorian pohjalta. Viestinnällä oli suuri rooli jokaisessa projektissa ja yhdessäkään projektissa ei tyydytty käyttämään vain yhtä viestintämediaa.

Haastateltujen esittelemissä projekteissa käytettyjen viestintämedioiden käyttötavat noudattelivat pääasiassa median synkronisaation teorian esittämiä suosituksia siitä, millaisiin viestintätarpeisiin mitkään viestintämediat sopivat. Muutamia poikkeuksia näihin suosituksiin ilmeni. Esimerkiksi kiireellisissä tilanteissa käytettiin korkean synkronisaation viestintämedioita viestintätarpeisiin, joihin matalan synkronisaation mediat olisivat teorian mukaan sopineet paremmin. Aikataulupaine ajoi muiden tarpeiden edelle ja vastausten saamista nopeasti pidettiin tärkeämpänä kuin viestin parempaa ymmärrettävyyttä.

Avainsanat ja -sanonnat: *Ketterä ohjelmistokehitys, asiakasviestintä, ryhmäohjelmat, yhteinen tietopohja, median ilmaisuvoimaisuuden teoria, median synkronisaation teoria*

Sisällysluettelo

1. Johdanto.....	1
2. Keskeiset käsitteet.....	4
2.1. Ketterä ohjelmistokehitys.....	4
2.2. Asiakas ja asiakkaan rooli.....	6
2.3. Yhteinen tietopohja.....	8
2.4. Viestintä ketterissä ohjelmistoprojekteissa.....	10
2.5. Viestintävälineen valinta.....	12
2.5.1 Median ilmaisuvoimaisuuden teoria.....	13
2.5.2 Median synkronisaation teoria.....	14
2.5.3 Eri medioiden ominaisuuksia.....	17
2.6. Scrum, Extreme Programming ja Kanban.....	18
2.6.1 Scrum.....	19
2.6.2 Extreme Programming.....	20
2.6.3 Kanban.....	21
2.6.4 Yhteenveto menetelmistä.....	22
3. Tutkimusasetelma.....	25
3.1. Usean tapauksen tapaustutkimus.....	25
3.2. Aineistonkeruu.....	26
3.3. Tutkimukseen osallistuneet yritykset ja haastateltavat.....	27
4. Kohdeprojektien kuvaukset.....	30
4.1. Projekti 1 – Web-sovellus.....	30
4.2. Projekti 2 – Kansainvälinen konsulttiprojekti.....	32
4.3. Projekti 3 – Kolmen kehitystiimin ohjaus.....	33
4.4. Projekti 4 – Lahoava järjestelmä vaihtoon.....	34
4.5. Projekti 5 – Perustukset kuntoon.....	36
4.6. Projekti 6 – Vanhan järjestelmän päivitys ja korvaus.....	37
5. Tulokset.....	38
5.1. Yritysten menetelmävalinnat.....	38
5.2. Projekteissa käytetyt sähköiset työkalut.....	40
5.2.1 Skype ja Lync.....	40
5.2.2 Flowdock ja Slack.....	41
5.2.3 Trello ja Jira.....	41
5.2.4 Confluence.....	42
5.2.5 Google Drive ja SharePoint.....	42
5.2.6 Käytettyjen viestintämedioiden ilmaisuvoimaisuus.....	42
5.3. Projektien ajallisen keston vaikutus viestintään.....	43
5.4. Asiakkaan tavoitettavuus.....	46
5.5. Asiakasviestinnästä tehtyjä havaintoja.....	48
6. Pohdinta.....	51
6.1. Projektien viestintätilanteet ja -välineet.....	51
6.2. Projekteissa ilmenneet aikatauluongelmat.....	54
6.3. Pohdintaa menetelmästä.....	55
7. Yhteenveto.....	58
Lähdeluettelo.....	60
Liite A – Esitietolomake.....	64
Liite B – Haastatteluskripti.....	65

1. Johdanto

Ohjelmistokehitys on yhteistyöprosessi, jossa tarvitaan teknistä osaamista, asiantuntemusta ohjelmiston tarvitsevan asiakkaan toimialalle suunniteltavan ohjelmiston käyttötilanteista ja tarpeista sekä prosessituntemusta. Perinteisessä ohjelmistokehityksessä nämä osa-alueet on jaettu eri kehitysvaiheissa toimiville rooleille ja tietämys välitetään vaiheesta toiseen siirryttäessä dokumenteilla, kuten vaatimusmäärittelyt, lähdekoodi ja testaus suunnitelmat [Chau and Maurer 2004]. Moderni yritysmaailma edellyttää kuitenkin ohjelmistokehitykseltä kykyä vastata epävarmoihin ja muuttuviin vaatimuksiin sekä teknologioihin [Niinimäki *et al.* 2009].

Nopeasti muuttuvien ympäristöjen ja epävarmojen projektien parissa toimivissa organisaatioissa epävirallinen viestintä on erityisen tärkeää. Muodollisilla viestintätavoilla, kuten määrittelydokumenteilla, on vaikea vastata tarpeeksi nopeasti muuttuviin vaatimuksiin. Uutiset muutoksista leviävät usein epävirallisia reittejä kehittäjien välillä [Herbsleb *et al.* 2000].

Tilanteissa, joissa asiakkaalla ei ole mahdollisuutta irrottaa asiantuntijoitaan ohjelmistoprojektin tarpeisiin, voidaan viestinnässä hyödyntää *ryhmäohjelmia* (groupware) fyysisen läsnäolon tarpeen vähentämiseksi. Ryhmäohjelmilla viitataan työkaluihin, jotka on suunniteltu auttamaan käyttäjäryhmää viestimään, koordinoimaan ja toimimaan yhdessä yhteisen tavoitteen saavuttamiseksi. Ryhmäohjelmat avustavat ryhmiä kasvokkaisessa vuorovaikutuksessa, mutta myös eri toimipisteisiin hajautettujen ryhmien välisessä viestinnässä [Arora and Goel 2012].

Tässä tutkielmassa kartoitan, *miten ketterien menetelmien projekteissa ylläpidetään yhteistä tietopohjaa asiakkaan ja kehitystiimin välillä*. Yhteisellä tietopohjalla tarkoitan tässä tutkielmassa muun muassa yhteisymmärrystä asiakkaan ja kehitystiimin välillä siitä, mitä ollaan kehittämässä, mitä tarkoitusta varten, missä aikataulussa ja kenen toimesta. Esittämästäni tutkimuskysymyksestä irrotan tarkasteltavaksi kolme erillistä kysymystä:

1. *Millaisia viestintätarpeita ketterissä projekteissa esiintyy asiakkaan ja kehitystiimin välillä?*

2. *Millaisia viestintävälaineitä eli viestintämedioita näihin tarpeisiin käytetään?*
3. *Miten hyvin valitut viestintävälaineet sopivat viestintätarpeisiin, kun niitä tarkastellaan taustateorian valossa?*

Tutkielman keskeiset käsitteet liittyvät viestintään ja yhteisen tietopohjan ylläpitämiseen ketterissä ohjelmistoprojekteissa. Keskeiseen käsitteistöön voidaan lukea myös erilaiset ketterät menetelmät, sillä tämän tutkielman lähtökohtana on ketteryys. Tutkielmassa perehdytään luvussa 2 kolmeen yleisimpään ketterän ohjelmistokehityksen menetelmään ja niiden tapaan käsitellä asiakasviestintää.

Tarkastelen kolmesta tamperelaisesta ohjelmistoyrityksestä yksilöhaastatteluilla keräämäni aineistoa tutkimusartikkeleista saadun taustaymmärryksen valossa. Kysymyksessä on siis vertaileva tapaustutkimus, jonka aineisto on kerätty yksilöhaastatteluilla.

Tapaustutkimuksessa tutkimuksen kohteena on yksi tai useampi tapaus [Järvinen ja Järvinen 1996, 53]. Näistä yksittäisistä tai useammista tapauksista pyritään saamaan yksityiskohtaista tietoa [Saarela-Kinnunen ja Eskola 2010, 190]. Tapaustutkimuksessa tavoitellaan siis kapeaa, mutta syvälle menevää otosta tutkittavasta ilmiöstä. Tapaustutkimus itsessään ei määrittele, onko kerättävä aineisto kvantitatiivista vai kvalitatiivista, vaan sen määrittelee valitut tiedonhankintatavat, kuten kyselyt tai haastattelut [Järvinen ja Järvinen 1996, 53].

Tapaustutkimuksen aineistonkeruussa hyödynnetään tavallisesti useaa eri menetelmää tarkoituksenaan kuvailla tutkittavaa ilmiötä, joten tapaustutkimusta ei voi pitää pelkkänä metodina aineistonkeruuseen [Saarela-Kinnunen ja Eskola 2010, 190]. Tavoitteena on usein tilastollisen yleistämisen sijaan laajentaa ja yleistää jo olemassaolevaa teoriaa [Järvinen ja Järvinen 1996, 55].

Kuvaan tutkielman keskeisiä käsitteitä ja teorioita luvussa 2. Esittelen, mitä ketterä ohjelmistokehitys on ja miten se eroaa perinteisistä menetelmistä. Asiakas on ensisijaisesti ohjelmistotuotteen tilaaja sekä kehitystyön maksaja. Asiakas voi olla kehitettävän ohjelmistotuotteen loppukäyttäjä, mutta toisinaan asiakas voi olla pelkästään loppukäyttäjien etua edustava taho. Avaan luvussa 2 asiakkaan roolia ketterissä projekteissa.

Esittelen yhteisen tietopohjan määritelmän, josta jatkan ketterien menetelmien viestintäkäytäntöihin. Tarkastelen myös viestintämedian valintaan ja niiden arviointiin liittyviä teorioita: median ilmaisuvoimaisuuden teoriaa sekä median synkronisaation teoriaa. Näistä kahdesta hyödynnän median synkronisaation teoriaa keräämäni aineiston analyysivaiheessa. Esittelen vielä kolme yleisintä ketterän ohjelmistokehityksen

menetelmää paremman käsityksen saamiseksi siitä, miten eri tavoin voi tehdä ketterää ohjelmistokehitystä.

Luvussa 3 kerron tämän tutkielman tutkimusasetelmasta. Valotan usean tapauksen tapaustutkimuksen taustoja ja esittelen, miten aineisto kerättiin tätä tutkielmaa varten. Kuvailen myös tutkielmaan osallistuneen kolmen yrityksen sekä haastateltujen taustatietoja. Osallistujien esittelemät projektit ja niihin liittyviä taustatietoja olen esitellyt omassa luvussaan.

Haastatteluissa nousi esille, ettei edes saman yrityksen sisältä löytynyt etukäteen määriteltyjä menetelmiä tai valikoimaa viestintävälineitä vaan valinnat olivat aina projektien tarpeiden sanelemia. Paljastui myös, että projektien ajallinen kesto vaikutti viestintään asiakkaiden ja kehitystiimien välillä.

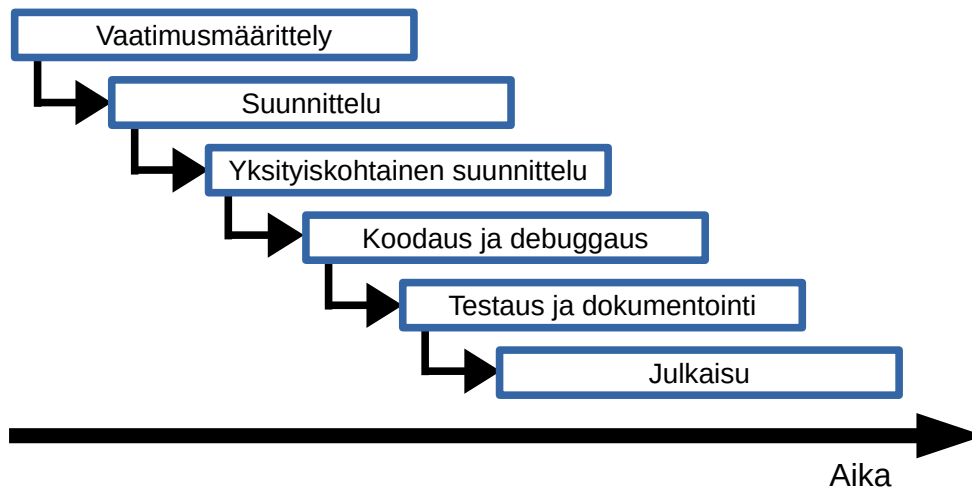
Kahdessa viimeisessä luvussa vedän saamiani tuloksia yhteen, vastaan esittämiini tutkimuskysymyksiin ja pohdin, miten haastatteluista saamani tulokset pitävät yhtä luvussa 2 esittelemieni viestintään liittyvien teorioiden kanssa. Arvioin myös, mitä tässä tutkielmassa olisin voinut tehdä paremmin ja mitä mahdollisia jatkotutkimuskohteita tämän tutkielman aiheella on.

2. Keskeiset käsitteet

Tässä luvussa tarkastelen aluksi, kuka on asiakas ja mikä on asiakkaan rooli. Asiakas ei välttämättä aina ole kehitettävän ohjelmistotuotteen loppukäyttäjä [Coplien and Bjørnvig 2010], joten käyn läpi, mikä asiakkaan rooli on yleisesti ketterän kehityksen projekteissa. Esittelen myös tutkielman aiheen kannalta tärkeimmän termin eli kuvaan yhteistä tietopohjaa ja sen merkitystä ohjelmistoprojektien onnistumisen kannalta. Yhteisen tietopohjan saavuttaminen riippuu viestinnän onnistumisesta, joten käsittelen myös viestintää ketterissä projekteissa. Lopuksi esittelen kolme yleisintä ketterän ohjelmistokehityksen menetelmää ja vertaan, miten asiakasviestintä ja yhteisen tietopohjan ylläpito niissä perinteisesti hoidetaan.

2.1. Ketterä ohjelmistokehitys

Vesiputousmalli on esimerkki perinteisistä ohjelmistokehityksen menetelmistä. Siihen viitataan usein, kun halutaan vertailla ketterien ja perinteisten menetelmien eroja. Royce [1987] esitteli 1970-luvulla suurten ohjelmistoprojektien projektinhallintaa käsittelevässä artikkelissaan vaiheittaisesti etenevän mallin, jossa edettiin lineaarisesti kehitysvaiheesta toiseen kohti ohjelmistotuotteen julkaisua. Kuvassa 1 on kuvattu tällainen kehitysmalli, johon nykyisin viitataan vesiputousmallina. Vesiputousmallin projektit kohtaavat suurimmat riskit vasta projektin loppuvaiheessa, koska testaus ja integrointi muihin järjestelmiin tapahtuu vasta projektin lopulla [Whitaker 2010, s. 252-254].



Kuva 1: Vesiputouksmalli [Royce 1987]

Ketterässä ohjelmistokehityksessä (agile software development) painotetaan yhteistyötä asiakkaan kanssa, tehokasta viestintää ja yksilöiden välistä vuorovaikutusta [Beck *et al.* 2001]. Taulukossa 1 on esitetty ketterän ohjelmistokehityksen julistus. Ideaalitulanteessa asiakkaan edustaja työskentelee samassa tilassa kehitystiimin kanssa ja on tällä tavoin välittömästi tarjoamassa tietoa asiakkaan toimialasta ja toimintatavoista tarpeen vaatiessa [Niinimäki *et al.* 2009].

Ketterän ohjelmistokehityksen julistus

Löydämme parempia tapoja tehdä ohjelmistokehitystä, kun teemme sitä itse ja autamme muita siinä. Kokemuksemme perusteella arvostamme:

yksilöitä ja kanssakäymistä	enemmän kuin	<i>menetelmiä ja työkaluja</i>
toimivaa ohjelmistoa	enemmän kuin	<i>kattavaa dokumentaatiota</i>
asiakasyhteistyötä	enemmän kuin	<i>sopimusneuvotteluja</i>
vastaamista muutokseen	enemmän kuin	<i>pitäytymistä suunnitelmassa</i>

Jälkimmäisilläkin asioilla on arvoa, mutta arvostamme ensiksi mainittuja enemmän.

Taulukko 1: Ketterän ohjelmistokehityksen julistus [Beck *et al.* 2001]

Perinteiset ohjelmistokehityksen menetelmät keskittyvät suunnitteluun etukäteen pyrkien minimoimaan tuotteen eri osien riippuvuudet, jolloin myös koordinaation ja viestinnän tarve vähenee projektin aikana [Chau and Maurer 2004]. Taulukossa 2 vertaillaan perinteisiä ja ketteriä menetelmiä keskenään. Perinteisten menetelmien näkökulmasta muutokset ovat ongelmia, jotka venyttävät aikataulua ja kasvattavat

kuluja. Yritysmailman vaatimusten ja perinteisten ohjelmistokehitysmenetelmien välillä on siis ristiriita. Selkein ero ketterien ja perinteisten menetelmien välillä on suhtautuminen muutoksiin. Ohjelmistokehitys yleisesti tähtää tuottamaan kustannustehokkaita ja laadukkaita ohjelmistotuotteita asiakkaille, mutta ketterät menetelmät pyrkivät vastaamaan asiakkaiden nopeasti muuttuviin tarpeisiin, kun taas perinteiset menetelmät yrittävät pitää muutokset minimissään [Arora and Goel 2012].

	Perinteiset menetelmät	Ketterät menetelmät
<i>Vaatimusmäärittelyt</i>	Runsaasti tarkkaan määriteltyjä vaatimuksia	Vaatimukset kuvataan käyttäjätarinoina
<i>Aikataulut</i>	Etukäteen määritelty, tavallisesti tarkka annettu päivämäärä projektin valmistumiselle	Projekti sisältää tietyn määrän kehitysjaksoja
<i>Muutoksiin sopeutuminen</i>	Olematonta, muutokset ovat ongelmia, jotka voivat sotkea aikatauluja	Kehitetty vastaamaan asiakkaan muuttuviin tarpeisiin
<i>Asiakkaan osallistuminen</i>	Projektin loppuksi, yleensä testausvaiheessa	Asiakas on mukana koko projektin ajan

Taulukko 2: Perinteisten ja ketterien menetelmien eroja [Whitaker, 2010, s. 251-267]

Ketterille menetelmille ominaista on asiakkaan aktiivinen rooli ja erityistä on idea samassa tilassa toimivasta asiakkaasta, jonka kanssa viestitään päivittäin. Yhteistyöstä asiakkaan kanssa on puhuttu aikaisemminkin käyttäjäkeskeisessä suunnittelussa (user-centered design), osallistuvassa suunnittelussa (participatory design) ja yhteistyössä tehdyssä järjestelmäsuunnittelussa (collaborative systems modeling). Yhteistyöllä asiakkaan kanssa on pyritty lisäämään asiakastyytyvää. [Bakalova and Daneva 2011]

2.2. Asiakas ja asiakkaan rooli

Asiakas on ensisijaisesti ohjelmistotuotteen tilaaja, joka maksaa ohjelmiston kehittämisestä [Coplien and Bjørnvig 2010]. Tämä tarkoittaa, ettei asiakas aina ole ohjelmistotuotteen loppukäyttäjä. Asiakas voi myös olla joku, joka tarjoaa valmiita ohjelmistotuotteita loppukäyttäjille tulevaisuudessa [Coplien and Bjørnvig 2010]. Tällöin asiakas on tilaaja, joka edustaa varsinaista loppukäyttäjää. Hyvä esimerkki tällaisesta tapauksesta on esimerkiksi korkeakoulun tietohallinto, joka tilaa opetuksen

tai hallinnon käyttöön ohjelmistotuotteen ulkopuoliselta toimittajalta. Näin ollen asiakkaalla saattaa olla suurempi tarve ohjelmistotuotteen valmistumiseen ajallaan kuin kaikkien ominaisuuksien löytymiseen ensimmäisestä tuotantoversiosta.

Ohjelmistotuotteen onnistumisen kannalta oleellinen tieto toimintamalleista ja toimialasta on asiakkaalla. Ketterän ohjelmistokehityksen menetelmissä luotetaan epäviralliseen ja aktiiviseen kasvokkaiseen vuorovaikutukseen asiakkaan kanssa. Perinteisissä menetelmissä taas viestitään asiakkaan kanssa vain tarpeen vaatiessa [Korkala *et al.* 2010]. Asiakkaan ja kehittäjien välisessä kasvokkaisella vuorovaikutuksella minimoidaan informaatiokatkosten riskit ja varmistetaan vastaukset kehittäjien kysymyksiin [Arora and Goel 2012].

Ominaisin piirre ketterille ohjelmistokehityksen menetelmille on asiakkaan aktiivinen ja jatkuva osallistuminen läpi kehityskaaren. Asiakkaan vastuulla on tarjota tietoa tarvittaessa ja tehdä liiketoiminnallisin perustein päätökset ohjelmistotuotteen vaatimuksista sekä kehitettävien ominaisuuksien tärkeysjärjestyksistä [Bakalova and Daneva 2011]. Asiakkaan vastuulla on siis toimittaa asiakkaan kannalta oikeat vaatimukset silloin, kun niitä tarvitaan [Hochmüller 2011]. Tiivis yhteistyö asiakkaan kanssa mahdollistaa toiminnallisuuksien arviointia useammin, jolloin myös vähennetään riskiä kehittää toiminnallisuuksia, joita asiakas ei arvosta [Holmström Olsson and Bosch 2015].

Riippumatta siitä, onko asiakas kehitettävän ohjelmistotuotteen loppukäyttäjä vai pelkästään tuotteen tilaaja, asiakkaan vastuulla on toimia loppukäyttäjien edustajana. Asiakkaan vastuulla on huolehtia, ettei projektin aikana kehitetä ohjelmistotuotteeseen asiakkaan kannalta turhia ominaisuuksia. Niinpä ketterissä menetelmissä luotetaan lyhyisiin kehitysjaksoihin, jotka mahdollistavat nopean palautteen saamisen asiakkaalta tehdystä kehitystyöstä ja näin asiakas pystyy myös vahvistamaan kehityksen etenevän halutulla tavalla [Beyer 2010].

Beyer [2010] huomauttaa, että ketterissä menetelmissä ei tehdä eroa loppukäyttäjän, tuotteen tilaajan eikä muidenkaan toimijoiden välillä. Oletuksena on, että jos kehitystiimi pystyy vastaamaan asiakkaan esittämiin vaatimuksiin, kehitettävä ohjelmistotuote tulee vastaamaan myös loppukäyttäjien tarpeita. Beyer korostaa tästä syystä käyttäjäkokemukseen erikoistuneiden suunnittelijoiden merkitystä ketterissä ohjelmistoprojekteissa.

2.3. Yhteinen tietopohja

Tietämys (knowledge) on yksilön kyky ja taito ratkoa ongelmia, ja mikäli tätä tietämystä halutaan jakaa, tulee se ulkoistaa dataksi. Data on fyysisesti olemassa olevia asioita ympäröivässä maailmassa ja niillä on merkitys havainnoijalle tietyssä kontekstissa. Dataa voidaan säilöä ja kuljettaa, koska se on fyysisesti olemassa. Myös viestintä on tietämyksen jakamiseen liittyvä aktiviteetti [Stapel and Schneider 2012]. Yhteistyön sujumisen kannalta on kriittistä jakaa tietämys ja etenkin tietoisuus siitä, miten yhteistyökumppani tulkitsee ja tuottaa tietoa [Bjørn *et al.* 2014].

Yhteinen tietopohja (common ground) viittaa siihen tietämykseen, joka on kaikille osallistujille yhteistä ja jonka kaikki osallistujat tunnistavat yhteiseksi. Ihmisillä on tapana kuvailla samoja tapahtumia tai ideoita eri tavalla riippuen siitä, mitä oletamme keskustelukumppanin tietävän. Yhteistä tietopohjaa käytetään tietynlaisena kehyksenä, jonka kautta muotoilemme puheemme [Olson and Olson 2000]. Yhteisen tietopohjan muodostamisen on havaittu olevan oleellista ohjelmistoprojektin aiheen ja toimialueen tietämyksen lisäksi myös siinä, miten yhteistyön prosessit ja menetelmät organisoidaan [Bjørn *et al.* 2014].

Yhteistä tietopohjaa ei muodosteta pelkästään keskustelukumppanin taustaan liittyvistä yleisistä tiedoista vaan myös perustuen keskustelun aikana kerättyyn tietoon tämän ulkonäöstä ja käyttäytymisestä. Jos esimerkiksi sanomme jotain oletustemme perusteella, mutta huomaamme keskustelukumppanimme ilmeestä tai sanallisesta vastauksesta ettei tämä ymmärtänyt, voimme muuttaa oletusta jakamastamme yhteisestä tietopohjasta ja korjata väärinkäsityksen [Olson and Olson 2000].

Keskusteluun osallistujat muodostavat yleensä yhteisen tietopohjan dynaamisesti. He löytävät samankaltaisuuksia ja eroavaisuuksia välillään ja mukauttavat sanomisiaan näiden löytöjen perusteella. Muodostamme yhteisen tietopohjan niiden vihjeiden perusteella, mitä meillä on käytössä. Mitä vähemmän vihjeitä on, sitä vaikeampi yhteistä tietopohjaa on muodostaa ja sitä todennäköisemmin väärinkäsityksiä syntyy [Olson and Olson 2000]. Aiemmissa tutkimuksissa ollaan havaittu, että maantieteellisesti hajautetut ohjelmistoprojektit ovat haasteellisia juurikin rajoitettujen vuorovaikutuskeinojen vuoksi [Herbsleb *et al.* 2000].

Bjørn ja muut [2014] havaitsivat hajautettua yhteistyötä tutkiessaan, että toimintatapoihin ja menetelmiin liittyvä yhteinen tietopohja oli yhteistyön toimimisen kannalta tärkeää. Yhteisten toimintatapojen puute ja tietämättömyys yhteistyökumppaneiden työskentelytavoista aiheutti ilmeisiä ongelmia. Heidän tutkimissaan tapauksissa oli selvää, että yhteisen tietopohjan muodostaminen eri toimipisteiden työntekijöiden välillä oli vaikeaa. Kun tiimit toimivat samassa

toimipisteessä, yhteisen tietopohjan muodostaminen on helppoa. Samassa paikassa toimivat tiimit jakavat saman kulttuurisen ja paikallisen kontekstin, mutta myös tiedon siitä kuka tekee mitäkin; tällöin vuorovaikutus on helpompaa kuin hajautetussa tiimissä [Olson and Olson 2000].

On siis selvää, että yhteisellä tietopohjalla on tärkeä rooli aina, kun ohjelmistoprojekti on maantieteellisesti hajautettu. Yhteinen tietopohja voidaan myös jakaa kahteen erilaiseen tyyppiin. Ensimmäinen liittyy ohjelmistoprojektin kohteena olevaan toimialaan, kun taas toinen liittyy yhteistyökumppaneiden toimintatapoihin. Vaikeudet muodostaa kumman tahansa tyyppistä yhteistä tietopohjaa vaikuttavat koko projektin menestykseen [Bjørn *et al.* 2014].

Merkittävä ominaisuus samassa tilassa toimimisella, joka puuttuu etänä työskennellessä, on kyky havainnoida kanssatyöntekijöiden mielentilaa ja ylipäänsä sitä ovatko he paikalla. Tämä kyky on oleellinen osa yhteisen tietopohjan muodostamista. Kuitenkin ihmiset, joilla on jo valmiiksi laajasti muodostettua yhteistä tietopohjaa, pysyvät viestimään tehokkaasti myös vähemmän ilmaisuvoimaisilla viestintämedioilla. Mitä enemmän yhteistä tietopohjaa ihmiset pystyvät muodostamaan, sitä tehokkaammin he kykenevät viestimään keskenään ja sitä tuottavampia he ovat [Olson and Olson 2000].

Abdullah ja muut [2011] käsittelevät yhteiseen tietopohjaan läheisesti liittyvää käsitettä *jaettu käsitteellistäminen* (shared conceptualization), joka mahdollistaa vaatimusmäärittelyyn liittyvien tehtävien suorittamisen ja hallinnan. Jaetulla käsitteellistämällä tarkoitetaan sitä, miten kehitettävään tuotteeseen liittyvät käsitteet muodostetaan ja jaetaan vuorovaikutuksessa muiden kanssa. Jaettu käsitteellistäminen on siis prosessi, joka liittyy yhteisymmärryksen muodostamiseen kehitettävästä tuotteesta. Jaettu käsitteellistäminen muodostetaan tapaamisissa asiakkaan kanssa, kun tuotteeseen liittyviä vaatimuksia kerätään. Tämä johtaa luonnollisesti vaatimusten tarkentumiseen ja kehittymiseen, kun kehittäjät ovat mukana näissä tapaamisissa.

Inayat ja muut [2014] tarkastelivat kirjallisuuskatsauksessaan ketterien menetelmien vaatimusmäärittelyyn liittyviä käytäntöjä ja haasteita. Ketterät menetelmät vastaavat perinteisten menetelmien vaatimusmäärittelyjen ongelmiin, mutta synnyttävät myös uusia haasteita vaatimusmäärittelyyn. Tiivis yhteistyö asiakkaan kanssa ratkaisee ongelmia viestinnässä, jonka seurauksena laajoja dokumentteja vaatimuksista ei tarvita, asiakas validoi vaatimuksia jatkuvasti ja projektin laajuus ei missään vaiheessa ylitä sille määriteltyjä resursseja.

Inayat ja muut [2014] mainitsevat kuitenkin ketterien menetelmien synnyttävän niille ominaisia haasteita. Niukka dokumentaatio synnyttää ongelmia projekteissa, joissa

kehitystiimi on maantieteellisesti jakautunut tai jos projekti on monimutkainen. Ketteryys aiheuttaa vaikeuksia budjetoinnissa ja aikataulujen arvioinnissa, mutta myös sopimusneuvotteluissa. Tämä johtuu siitä, että ketterät menetelmät edellyttävät sitä, että vaatimusten muuttuminen kesken projektin on mahdollista, mikä luonnollisesti tekee kaiken etukäteen suunnittelun vaikeaksi.

2.4. Viestintä ketterissä ohjelmistoprojekteissa

Jatkuvasti muuttuvien tarpeiden maailmassa etukäteen suunnitteleminen on hankalaa, joten perinteisten ohjelmistokehitysmenetelmien rinnalle on kehittynyt tehokkaaseen viestintään perustuvia ketteriä menetelmiä [Chau and Maurer 2004].

Perinteisistä menetelmistä poiketen, ketterässä kehityksessä viestinnän tarvetta ei ole pyritty vähentämään vaan viestinnästä on haluttu tehdä nopeaa ja vaivatonta [Niinimäki *et al.* 2009]. Tätä edistetään edellyttämällä kaikkia tiimin jäseniä jakamaan sama avoin työskentelytila ja kannustamalla kasvokkaiseen vuorovaikutukseen. Tavoitteena on parantaa tiimin sisäistä viestintää, mutta myös viestintää asiakkaan suuntaan [Arora and Goel 2012].

Jokainen toimialue asettaa ohjelmistokehittäjille vaatimuksen ymmärtää toimialueen erityisiä ongelmia ja ammattikieltä. Tietyissä tilanteissa ammattikieli sisältää toimialueen peruskonsepteja. Esimerkiksi ohjelmiston kehittäminen kansainväliseen pankkitoimintaan edellyttää ammatillista ymmärrystä rahoitusalaista yleisesti. Tällaisessa tilanteessa kehitystiimi tarvitsee myös tietoa paikallisesta lainsäädännöstä ja asiakkaana olevan pankin toimintatavoista. [Bjørn *et al.* 2014]

Parhaiten ketterät menetelmät menestyvät organisaatioissa, joiden yrityskulttuuri tukee ripeää viestintää, ihmisten välistä luottamusta ja nopeaa palautteen saamista asiakkailta [Korkala *et al.* 2010]. Ketteriä tiimejä kuvaa intensiivinen yhteistyö ja itseohjautuvuus päämääränä toimivan ohjelmistotuotteen toimittaminen asiakkaalle sovittuun määräaikaan mennessä. Käytännössä ketterät tiimit ovat vastuussa itsestään ja koko projektin kohtalosta. Tiimin jäsenien tarvitsee viestiä ja tehdä yhteistyötä lyhyiden iteratiivisten jaksojen aikana niin, ettei pienintäkään mahdollisuutta epäonnistumiseen ole [Arora and Goel 2012].

Suurimmat haasteet maantieteellisesti hajautetuissa ketterän ohjelmistokehityksen projekteissa sijaitsevat viestinnässä, toiminnan koordinoinnissa ja hallinnassa. Viestinnän on raportoitu olevan yksi avaintekijä ohjelmistokehityksessä ja todisteet viittaavat sen olevan vahvasti kytköksissä koordinoinnin ja hallinnan tehokkuuteen

[Fernando *et al.* 2011]. Vähäinen viestintä, pienien vivahde-erojen katoaminen viestinnästä kasvokkaisen vuorovaikutuksen puuttuessa ja koordinoinnin vaikeus hidastavat työtahtia hajautetuissa ohjelmistoprojekteissa [Herbsleb *et al.* 2000]. Tehokkaampaan viestintään kannustavia ketterän ohjelmistokehityksen menetelmiä on pidetty ratkaisuna ongelmiin hajautetuissa ohjelmistoprojekteissa [Fernando *et al.* 2011].

Ketterä ohjelmistokehitys edellyttää onnistuakseen, että asiakas on tietoinen vastuistaan projektissa ja että edes joku on edustamassa asiakasta ketterään kehitykseen liittyvissä kokouksissa [Korkala *et al.* 2010]. Herbsleb ja muut [2000] havaitsivat suuria raportoituja ongelmia avun saannissa raskaiden työtaakkojen aikana etänä työskenteleviltä kollegoilta verrattuna samassa tilassa työskenteleviin. Mahdolliseksi selitykseksi ilmiölle he esittivät etänä saadun avun heikkoa laatua. Toisaalta paikallinen työyhteisö oli isompi kuin etänä työskentelevistä muodostunut, mikä saattoi osaltaan selittää avun saamista helpommin paikallisilta kollegoilta. Kolmantena vaihtoehtoisena selityksenä tutkimuksessa esitettiin, että koska käytetyillä viestintätavoilla oli vaikea ilmaista kiireellisyyttä, etänä oleva kollega aliarvioi pyydetyn avun tarpeen.

Monissa ryhmätyötilanteissa tietoisuus muista tarjoaa tietoa, joka on elintärkeää sujuvan ja tehokkaan yhteistyön kannalta. Ryhmätietoisuudella tarkoitetaan ymmärrystä siitä, ketkä ovat yhteistyökumppaneita, mitä he tekevät ja kuinka jokaisen oma toiminta vaikuttaa muiden työhön. Ryhmätietoisuuden tuomia etuja ovat viestinnän yksinkertaistuminen ja toiminnan koordinoinnin parantuminen. [Gutwin *et al.* 2004]

Viestintäteknologiaa hyödyntäviä yhteistyötilanteita on kahden tyyppisiä. Ensimmäisessä yhteistyötä tukevia ohjelmia käyttävät samassa tilassa työskentelevät tiimit pyrkimyksensä vähentää kokousten määrää. Toisessa yhteistyötilanteessa tiimin jäsenet eivät toimi samassa tilassa vaan tekevät yhteistyötä etäältä. [Graveline *et al.* 2000]

Alzoubi ja muut [2014] perehtyivät maantieteellisesti hajautetun ketterän ohjelmistokehityksen viestintähaasteisiin kirjallisuuskatsauksessaan. He havaitsivat, että useimmin tutkimuksissa mainittiin maantieteellisen etäisyyden aiheuttamat ongelmat viestinnässä.

Jopa samassa tilassa työskentelevien työntekijöiden välinen viestintä vähenee radikaalisti työpisteiden välisen etäisyyden ollessa yli 30 metriä. Tuota suuremmilla etäisyyksillä keskinäisen viestinnän on havaittu vähenevän samalle tasolle samassakin rakennuksessa työskentelevien kanssa kuin useiden kilometrien päässä työskentelevien kollegojen kanssa [Herbsleb *et al.* 2000]. Sopivan viestintävälineen valinta on kuitenkin

haastavaa ja väärän valinnan on raportoitu aiheuttavan monia ongelmia. [Fernando *et al.* 2011]

Työn jakaminen monen eri sijainnin välillä vaikuttaa kehityksen elinkaareen. Pidentynyttä kehitysaikaa saattaa selittää sijaintien välille syntynyt keskinäinen riippuvuus ja käytetyt viestintämallit [Herbsleb *et al.* 2000]. Ketterien tiimien jäsenien on havaittu käyttävän viestintään verkkosivuja, pikaviestintää, wikejä, puhelinta, telekonferensseja, videokonferensseja ja työpöydän jakoa [Fernando *et al.* 2011].

Viestinnän tehokkuuden on osoitettu riippuvan valitun viestintävälineen sopivuudesta. Monien käytännön ongelmien syyt ovat johdettavissa viestinnän ongelmiin ja vajaaseen tietoisuuteen muista tiimin jäsenistä. Näitä ongelmia voivat olla tehoton tiedonkulku, viivästyksiset tehtävien valmistumisessa ja samojen tehtävien tekeminen useaan kertaan. [Sindhgatta *et al.* 2011]

Hummel ja muut [2013] tekivät kirjallisuuskatsauksen viestinnästä ketterässä ohjelmistokehityksessä. He yllättyivät huomattavasti, etteivät kyenneet löytämään tutkimusta, joka olisi osoittanut ketterien menetelmien parantavan viestintää vaikka viestintää pidetään yhtenä keskeisistä ketterän kehityksen piirteistä. Kuten useat muutkin tutkijat, Hummel ja muut havaitsivat viestinnän olevan helpompaa samassa tilassa toimiville tiimeille ja kasvokkaisen vuorovaikutuksen vähentävän väärinkäsitysten mahdollisuutta. He huomauttavat kuitenkin epävirallisen viestinnän sopivan parhaiten pienille tiimeille ja ylläpitoprojektien edellyttävän perinteisempää dokumentaatiota tietämyksen säilyttämiseen.

Käytännössä ketterien ja perinteisten menetelmien raja voi olla varsin häilyvä. Bakalova ja Daneva [2011] vertasivat tapaustutkimuksessaan kahta ohjelmistoyritystä, joista toinen käytti projekteissaan perinteisiä menetelmiä ja toinen ketteriä menetelmiä. Perinteisiä menetelmiä käyttävä yritys oli kuitenkin omaksunut ketteristä menetelmistä tuttuja viestintätapoja ottaen samalla asiakkaan tiiviimmin mukaan kehitysprosessiin. Bakalova ja Daneva raportoivat projektien onnistumisen ja asiakastyytyväisyyden olleen samalla tasolla molemmissa yrityksissä. He päättelivät näytön perusteella asiakkaan aktiivisen osallistumisen olevan olennainen tekijä korkean asiakastyytyväisyyden saavuttamisessa.

2.5. Viestintävälineen valinta

Tehtävään tai tilanteeseen sopimattoman viestintävälineen valinta johtaa yleensä ongelmiin viestinnässä [Fernando *et al.* 2011]. Viestinnän onnistuminen edellyttää sitä,

että vastaanottaja ymmärtää viestin siten kuin lähettäjä sen on tarkoittanut ja että sekä vastaanottaja että lähettäjä ovat yhtä mieltä siitä, että vastaanottaja on todella ymmärtänyt viestin [Dennis and Valacich 1999]. Yksi tapa arvioida viestintävälineen eli median sopivuutta on tarkastella sitä *median ilmaisuvoimaisuuden teorian* (media richness theory, MRT) kautta. Sitten MRT:n saaman kritiikin pohjalta on syntynyt *median synkronisaation teoria* (media synchronicity theory, MST).

2.5.1 Median ilmaisuvoimaisuuden teoria

Rikkaalla medialla tarkoitetaan niitä viestintävälineitä, jotka mahdollistavat monipuolisen kielenkäytön, laajan skaalan erilaisia sanattomia vihjeitä, hyvät mahdollisuudet viestin personointiin ja nopean palautteen [Dennis and Valacich 1999]. Toisin sanoen näissä viestintävälineissä käyttäjän on mahdollista ilmaista viestinsä hyvin vapaalla tavalla, vastaanottajilla on viestin sisällön lisäksi muitakin vihjeitä viestin ymmärtämiseen, kuten äänensävy, ja mahdollisuus nopeaan vuorovaikutukseen.

Graveline ja muut [2000] toteavat, että median ilmaisuvoimaisuuden teoria tarkastelee viestintää moniselitteisyyden ja varmuuden näkökulmasta. Sen mukaan ilmaisuvoimaltaan rikas media on paras vaihtoehto moniselitteisiin viestintätilanteisiin.

Median ilmaisuvoimaisuuden teoria väittää, että tietyt mediat pystyvät paremmin välittämään informaatiota epävarmuuden tilanteissa. Epävarmuutta ilmenee, kun kehikko viestin ymmärtämiseen on olemassa, mutta prosessissa on moniselitteisyyttä. Moniselitteisyyttä ilmenee, kun informaatiolle tai sen tulkintakehikolle on useita (ja mahdollisesti ristiriitaisia) tulkintoja. Monitulkintaisuus vaatii neuvottelua viestijöiden välillä yhteisymmärryksen saavuttamiseksi viestin tulkinnasta. [Dennis and Valacich 1999]

Ilmaisuvoimaisuudeltaan heikompiakin viestintämedia riittää tehokkaaseen viestintään viestittävän asian ollessa yksiselitteinen ja kun epävarmuustekijöitä ei juurikaan ole. Tiedonpuute sen sijaan lisää epävarmuutta, ja tällöin viestittävä asia muuttuu selkeästi määritellystä tilanteesta hyvin määritellyksi ongelmaksi: tällaisessa tilanteessa tarvitaan lisää tietoa. Tällöin tarve rikkaammalle viestintämedialle kasvaa. Jos taas viestittävä asia on monitulkintainen, mutta tiedonpuute on vähäistä, ongelmia on niukasti. Ongelmat ovat kuitenkin monitulkintaisia ja epäselviä, joten ilmaisuvoimaltaan rikkaalle viestintämedialle on tarvetta. Kun viestittävä asia on moniselitteinen ja tiedonpuutteen luoma epävarmuus on suuri, myös tarve ilmaisuvoimaltaan rikkaalle viestintämedialle on suuri. [Takeda 2007]

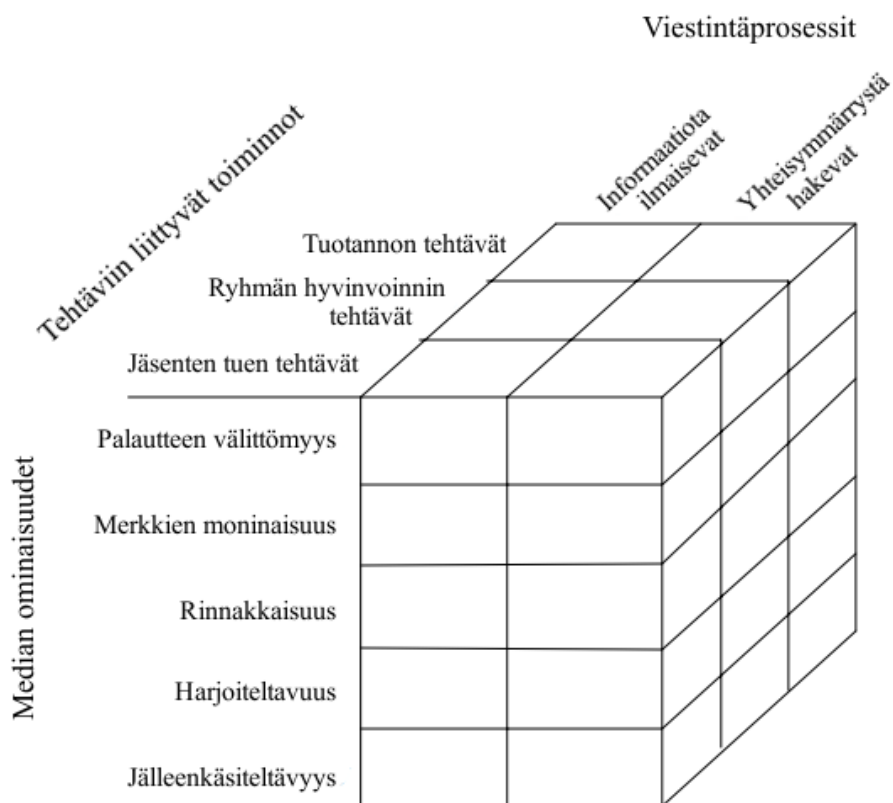
Graveline ja muut [2000] havaitsivat tutkimuksessaan, että ihmissuhteisiin liittyviä aiheita käsiteltiin enemmän pikaviestimissä kuin sähköpostissa. Tämä havainto tukee median ilmaisuvoimaisuuden teorian väitettä siitä, että ajallisesti synkronoidut, monikanavaiset viestintävälineet tukevat ihmissuhteisiin liittyvää ja sosioemotionaalista viestintää paremmin kuin asynkroniset viestintävälineet. Tutkimukseen osallistuneet tiimit käyttivät ajallisesti synkronoituja ja monikanavaisia viestintävälineitä monitulkintaisissa viestintätilanteissa, kun taas yksiselitteisissä viestintätilanteissa suosittiin asynkronisia yksikanavaisia viestintävälineitä kuten sähköpostia. Tämä kaikki tuki median ilmaisuvoimaisuuden teorian oletusta, että monitulkintaiset tilanteet edellyttävät ilmaisuvoimaisempia viestintävälineitä.

Rice ja muut [1998] toteavat kuitenkin, että käyttök konteksti ja todellinen käyttötapa vaikuttavat viestintämedian valinnassa yksilötasolla median ilmaisuvoimaisuutta ja viestintätilanteen monitulkintaisuutta enemmän. Tämä on hieman paradoksaalista, sillä median ilmaisuvoimaisuuden teoria kehitettiin nimenomaan huomioimaan kontekstuaaliset tekijät. Rice ja muut myöntävät teorian tunnustaneen oikein tiettyjä viestintätilanteiden ongelmien piirteitä, mutta aliarvioineen viestintäkontekstin luonteen.

2.5.2 Median synkronisaation teoria

Median ilmaisuvoimaisuuden teoria selittää näennäisesti viestintään vaikuttavia tekijöitä hyvin, mutta käytännön tutkimuksissa teoria on saanut ristiriitaisia tuloksia [Dennis and Valacich 1999]. Median synkronisaation teoriolla on pyritty vastaamaan tähän ongelmaan ja tavoittamaan viestintään vaikuttavia tekijöitä paremmin kuin median ilmaisuvoimaisuuden teoria.

Median synkronisaation teoria tarkastelee, miten eri viestintämedioiden käyttö vaikuttaa onnistuneisiin tehtäväsuorituksiin viestintäprosesseissa. Median synkronisaatiolla tarkoitetaan sitä, missä määrin yksilöt työskentelevät yhdessä saman tehtävän parissa samaan aikaan. MST tarkastelee viestintää kolmen ulottuvuuden kautta, jotka on esitetty kuvassa 2. [Fernando *et al.* 2011]



Kuva 2: Median synkronisaation teorian ulottuvuudet [Dennis and Valacich 1999]

Ensimmäinen ulottuvuus tarkastelee kahta keskeistä viestintäprosessia, jotka esiintyvät yhteistyössä. Nämä kaksi prosessia ovat yhteisymmärrystä hakevat (*converging*) ja informaatiota ilmaisevat (*conveying*) viestintäprosessit. Informaatiota ilmaisevat viestintäprosessit välittävät tietoa. Matala synkronisaatiotaso antaa viestin vastaanottajalle enemmän aikaa syventyä viestiin ja sen sisältämään tietoon. Näin ollen matalan synkronisaatiotason viestintämediat sopivat informaatiota ilmaisevaan viestintään. Yhteisymmärrystä hakevissa viestintäprosesseissa oleellinen informaatio on jo tiedossa, mutta viestinnällä pyritään löytämään yhteinen käsitys informaation tulkinnasta. Tällainen viestintä edellyttää korkeaa synkronisaatiotasoa. [Fernando *et al.* 2011]

Yleensä viestintämediat, jotka tukevat hyvin informaatiota ilmaisevia viestintäprosesseja, menestyvät heikommin yhteisymmärrystä hakevissa viestintäprosesseissa ja päinvastoin. Tiimien työtehtävät sisältävät tyypillisesti kummankin tyyppisiä viestintätarpeita, joten MST suosittelee vaihtelevaan erilaisten viestintämedioitten välillä viestintätarpeiden mukaan. [Fernando *et al.* 2011]

Median synkronisaation teorian toinen ulottuvuus määrittelee viisi median ominaisuutta, joiden avulla median synkronisuutta voidaan arvioida. Taulukossa 3 on lyhyesti kuvattu nämä ominaisuudet.

Median viisi ominaisuutta median synkronisaation teorian mukaan

<i>Palautteen välittömyys</i>	Kuinka pian palaute saadaan viestin vastaanottajalta
<i>Merkkien moninaisuus</i>	Kuinka monella eri tavalla informaatio voidaan esittää ja välittää
<i>Rinnakkaisuus</i>	Kuinka monta samanaikaista tehokasta keskustelua voidaan muodostaa
<i>Harjoiteltavuus</i>	Mahdollisuus muotoilla ja hioa viesti ennen lähettämistä
<i>Jälleenkäsiteltävyys</i>	Mahdollisuus noutaa ja käsitellä aiemmin vastaanotettu viesti paremman ymmärryksen saavuttamiseksi

Taulukko 3: Median viisi ominaisuutta [Fernando et al. 2011]

Ensimmäinen viidestä ominaisuudesta on *palautteen välittömyys*. Tämä mittaa sitä, kuinka hyvin viestintämedia tukee nopean palautteen antamista viestin lähettäjälle. Viestintämedialla voi olla mahdollisuus ripeään kaksisuuntaiseen viestintään [Dennis and Valacich 1999].

Toisella ominaisuudella, *merkkien moninaisuudella*, tarkoitetaan tapojen määrää, joilla informaatio voidaan esittää ja välittää. Merkkien moninaisuus voi vaikuttaa ainakin neljällä tavalla viestintään ja viestin ymmärrettävyyteen. Ensimmäiseksi informaatio voi olla helpompi välittää tietyssä muodossa, esimerkiksi kuvana, kuin toisessa, kuten puheen avulla. Toiseksi verbaalinen ja non-verbaalinen viestintä yhdessä mahdollistavat sen, että viestin lähettäjä sisällyttää viestiin muutakin informaatiota kuin viestin sanat. Kolmanneksi, mikäli viestin tuottaminen tai vastaanottaminen tiettyä tapaa käyttäen on hitaampaa tai kalliimpaa, käyttäjät saattavat mukauttaa tapaansa viestiä. Viimeisenä on huomioitava, että verbaalisten ja non-verbaalisten merkkien puuttuessa katoaa myös vaikutelma sosiaalisesta läsnäolosta. Viestinnästä häviää tuolloin inhimillisyys ja sosiaalisuus. [Dennis and Valacich 1999]

Rinnakkaisuus viittaa samanaikaisten keskustelujen määrään, joita valitulla viestintämedialla on mahdollista tehokkaasti pitää yllä. Perinteiset mediat, kuten puhelin, sallivat vain yhden keskustelun kerrallaan, kun taas monet sähköiset mediat on mahdollista järjestellä niin, että useampaa keskustelua on mahdollista ylläpitää samaan aikaan. Samanaikaisten keskustelujen määrän kasvaessa on kuitenkin yhä vaikeampaa seurata ja koordinoita niitä. [Dennis and Valacich 1999]

Harjoiteltavuudella tarkoitetaan niitä viestintämedian ominaisuuksia, jotka sallivat lähettäjän harjoitella tai hienosäätää viestiä ennen sen lähettämistä vastaanottajalle. *Jälleenkäsiteltävyydellä* taas viitataan mahdollisuuteen tarkastella ja muokata lisää jo lähetettyä viestiä viestintätilanteessa. [Dennis and Valacich 1999]

Median synkronisaation teorian kolmas ulottuvuus tarkastelee tehtäviin liittyviä toimintoja, joita ovat tuotantoon, ryhmän hyvinvointiin ja jäsenten tukeen liittyvät toiminnot. Tämä ulottuvuus painottaa, että sosiaaliset tarpeet tulee huomioida viestinnän parantamiseksi [Fernando *et al.* 2011]. Kärjistetyksi tämä tarkoittaa esimerkiksi sitä, että mikäli ryhmän jäsenet eivät tule keskenään toimeen, myös viestintä väistämättä kärsii.

2.5.3 Eri medioiden ominaisuuksia

Taulukossa 4 on arvioitu muutamien eri viestintämedioiden ilmaisuvoimaisuutta. Monissa tapauksissa viestintämedian ilmaisuvoimaisuus vaihtelee, sillä viestintämedian säädöt voivat lisätä tai heikentää ilmaisuvoimaisuutta [Dennis and Valacich 1999]. Esimerkiksi html-muotoiltu sähköposti voi sisältää kaavioita ja kuvia tekstin rinnalla, kun taas komentoriviltä toimivalla sähköpostiohjelmalla voi tarkastella vain viestin tekstisisältöä. Asynkronisella ryhmäohjelmalla tarkoitetaan taulukossa 4 yhteistyöhön suunniteltua ohjelmistoa, joka ei edellytä käyttäjiltä jatkuvaa läsnäoloa vaan he saavat muutokset näkyville käyttäessään ohjelmistoa seuraavan kerran. Tällainen ryhmäohjelma on esimerkiksi wiki-alusta, jolla käyttäjät voivat luoda, muokata ja tarkastella sisältöä sekä keskustella siitä.

	Palautteen välittömyys	Merkkien moninaisuus	Rinnakkaisuus	Harjoiteltavuus	Jälleenkäsiteltävyys
<i>Kasvokkain</i>	korkea	matala – korkea	matala	matala	matala
<i>Videokonferenssi</i>	keskitaso – korkea	matala – korkea	matala	matala	matala
<i>Puhelin</i>	keskitaso	matala	matala	matala	matala
<i>Kirje</i>	matala	matala – keskitaso	korkea	korkea	korkea
<i>Ääniviesti</i>	matala	matala	matala	matala – keskitaso	korkea
<i>Sähköposti</i>	matala – keskitaso	matala – keskitaso	keskitaso	korkea	korkea
<i>Pikaviestin</i>	keskitaso	matala – keskitaso	keskitaso	matala – keskitaso	matala – keskitaso
<i>Asynkroninen ryhmäohjelma</i>	matala	matala – korkea	korkea	korkea	korkea

Taulukko 4: Eri viestintämedioiden ilmaisuvoimaisuus [Dennis and Valacich 1999]

Taulukosta 4 on nähtävissä myös, ettei yksikään viestintämedia ole vahva kaikkien ominaisuuksien suhteen. Eri viestintätarpeet edellyttävät eri ominaisuuksia, joten ilmaisuvoimaisin media riippuu siitä, millaista asiaa ollaan viestimässä [Dennis and Valacich, 1999].

2.6. Scrum, Extreme Programming ja Kanban

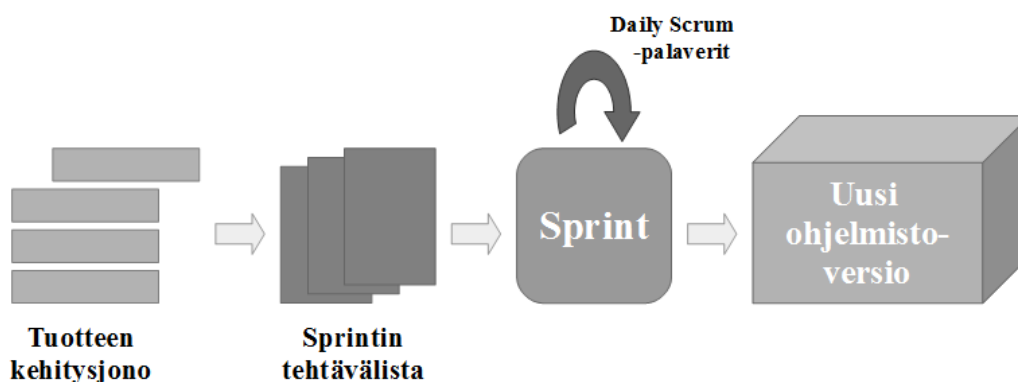
Yleisimmin käytössä olevat ketterät menetelmät ovat Scrum, Extreme Programming (XP) sekä Kanban, joista Scrum on tällä hetkellä selvästi käytetyin [Matharu *et al.* 2015]. Näitä menetelmiä yhdistää viestinnän ja yhteistyön korostaminen ja sopeutuminen vaatimusten muutoksiin ohjelmistokehityksen aikana. Lähemmin

tarkasteltuna näillä kolmella menetelmällä on kuitenkin toisistaan poikkeavat lähestymistavat ketterään kehitykseen.

2.6.1 Scrum

Scrum-menetelmässä ohjelmistoa kehitetään lyhyissä kehitysjaksoissa, joita kutsutaan nimellä sprint. Sprint on enimmillään muutaman viikon mittainen ja sen aikana toteutetaan sprintin alussa päätetyt ominaisuudet. Halutut ominaisuudet esitetään käyttäjätarinoiden muodossa. Käyttäjätarinoiden tarkoitus ei ole kuvata ominaisuuksia täydellisesti vaan toimia muistutuksena siitä, mikä ominaisuuden tarkoitus on. Käyttäjätarinoiden tulee olla laajuudeltaan yhden sprintin aikana toteutettavissa olevia. [Beyer 2010]

Käyttäjätarinat sijoitetaan aluksi *tuotteen kehitysjonoon* (product backlog), josta jokaisen sprintin alussa valitaan tärkeimmät *sprintin tehtävälisalle* (sprint backlog). Kehitystiimi päättää yhdessä jokaisen sprintin alussa, mitä tehtävälisalle otetaan, ja tehtävälisalle valitaan vain sen verran tehtäviä, että ne on mahdollista toteuttaa sprintin aikana. Sprintin lopussa tiimi arvioi, miten sprintti sujui, mitä ongelmia tiimi kohtasi ja mitä asioita seuraavaa sprinttiä varten olisi tarpeen muuttaa. [Beyer 2010]



Kuva 3: Scrum kehitysprosessi [Matharu et al. 2015]

Kuvassa 3 on kuvattu Scrum-projektin sprintin vaiheet. Tuotteen kehitysjonosta siirretään jokaisen sprintin alussa sopiva määrä tehtäviä sprintin tehtävälisalle. Tämän jälkeen varsinainen kehitystyön sisältävä sprint alkaa. Jokainen päivä alkaa Daily Scrum -tapaamisella, jossa jokainen kehitystiimin jäsen kertoo, mitä on tehnyt, mitä aikoo seuraavaksi tehdä ja onko ilmennyt kehitystyötä haittaavia ongelmia. Tarkoituksena on pitää kaikki kehitystiimin jäsenet tietoisina sprintin tehtävien edistymisestä ja pitää huoli, etteivät mahdolliset ongelmat pysäytä kehitystyötä tarpeettoman pitkiksi ajoiksi.

Jokaisen sprintin lopussa valmistuu uusi versio kehitettävästä ohjelmistosta. [Matharu *et al.* 2015]

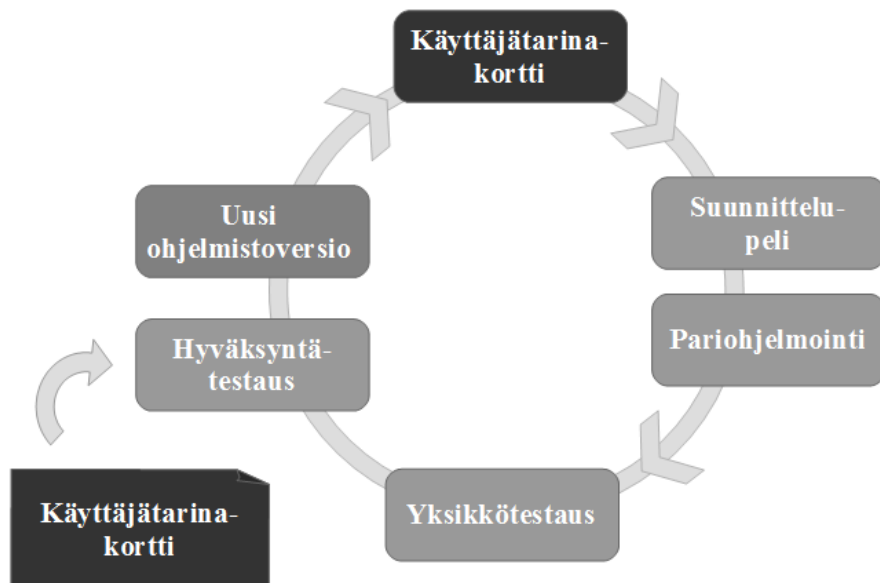
Scrum-menetelmässä *tuoteomistaja* (product owner) on yksi tärkeimmistä *toimijoista* (stakeholder). Tuoteomistaja toimii asiakkaiden edustajana ja pyrkii varmistamaan, että asiakkaiden tarpeet otetaan huomioon ohjelmiston toiminnallisuuksia kehitettäessä [Holmström Olsson and Bosch 2015]. Tuoteomistaja ohjaa tuotteen kehitysprosessia ja määrittelee tärkeysjärjestyksen ohjelmistotuotteen toiminnallisuuksille [Matharu *et al.* 2015]. Tuoteomistaja on tyypillisesti pääkäyttäjä kehitettävälle tuotteelle tai joku, jolla on hyvä ymmärrys asiakkaista, markkinoista, kilpailijoista ja kehitettävän ohjelmistotuotteen toimialueen tulevista trendeistä [Holmström Olsson and Bosch 2015]. Koska Scrum ei salli muutoksia vaatimuksiin kesken sprintiksi kutsuttua kehitysjaksoa, ei tuoteomistajan läsnäolo ole välttämätöntä koko ajan.

Scrum-projektin jokainen kehitysjakso sisältää kaikki ohjelmistokehityksen vaiheet [Matharu *et al.* 2015]. Niinpä tuoteomistajan on hyvä olla läsnä jokaisen kehitysjakson alussa, jossa valitaan kehitettävät ominaisuudet kyseiseen sprinttiin. Läsnäolosta on hyötyä myös kehitysjakson lopussa, jossa arvioidaan sprintin onnistumista ja mahdollisia parannuskohteita seuraavaan iteraatioon.

2.6.2 Extreme Programming

Extreme Programming taas edellyttää Scrum-menetelmää intensiivisempää osallistumista asiakkaalta, sillä menetelmä perustuu jatkuvaan asiakaspalautteeseen [Matharu *et al.* 2015]. Siinä myös oletetaan, että asiakas tai asiakkaan edustaja, joka on todellinen ja kokenut loppukäyttäjä, kyetään saamaan tiimin täysivaltaiseksi jäseneksi [Beyer 2010].

Vaatimusmäärittelyt toteutetaan käyttäjätarinoina, joista kehittäjät pilkkovat pienempiä tehtäviä ja asiakas puolestaan asettaa nämä tehtävät kehityksen kannalta tärkeysjärjestykseen. Ohjelmistotuotetta kehitetään koko ajan pieninä paloina eteenpäin Scrum-menetelmän tapaan lyhyissä kehitysjaksoissa. Kuvassa 4 on kuvattu Extreme Programming kehitysprosessi, joka toistuu jokaisen kehitettävän ominaisuuden kohdalla koko ohjelmistoprojektin ajan. [Matharu *et al.* 2015]



Kuva 4: Extreme Programming kehitysprosessi [Matharu et al. 2015]

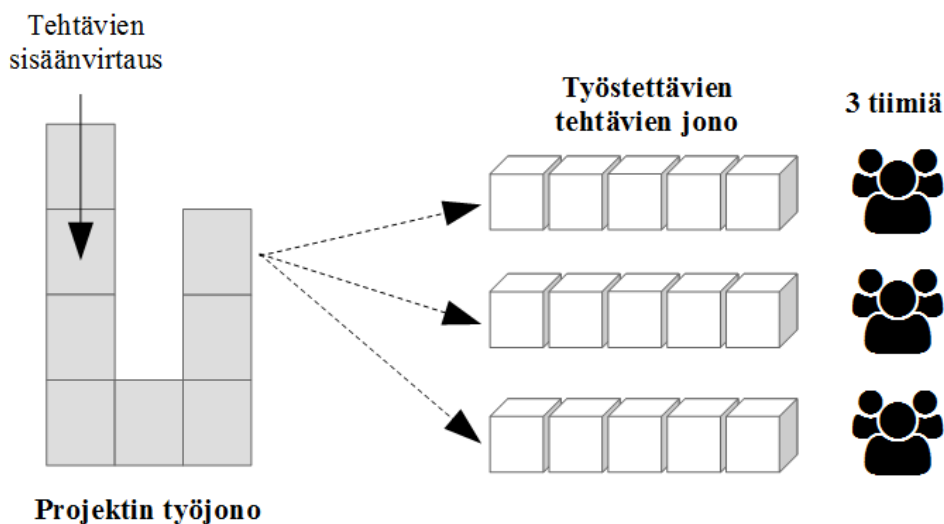
Projektin alussa vaatimukset kerätään käyttäjätarinoiden muotoon. Tämän jälkeen asiakkaan kanssa neuvotellaan niin kutsutussa suunnittelupelissä, mitkä käyttäjätarinat ovat tärkeimpiä ja paljonko kunkin toteuttaminen kuluttaa resursseja. Kehittäjät antavat asiakkaalle ”kehitysbudjetin” perustuen siihen, miten paljon edellisessä iteraatiossa tai versiossa saatiin toteutettua. Kehitettäviksi valittavien tarinoiden kustannukset eivät saa ylittää tätä annettua budjettia. Extreme Programming-menetelmässä käytetään yleisesti parikoodausta ja testilähtöistä ohjelmointia. Kun valittu käyttäjätarina on toteutettu ja sen yksikkötestit menevät läpi, käytetään kyseistä käyttäjätarinaa hyväksymistestauksen pohjana. Tämän jälkeen kehitetty ohjelmistoversio julkaistaan ja uusi kehitysjakso aloitetaan. [Martin 2003, s. 15]

2.6.3 Kanban

Kanban-menetelmässä kehitystyö organisoidaan Kanban-taulun avulla, jossa tehtävät on jaettu jonossa oleviin, kehityksessä oleviin ja valmiisiin tehtäviin. Tarkoituksena on pitää kehityksessä olevien töiden määrä minimissään ja karsia ylimääräistä työtä toteuttamalla ominaisuuksia vasta sillä hetkellä, kun niille on todellinen tarve. Asiakkaan tehtävä on tuottaa vaatimuksia Kanban-tauluun, joten tarvetta jatkuvalla läsnäololle ei ole. [Matharu et al. 2015]

Kanban-menetelmässä tehtävät etenevät tilasta toiseen. Tehtävät kuvataan kortteina Kanban-taululla, jota käytetään visuaalisena hallintajärjestelmänä. Kuvassa 5 on kuvattu Kanban-menetelmän eteneminen kolmen tiimin esimerkkinä. Prosessi alkaa

asiakkaan tekemästä tehtäväpyynnöstä, joka siirtyy kehitysjonoon. Kehittäjätiimit poimivat jonosta tehtäviä työstettäväksi ennalta määritellyn ylärajan mukaan, joka esimerkiksi on asetettu viiteen tehtävään. Valmistuneiden tehtävien kortit kuitataan seuraavaan tilaan siirtämällä ne seuraavan tilan työlistalle, josta ne taas poimitaan seuraavan vaiheen työstettäväksi. [Fitzgerald *et al.* 2014]



Kuva 5: Kanban kehitysprosessi [Fitzgerald *et al.* 2014]

Kanban on saanut alkunsa Toyotan tuotantojärjestelmästä, joten tehtävän kulun tilasta toiseen voi ajatella etenevän samaan tapaan kuin auton osa etenee tuotannosta varastoon ja sieltä asiakkaalle. Työstettävänä olevien tehtävien määrää rajoitetaan, joten kyse on jatkuvasta integraatiosta, joka myös mahdollistaa säännöllisen palautteen asiakkaalta. [Fitzgerald *et al.* 2014]

Kanban-menetelmän idean voi tiivistää sen kolmeen osaan: työnkulun visualisoimiseen taululla siirtyvinä kortteina, samanaikaisen työn rajoittamiseen tiimien tehtäväjonoilla ja yksittäisten tehtävien läpimenoajan mittaamiseen työjonoon saapumisesta tehtävän valmistumiseen. Läpimenoaika vaihtelee tietysti ohjelmistokehityksessä tehtävien erilaisuuden vuoksi, mutta sen mittaaminen tarjoaa kuitenkin kehitystiimeille ymmärrystä kehitysprosessin tehokkuudesta ja mahdollisista kehityskohteista. Mikäli tietynlaisiin tehtäviin kuluu huomattavasti enemmän aikaa kuin toisiin, voidaan tarkistaa, olisiko näiden työtehtävien kohdalla mahdollista helpottaa ja tehostaa kehitysprosessia. [Hämäläinen 2016]

2.6.4 Yhteenveto menetelmistä

Taulukkoon 5 on koostettu kolmen esitellyn ketterän menetelmän viestintään liittyviä piirteitä. Sekä Scrum-menetelmässä että XP:ssä tarpeet ja sitä kautta määrittävät

järjestelmän vaatimukset kirjataan käyttäjätarinoina, joista yhteistyössä pilkotaan kehitystehtäviä työjonoihin. Kanban-menetelmässä asiakkaan tehtävä on tuottaa vaatimuksia kortteihin, jotka etenevät Kanban-taululla perinteisen vesiputousmallin tyyliä, mutta ketterille menetelmille ominaiseen tapaan pieninä paloina.

Asiakas ei välttämättä ole kehitettävän ohjelmistotuotteen loppukäyttäjä vaan ensisijaisesti tuotteen tilaaja. Ketterissä menetelmissä asiakkaan tehtävä on toimia tiiviissä yhteistyössä kehitystiimin kanssa parhaan lopputuloksen saavuttamiseksi. XP ja Kanban olettavat asiakkaan itse osallistuvan projektiin, mutta Scrum-menetelmässä asiakkaan etua ajaa nimetty tuoteomistaja, jonka ei välttämättä tarvitse olla asiakkaan oma työntekijä.

XP edellyttää asiakkaalta aktiivista osallistumista yhtenä kehitystiimin jäsenenä. Scrum ja Kanban eivät edellytä asiakkaan jatkuvaa läsnäoloa. Scrum-menetelmässä asiakkaan tai asiakkaan edustajan läsnäoloa tarvitaan sprinttien alussa ja lopussa, jolloin suunnitellaan seuraavan kehitysjakson tehtäviä ja arvioidaan tehdyn työn laatua. Kanban-menetelmässä ei ole määritelty vaatimusta asiakkaan läsnäololle.

	Scrum	Extreme Programming	Kanban
<i>Vaatimusmäärittely</i>	Käyttäjätarinat tuotteen kehitysjonossa	Käyttäjätarinat	Kortteina Kanban- taulussa
<i>Asiakkaan edustaja</i>	Nimetty tuoteomistaja	Asiakas itse	Asiakas itse
<i>Asiakkaan läsnäolo</i>	Vähintään sprinttien alussa ja lopussa läsnä	Samassa tilassa kehittäjien kanssa	Tarvittaessa, ei välttämätöntä
<i>Asiakkaan rooli</i>	Tuoteomistaja tai yhteyshenkilö tuoteomistajalle. Tuoteomistaja varmistaa projektin etenevän asiakkaan edun mukaisesti	Ominaisuuksien priorisointi, toimialaosaja projektissa ja tehdyn työn arviointi	Ominaisuuksien priorisointi ja syöttäminen kehitysjonoon sekä tehdyn työn arviointi
<i>Päivitystaajuus</i>	Uusi versio jokaisen sprintin lopussa	Jatkuva integraatio	Jatkuva integraatio

Taulukko 5: Ketterän ohjelmistokehityksen menetelmien viestintätapojen vertailua

Asiakkaan tai asiakkaan edustajan rooli kaikissa kehitysmenetelmissä on varmistaa, että kehitystyössä edetään tärkeimmät ominaisuudet edellä ja että valmistuneet työt vastaavat asiakkaan tarvetta. XP:ssä asiakkaan roolina on toimia myös

toimialaosaajana kehitystiimissä vaikkakin myös Scrum- ja Kanban-menetelmässä asiakas toimii tietolähteenä toimialaan liittyvissä kysymyksissä.

Uusia versioita asiakkaan testattavaksi ja nähtäväksi syntyy Scrum-menetelmässä jokaisen kehitysjakson lopussa. Sen sijaan XP ja Kanban ovat jatkuvan integraation menetelmiä ja niissä kehitettävästä ohjelmistosta syntyy uusia versioita jokaisen valmistuneen ominaisuuden yhteydessä.

3. Tutkimusasetelma

Kuten johdantoluvussa mainitsin, tämän tutkielman tarkoitus on kartoittaa, *miten ketterien menetelmien projekteissa ylläpidetään yhteistä tietopohjaa asiakkaan ja kehitystiimin välillä*. Tässä luvussa kuvaan tähän tutkielmaan valitsemani lähestymistavan ja kerron, miten tämän tutkielman aineisto kerättiin. Kohdassa 3.3. esittelen tähän tutkielmaan osallistuneet haastateltavat sekä taustatietoja yrityksistä, joissa he työskentelevät.

3.1. Usean tapauksen tapaustutkimus

Luvussa 2 yhteinen tietopohja osoittautui viestintää ohjaavaksi kehykseksi, jolla pyritään tekemään vuorovaikutuksesta ja yhteistyöstä mahdollisimman sujuvaa. Aiemmat tutkimukset osoittivat myös, että yhteisen tietopohjan muodostamisessa hyödynnetään varsinaisen sisällön lisäksi sanatonta viestintää kuten eleitä ja äänenpainoja [Olson and Olson 2000]. Tästä syystä lähdin kartoittamaan ketterissä projekteissa käytettyjä viestintävälineitä ja edellisessä luvussa esittelemäni taustateorian valossa niiden soveltuvuutta erilaisissa viestintätilanteissa asiakkaan ja kehitystiimin välillä.

En voinut etukäteen tietää, mitä erilaisia viestintämenetelmiä ja -välineitä eri yrityksissä ja projekteissa käytetään, joten avoimien kysymysten käyttäminen oli välttämätöntä. Tältä pohjalta sopivimmaksi aineistonkeruumenetelmäksi valikoitui puolistrukturoitu haastattelu, joista kerron lisää kohdassa 3.2.

Alkuperäinen ajatukseni oli tehdä yksi haastattelu useammassa ohjelmistoalan yrityksessä yhtä yksittäistapausta kattavamman kuvan saamiseksi. Haastatteluissa ilmeni kuitenkin nopeasti, ettei yrityksissä ollutkaan vakiintuneita viestintäkäytäntöjä projekteille. Tarkemman kuvan yksittäisen yrityksen toimintakulttuurista saisi haastatteleamalla muutamaa eri tason työntekijää samasta yrityksestä. Haastatellut esittelivät haastatteluissa valitsemaansa projektia ja viestintää kyseisessä projektissa.

Jokainen haastattelu edusti omaa tapaustaan, joten tämän tutkimuksen menetelmäksi valikoitui usean tapauksen tapaustutkimus.

Usean tapauksen tapaustutkimus soveltuu tilanteisiin, joissa halutaan joko vahvistaa aiemmassa tapaustutkimuksessa saatuja tuloksia tai tuottaa vastakkaisia tuloksia ennakoituista syistä [Järvinen ja Järvinen 1996, 56]. Median synkronisaation teorian pohjalta olisi odotettavissa, että tämän tutkimuksen haastateltavien esittelemien projektien viestintävälineiden valinta riippuu siitä, millaisista viestintätilanteista on kyse ja mihin tavoitteisiin viestinnällä pyritään [Fernando *et al.* 2011].

Tapaustutkimuksella voidaan pyrkiä kuvailemaan tutkittua tapausta tai tapauksia, sillä voidaan testata teoriaa tai sitä voidaan myös käyttää teorian luomiseen [Järvinen ja Järvinen 1996, 53]. Tässä tutkimuksessa keskityn kuvailemaan tutkimiani tapauksia ja vertailemaan niitä keskenään. Tapaustutkimuksella pyritään vastaamaan kysymyksiin ”miten” ja ”miksi” ja se sopii tämän tutkimuksen kaltaiseen kuvailevaan tutkimukseen [Järvinen ja Järvinen 1996, 55]. Koska otos on hyvin suppea, tapaustutkimuksissa pyritään tilastollisen yleistämisen sijaan tekemään analyyttinen yleistys, joka siis laajentaa ja yleistää teoriaa [Järvinen ja Järvinen 1996, 55].

3.2. Aineistonkeruu

Tämän tutkielman aineistonkeruussa hyödynsin puolistrukturoituja haastatteluja. Puolistrukturoidussa haastattelussa on valmiita kysymyksiä ja avoimia keskusteluteemoja [Järvinen ja Järvinen 1996, 102]. Kysymykset ovat kaikille haastatelluille samoja, jolloin samat asiat tulevat käsitellyiksi kaikissa haastatteluissa, mutta haastatellut saavat vastata omin sanoin [Eskola ja Vastamäki 2010, 28].

Keskustelempämpi lähestymistapa haastatteluun tarjosi myös mahdollisuuden kysyä tarkentavia kysymyksiä sekä kysyä uusia kysymyksiä aiheista, jotka nousivat esille vasta haastattelutilanteessa. Järvinen ja Järvinen [1996, 102-103] toteavatkin tämän avoimuuden olevan yksi haastattelumenetelmän vahvuuksista tuoda esiin uusia näkökulmia. Lähestymistapani näihin haastatteluihin muistutti siis teemahaastattelua, jossa keskustelunomaisemmalla haastattelutilanteella pyritään saamaan vuorovaikutuksessa haastateltavan kanssa selville tutkimuksen aihepiiriin kuuluvia kiinnostavia asioita [Eskola ja Vastamäki 2010, 26].

Haastateltavat täyttivät ennakolta liitteestä A löytyvän esitietolomakkeen, jossa kerättiin haastattelun kannalta oleellista taustatietoa haastateltavista, kuten kokemuksesta ketteristä menetelmistä ja työrooleista, joissa he ovat toimineet eri

projekteissa. Esitietolomakkeella pyydettiin haastateltavia myös miettimään yhtä ketterää projektia haastattelua varten, jotta itse haastattelussa päästiin nopeasti käsittelemään projektissa käytettyä viestintää. Esimerkkiprojektin valitsemisella halusin fokusoida haastatteluja paremmin vertailtavaan muotoon. Tarkastelemalla vain yhtä projektia on helpompi arvioida viestintään vaikuttavia taustoja ja tilanteita.

Keskustelua ohjanneet kysymykset liittyivät yhteiseen tietopohjaan. Muodostin kysymykset taustateorian pohjalta ja pyrkimyksenäni oli kysyä oleellisia asioita esimerkkiprojektien taustoista, viestintävälineiden valintaan vaikuttaneista syistä ja projekteissa tapahtuneesta viestinnästä asiakkaiden ja kehitystiimien välillä. Varsinaisiin haastatteluihin ei tarvinnut tehdä pilotin perusteella muutoksia, joten myös pilottihaastattelusta saatu data on sisällytetty aineistoon. Haastatteluskripti keskustelua ohjanneiden kysymyksien kera löytyy liitteestä B.

Yksittäinen haastattelu kesti keskimäärin tunnin ja ne nauhoitettiin. Nauhoitettujen haastattelujen sisältö ryhmiteltiin yhtenevien teemojen suhteen ja näitä yhteisiä teemoja tarkasteltiin aiempien tutkimusten löytöjen valossa.

3.3. Tutkimukseen osallistuneet yritykset ja haastateltavat

Tutkielman aineisto on kerätty yksilöhaastatteluilla kolmesta ohjelmistoalan yrityksestä, jotka ovat hyödyntäneet projekteissaan ketteriä ohjelmistokehityksen menetelmiä. Yritysten perustiedot kuvataan taulukossa 6. Haastateltavia rekrytoitiin henkilökohtaisesti osoitetuilla sähköpostiviesteillä Tampereen yliopistolla opiskelleiden ihmisten joukosta, jotka nykyään toimivat tamperelaisissa ohjelmistoyrityksissä. Kutsun yhteydessä pyydettiin myös välittämään kutsu muille mahdollisille haastateltaville. Jokaisesta yrityksestä pyrittiin tavoittamaan suunnittelun ja vaatimusmäärittelyn tehtävissä sekä kehitystehtävissä toimineita ihmisiä eri näkökulmien saamiseksi.

Yritys 1 on alle 20 työntekijän ohjelmistoyritys. Se on toiminut ohjelmistoalalla 9 vuotta. Yritys 1 tarjoaa sekä suunnitteluun että ohjelmistokehitykseen liittyviä palveluita asiakkailleen. Vakiintuneiden kehittäjä- ja suunnittelutiimien sekä hyvän asiakasyhteistyön avulla pyritään vastaamaan nopeasti asiakkaiden tarpeisiin. Yrityksellä 1 on ollut asiakkaita sekä julkiselta että yksityiseltä sektorilta. Pilottihaastatteluun osallistunut haastateltava oli tästä yrityksestä.

	Työntekijöitä	Asiakkaita	Toiminut alalla	Toimipisteitä
<i>Yritys 1</i>	Alle 20	Julkiselta ja yksityiseltä sektorilta	9 vuotta	1
<i>Yritys 2</i>	Yli 200	Julkiselta ja yksityiseltä sektorilta	15 vuotta	3
<i>Yritys 3</i>	Yli 300	Julkiselta ja yksityiseltä sektorilta	10 vuotta	4

Taulukko 6: Yritysten perustietoja

Yritys 2 on 2000-luvun alusta saakka toiminut yli 200 työntekijän ohjelmistoyritys. Se tarjoaa asiakkailleen sähköisten palveluiden suunnittelun ja toteutuksen lisäksi pilvipalveluita sekä monipuolisesti erilaisia koulutus- ja konsultointipalveluita. Yrityksellä 2 on asiakkaita sekä julkiselta että yksityiseltä sektorilta. Tästä yrityksestä pääsin haastattelemaan vanhempaa suunnittelijaa sekä tuoteomistajan roolissa toimivaa henkilöä.

Yritys 3 on toiminut ohjelmistoalalla 10 vuotta ja sillä on yli 300 työntekijää. Se on pääasiassa suuntautunut ohjelmistojen tuotekehityspalveluiden tarjoamiseen asiakkailleen, mutta tarjoaa myös ketterään ohjelmistokehitykseen liittyvää konsultointia ja koulutuksia. Yrityksellä on asiakkaita julkiselta ja yksityiseltä sektorilta.

Haastatteluihin tavoitettiin kuusi henkilöä. Haastateltujen taustatiedot on esitetty taulukossa 7. Olen nimennyt haastateltavat joko heidän antamansa nykyisen työroolin tai esimerkkiprojektissa heillä olleen työroolin mukaan. Haastateltavista enemmistö oli toiminut käyttäjäkokemukseen liittyvien tehtävien parissa, mutta useimmilla oli kokemusta myös ohjelmistotuotannon muista tehtäväkentistä. Lähes kaikilla haastatelluilla oli käyttäjäkokemukseen ja käyttöliittymiin liittyvistä tehtävistä yli viiden vuoden kokemus. Ainoana poikkeuksena tähän oli projektipäällikkö, jolla ei ollut lainkaan kokemusta käyttöliittymien tehtäväalueelta, mutta muista esitietolomakkeella kysytyistä ohjelmistokehityksen osa-alueista hänellä oli yli viiden vuoden kokemus.

	Sovelluskehittäjä – yritys 1 –	Vanhempi suunnittelija – yritys 2 –	Tuoteomistaja – yritys 2 –
<i>Projektinhallinta- tehtävistä</i>	1 - 3 vuotta	Yli 5 vuotta	Alle vuosi
<i>Tarpeiden ja vaatimusten määrittelytehtävistä</i>	3 - 5 vuotta	Yli 5 vuotta	Yli 5 vuotta
<i>UX- ja UI- suunnittelutehtävistä</i>	Yli 5 vuotta	Yli 5 vuotta	Yli 5 vuotta
<i>Järjestelmän suunnittelutehtävistä</i>	Yli 5 vuotta	Ei lainkaan	Alle vuosi
<i>Kehitystehtävistä</i>	Yli 5 vuotta	Ei lainkaan	Alle vuosi

	Projektipäällikkö – yritys 3 –	Palvelusuunnittelija – yritys 3 –	UX-suunnittelija – yritys 3 –
<i>Projektinhallinta- tehtävistä</i>	Yli 5 vuotta	Yli 5 vuotta	Yli 5 vuotta
<i>Tarpeiden ja vaatimusten määrittelytehtävistä</i>	Yli 5 vuotta	Yli 5 vuotta	Yli 5 vuotta
<i>UX- ja UI- suunnittelutehtävistä</i>	Ei lainkaan	Yli 5 vuotta	Yli 5 vuotta
<i>Järjestelmän suunnittelutehtävistä</i>	Yli 5 vuotta	Yli 5 vuotta	Ei lainkaan
<i>Kehitystehtävistä</i>	Yli 5 vuotta	Yli 5 vuotta	Ei lainkaan

Taulukko 7: Haastateltavien taustatiedot

UX-suunnittelijalla ja vanhemmalla suunnittelijalla ei ollut lainkaan kokemusta järjestelmien teknisestä suunnittelusta tai kehitystehtävistä. Sen sijaan palvelusuunnittelijalla oli kaikista haastatelluista kattavin kokemus, sillä hänellä oli yli viiden vuoden kokemus kaikista kysytyistä osa-alueista. Neljällä haastatelluista oli yli viiden vuoden kokemus projektinhallintatehtävistä ja jokaisella haastatellulla oli kokemusta niistä ainakin hieman.

4. Kohdeprojektien kuvaukset

Haastateltavia oli pyydetty etukäteen miettimään yhtä projektia, joka oli opettanut heille eniten viestinnästä asiakkaan kanssa. Tässä luvussa esittelen haastateltavien haastatteluun valitsemat projektit ja lopussa kuvaan lyhyesti haastatteluissa esiin nousseita sähköisiä työkaluja.

Jokaisen kuvauksen yhteyteen olen laatinut kuvaajan siitä, miten projekti on sisäisesti organisoitu. Kuvaajissa olen pyrkinyt esittämään, mikä oli haastatellun paikka projektissa ja millaisen ketjun läpi viestittävät asiat kulkivat matkalla asiakkaalta kehitystiimille. Olen myös karkealla tavalla esittänyt, miten projektiin osallistuvat toimijat, asiakas mukaan lukien, olivat jakautuneet maantieteellisesti.

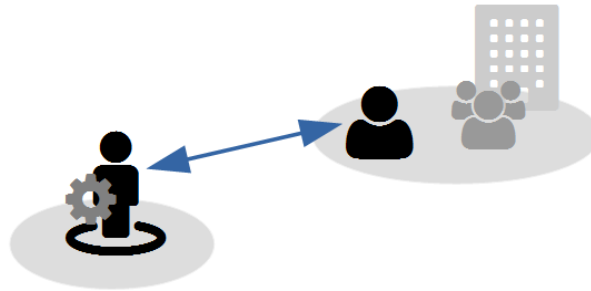
4.1. Projekti 1 – Web-sovellus

Projektissa oli kyse perinteisen webpalvelun kehitysprojektista. Sen perustiedot on koottu taulukkoon 8. Projektin aikataulu oli haastateltavan mukaan tiukka ennalta määritellyn valmistuspäivän suhteen, mutta hän ei halunnut tarkemmin kertoa projektin ajallisesta kestosta. Vaaditut toiminnallisuudet saatiin toimitettua asiakkaalle määräajassa ja viimeiset hienosäädöt ylittivät alkuperäisen aikataulun vain viikolla.

Sovelluskehittäjä	
<i>Projektin kesto</i>	-
<i>Kehitystiimin koko</i>	1-3 hengen kehitystiimi
<i>Tiimin hajautuminen</i>	Samassa tilassa
<i>Käytetty kehitysmenetelmä</i>	Sovellettu Scrum-Kanban-yhdistelmä
<i>Asiakkaan osallistuminen</i>	Tapaamisia parin viikon välein
<i>Viestintään käytetyt työkalut</i>	Trello, puhelimet, sähköposti, Googlen taulukkolaskentaohjelma, demoympäristö

Taulukko 8: Haastatellun sovelluskehittäjän esittelemä projekti

Kuvassa 6 on kuvaaja projektin 1 rakenteesta. Vaaleanharmaa ellipsi kuvaa yksittäistä maantieteellistä toimipistettä, ja tummat ihmishahmot ovat projektiin osallistuneita toimijoita. Ihmishahmojen määrä ei kuvaa toimijoiden todellista määrää, joskin yksinään esitetyt ihmishahmot esittävät yksittäisiä avainhenkilöitä. Ratas ihmishahmojen yhteydessä tarkoittaa, että kyseinen henkilö tai ryhmä toimii kehitystehtävissä projektissa. Ihmishahmo tumman renkaan sisällä kuvaa tutkimukseen osallistunutta haastateltavaa.



Kuva 6: Projekti 1, web-sovellus

Kehitystyö aloitettiin nolapisteestä, joskin asiakkaalta oli olemassa nopeasti tehty prototyyppi, joka tarjosi idean siitä, mitä haluttiin. Kehitystiimin koko vaihteli projektin aikana yhdestä kolmeen henkilöön ja toisinaan projektin hallinnollisiin tehtäviin saatiin apua kehitystiimin ulkopuolelta. Pääasiassa haastateltava vastasi yksin kehitystyön etenemisestä. Asiakas osallistui projektin suunnitteluvaiheeseen sekä tarjosi oman kehittäjänsä avuksi tiettyihin projektin vaiheisiin. Kehitystiimi ei tehnyt töitä asiakkaan luona vaan toimi omissa toimitiloissaan. Tiimi ei ollut maantieteellisesti hajautettu vaan kaikki tekivät työtä samalla toimistolla.

Projektissa käytettiin räätälöityä Scrum-Kanban-menetelmää: kehitystyö jaettiin projektissa parin viikon sprintteihin ja sprinttien sisällä kehitystyöt pilkottiin Kanban-työkalulla hallinnoitaviin kortteihin. Kehitystiimi tapasi asiakkaan kanssa kasvotusten sprinttien välissä ja tuolloin tarkistettiin töiden etenemistä sekä päätettiin tehtävät työt seuraavaa sprinttiä varten.

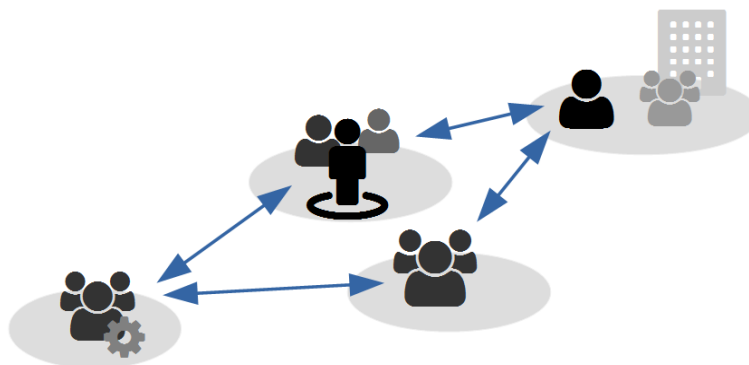
Projektin viestinnässä käytettiin apuna Trello-palvelua, joka toimi asiakkaan ja kehittäjien yhteisenä Kanban-työkaluna ja tehtävälistanä, puhelinta kiireellisiin viestintätarpeisiin, sähköpostia sekä verkossa jaettua taulukkolaskentaohjelmaa.

4.2. Projekti 2 – Kansainvälinen konsulttiprojekti

Tämä projekti oli jatkunut useamman vuoden, haastateltava kertoi tulleensa mukaan yli kaksi vuotta takaperin (taulukko 9). Haastateltavan oma työrooli on ollut toimia asiakasta edustavana konsulttina projektissa. Projektilla oli ulkoinen toimittaja eli kehitystiimi ei ollut haastateltavan omasta yrityksestä. Kehitystiimin koko oli 12 henkeä ja asiakasta edusti kaksi konsulttiyritystä, joista molemmista projektiin osallistui kolme konsulttia. Haastatellun työnantaja oli toinen näistä konsulttiyrityksistä.

Vanhempi suunnittelija	
<i>Projektin kesto</i>	Yli 2 vuotta (jatkuu edelleen)
<i>Kehitystiimin koko</i>	3 + 3 asiakasta edustavaa konsulttia ja 12 hengen kehitystiimi
<i>Tiimin hajautuminen</i>	Toteutustiimi Tampereella ja konsultit Tampereella 2 yrityksessä
<i>Käytetty kehitysmenetelmä</i>	Scrum
<i>Asiakkaan osallistuminen</i>	Haastateltava toimi asiakkaan edustajana, viikoittain tilaajan sisäinen palaveri, daily scrum asiakkaan edustajien ja kehitystiimin välillä
<i>Viestintään käytetyt työkalut</i>	Skype, puhelimet, sähköposti, Lync-videopalaverit, Jira, Confluence

Taulukko 9: Haastatellun vanhemman suunnittelijan esittelemä projekti



Kuva 7: Projekti 2, kansainvälinen konsulttiprojekti

Projektin osapuolet olivat maantieteellisesti hajautettuja (kuva 7). Tuotteen omistajat olivat Suomesta ja Ruotsista, asiakas oli Helsingissä, toteutustiimi toimi Tampereella ja myös ulkopuoliset konsultit työskentelivät Tampereella kahdessa eri yrityksessä. Toisen konsulttiyrityksen vastuulla oli projektin tekninen puoli ja haastateltavan yritys edusti liiketoiminnan puolta projektissa.

Projektissa käytettiin Scrum-menetelmää projektinhallintaan kahden viikon sprinteissä. Asiakas ei osallistunut suoraan projektiin vaan konsultit pyrkivät valvomaan asiakkaan etua. Joka viikko oli tilaajan sisäinen palaveri, jossa konsultit ja asiakas kävivät läpi, mitä tapahtuu, missä aikataulussa ja mitä päätöksiä tarvitsee tehdä. Tämän tiedon pohjalta konsultit pyrkivät edistämään toteuttajien parissa näitä asioita.

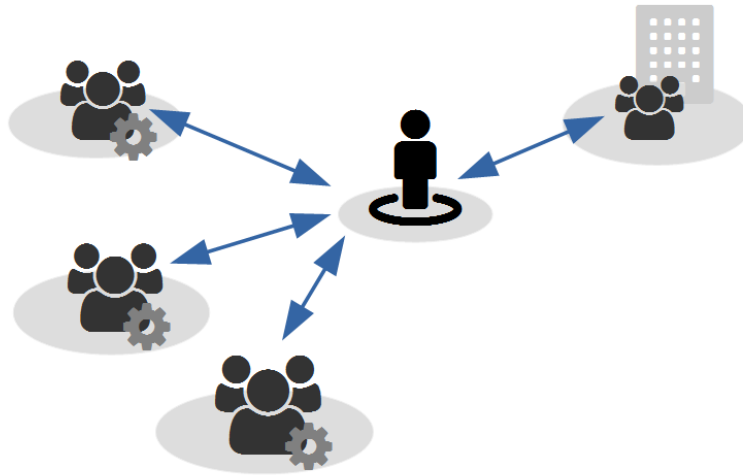
Viestintävälineinä projektissa oli käytössä Skype ja puhelin päivittäiseen nopeaan viestintään. Sähköpostia käytettiin sekä ruotsalaisille asiakkaille viestintään että viestintään vaikeasti tavoitettavien loppukäyttäjien kanssa. Videoneuvotteluja hyödynnettiin isompiin linjauksiin ja päätösten tekoon. Jira toimi projektin tehtävien hallintaan ja Confluence-wikiä käytettiin dokumentaatioon.

4.3. Projekti 3 – Kolmen kehitystiimin ohjaus

Projekti on kestänyt neljä vuotta ja sen tarkoitus on jatkaa vielä kaksi vuotta (taulukko 10). Haastateltava on ollut mukana kolme vuotta. Kehitystiimejä on kolmesta eri yrityksestä, kun taas haastateltava itse toimii asiakkaan edustajana alihankkijana kehitystiimien ja asiakkaan välissä. Projektissa on 25-30 kehittäjää ja kaikkiaan projektissa on työskennellyt 35 toimijaa.

Tuoteomistaja	
<i>Projektin kesto</i>	4 vuotta (jatkuu vielä 2 vuotta)
<i>Kehitystiimin koko</i>	kolmesta yrityksestä kaikkiaan 25-30 toimijaa
<i>Tiimin hajautuminen</i>	Kehittäjät omien yritystensä toimipisteissä, asiakasta edustavia konsultteja Tampereella ja Helsingissä
<i>Käytetty kehitysmenetelmä</i>	Scrum ja Kanban
<i>Asiakkaan osallistuminen</i>	Haastateltava toimi asiakkaan edustajana, asiakas osallistui kahden viikon välein preplanning- ja review-tapaamisiin
<i>Viestintään käytetyt työkalut</i>	Sähköposti, puhelimet, Skype, Flowdoc

Taulukko 10: Haastatellun tuoteomistajan esittelemä projekti



Kuva 8: Projekti 3

Tiimit työskentelevät omissa toimipisteissään ja asiakasta edustaneessa ryhmässä on työntekijöitä kahdesta eri yrityksestä (kuva 8). Haastateltavan edustaman yrityksen neljästä projektiin osallistuvasta henkilöstä kaksi työskentelee Helsingistä käsin ja kaksi Tampereelta.

Kaksi projektin kehitystiimeistä käytti Scrum-menetelmää ja yksi tiimi siirtyi ajan myötä käyttämään Kanban-menetelmää. Scrum-menetelmää käyttävät tiimit, joiden sisällä vaihtuvuus on ollut suurinta ja Kanban-menetelmään siirtyi tiimi, jonka tekijät ovat pitkään pysyneet samana. Alussa noudatettiin tiukasti Scrum-menetelmää, jotta kaikille kehitystiimeille muodostuisi käsitys kokonaisuudesta ja oman tiimin työn paikasta osana kokonaisuutta.

Haastateltava edusti projektissa asiakasta. Myös asiakkaan oma tuoteomistaja osallistui aktiivisesti mukaan projektiin, joskin kiire rajoitti kasvokkaista vuorovaikutusta vain sprinttien välissä pidettäviin suunnittelu- ja arviointitilaisuuksiin. Projektissa viestinnässä hyödynnettiin sähköpostia, puhelinta, Skypeä ja pikaviestinohjelma Flowdockia.

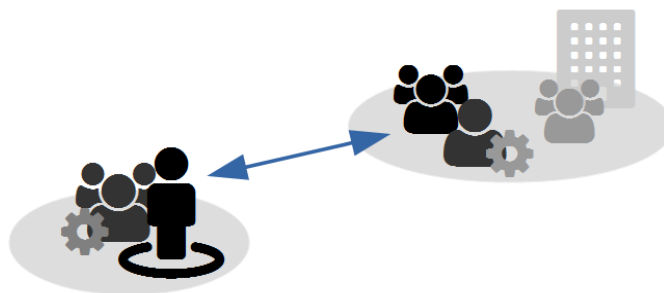
4.4. Projekti 4 – Lahoava järjestelmä vaihtoon

Projekti kesti kolme vuotta ja siinä oli viiden hengen kehitystiimi (taulukko 11). Alkuperäinen tavoite järjestelmän käyttöönotolle oli ensimmäisen vuoden jälkeen, mutta projektin aikana ilmeni erilaisia ongelmia, jonka seurauksena projektin lopullinen kesto oli kolme vuotta. Kehitystiimi toimi samassa huoneessa Tampereella, mutta

loppua kohden yksi tiimiläisistä muutti Helsinkiin ja teki töitä sieltä käsin (kuva 9). Asiakkaalta osallistui yksi kehittäjä projektiin ja hän toimi asiakkaan tiloissa.

Projektipäällikkö	
<i>Projektin kesto</i>	3 vuotta
<i>Kehitystiimin koko</i>	5 hengen kehitystiimi
<i>Tiimin hajautuminen</i>	Asiakkaan oma kehittäjä toimi asiakkaan tiloissa, muut kehittäjät olivat pääasiassa samassa tilassa
<i>Käytetty kehitysmenetelmä</i>	Ei tiettyä menetelmää, jatkuvalla integraatiolla tiiviisti asiakkaan kanssa vuorovaikutuksessa
<i>Asiakkaan osallistuminen</i>	Eritasoisia yhteyshenkilöitä asiakkaalta, joihin kehitystiimi oli tiiviisti yhteydessä päivittäin
<i>Viestintään käytetyt työkalut</i>	Skype (chat, ruudunjako), puhelimet, Google Drivessä jaettuna vaatimustiedostoja, sähköposti, Trello, Jira

Taulukko 11: Haastatellun projektipäällikön esittelemä projekti



Kuva 9: Projekti 4, lahoava järjestelmä vaihtoon

Projektissa ei käytetty mitään tiettyä vakiintunutta menetelmää, mutta kyseessä oli jatkuvan integraation projekti. Järjestelmää kehitettiin iteratiivisesti pieninä paloina kerrallaan, jotka nopeasti päivitettiin asiakkaalle testattavaksi demoympäristöön.

Asiakkaalta oli useita eritasoisia yhteyshenkilöitä, joihin oltiin yhteydessä toisinaan päivittäinkin: budjetista ja loppuvaiheessa kehitettävien ominaisuuksien tärkeysjärjestyksestä huolehtiva ohjausryhmä, toimialan ja järjestelmän suunnittelun yhteen saattava tuoteomistaja sekä asiakkaan palveluksessa oleva ohjelmistokehittäjä.

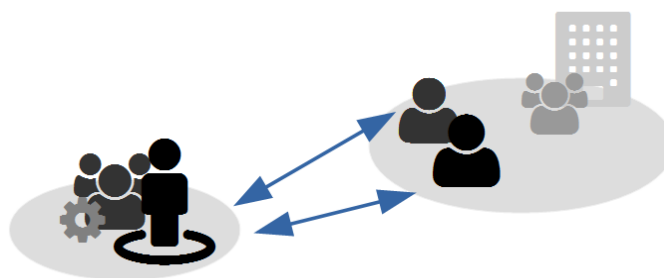
Projektissa käytettiin Skypeä ruudunjakoon ja pikaviestintään, puhelinta käytettiin kiireellisissä tapauksissa, dokumentaatio jaettiin pilvipalvelun avulla, sähköpostia käytettiin muodolliseen viestintään, josta oli tarpeen jäädä jälki. Trelloa käytettiin aluksi projektinhallintaan, mutta siitä siirryttiin pian ongelmien vuoksi Jiraan.

4.5. Projekti 5 – Perustukset kuntoon

Projektissa on lähdetty vaihtamaan ongelmallista vanhaa ohjelmistoversiota uudempaan, jota on lähdetty rakentamaan kokonaan uusiksi ”perustukset kuntoon” -periaatteella. Projekti on ollut käynnissä vajaan vuoden, mutta tulee jatkumaan vielä ainakin vuoden verran (taulukko 12). Haastateltava on ollut mukana projektissa alusta saakka.

Palvelusuunnittelija	
<i>Projektin kesto</i>	10 kuukautta (jatkuu edelleen)
<i>Kehitystiimin koko</i>	7 hengen kehitystiimi
<i>Tiimin hajautuminen</i>	Samassa tilassa
<i>Käytetty kehitysmenetelmä</i>	Scrum soveltaen
<i>Asiakkaan osallistuminen</i>	Asiakkaalta kaksi yhteyshenkilöä
<i>Viestintään käytetyt työkalut</i>	Slack, Trello, SharePoint, Skype

Taulukko 12: Haastatellun palvelusuunnittelijan esittelemä projekti



Kuva 10: Projekti 5

Kehitystiimi toimii samassa tilassa Tampereella ja asiakkaan tilat sijaitsevat Helsingissä (kuva 10). Projektiin valittu menetelmä oli haastateltavan mukaan ”soveltaen Scrumia Kanban-hengessä”. Aluksi asiakkaalta oli yhteyshenkilönä ainoastaan markkinointipäällikkö, mutta kun tarve tekniselle tietämykselle asiakkaan järjestelmistä kävi ilmeiseksi, saatiin asiakkaalta mukaan myös asiakkaan omat järjestelmät tunteva yhteyshenkilö. Asiakkaan yhteyshenkilöt osallistuvat päivittäin nopeisiin tilannekatsauksiin Skypen välityksellä, jotta keskitetysti pystyttiin pitämään asiakas ajan tasalla edistyksestä ja kehittäjät saivat tarvitsemansa vastaukset asiakkaalta.

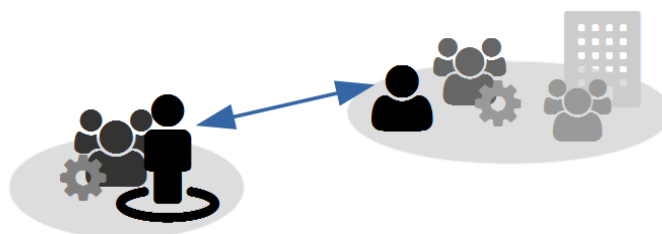
Viestintään käytettiin pikaviestintä päivittäiseen viestintään asiakkaan kehittäjien ja kehitystiimin välillä, Skypeä käytettiin päivittäisiin tilannekatsauksiin, Trellolla hallinnoidaan projektiin liittyviä kehitystehtäviä ja asiakkaan havaitsemia bugeja, puhelinta on hyvin harvoin tarvinnut päivittäisten tilannekatsausten vuoksi, SharePointtia on käytetty asiakkaan omien taulukkolaskentatiedostojen jakoon.

4.6. Projekti 6 – Vanhan järjestelmän päivitys ja korvaus

Projekti on kestänyt kuusi kuukautta ja se jatkuu edelleen noin vuoden verran, mutta koska kyseessä on iteratiivinen projekti, ei tarkkaa loppupäivämäärää ole tiedossa (taulukko 13). Kehitystiimi on kooltaan 3-4 henkilöä ja asiakkaalta on muutamia omia kehittäjiä mukana projektissa. Kehitystiimi työskentelee samassa tilassa Tampereella pois lukien asiakkaan palveluksessa olevat ohjelmistokehittäjät, jotka toimivat asiakkaan omissa tiloissa.

UX-suunnittelija	
<i>Projektin kesto</i>	6 kuukautta (jatkuu edelleen)
<i>Kehitystiimin koko</i>	3-4 hengen kehitystiimi ja muutamia asiakkaan omia kehittäjiä
<i>Tiimin hajautuminen</i>	Kehitystiimi samassa tilassa ja asiakkaan kehittäjät asiakkaan tiloissa
<i>Käytetty kehitysmenetelmä</i>	Iteratiivista kehitystä
<i>Asiakkaan osallistuminen</i>	Asiakkaalla yhteyshenkilö ja omia kehittäjiä mukana
<i>Viestintään käytetyt työkalut</i>	Sähköposti, puhelin, Skype

Taulukko 13: Haastatellun UX-suunnittelijan esittelemä projekti



Kuva 11: Projekti 6

Asiakkaalta on yksi avainhenkilö, joka ei kuitenkaan toimi varsinaisesti tuoteomistajan roolissa (kuva 11). Asiakkaan kanssa tavataan kasvokkain jokaisen kehitysvaiheen alussa ja lopussa. Mikäli joudutaan kesken kehitysvaihetta poikkeamaan sovitusta tai vastaan tulee ennakoimattomia ongelmia, saatetaan asiakkaan kanssa myös silloin tavata kasvokkain.

Sähköpostilla ja puheluilla annetaan statuspäivityksiä kerran viikossa. Sähköposti ja Skype ovat tyypillisimmät viestintäkanavat tässä projektissa.

5. Tulokset

Tässä luvussa käyn läpi haastatteluista saatuja tuloksia. Kävin läpi haastattelut ja etsin niistä yhteneviä kommentteja ja teemoja. Olen nostanut esiin näitä haastatteluissa toistuneita teemoja omiksi kohdiksi. Kuvatessani esiin nousseita havaintoja käytän kursiivia niissä ilmauksissa, jotka ovat suoraan haastateltavan käyttämiä ilmauksia. Merkintä [...] tarkoittaa, että tekstistä on jätetty osa pois, ja hakasulkeissa olevat tekstinpätkät ovat omia tulkintojani haastateltavan käyttämistä sanoista.

5.1. Yritysten menetelmävalinnat

Yhdessäkään haastatelluista yrityksistä ei käytetty kaikkiin projekteihin etukäteen määriteltyä menetelmää, kuten Scrum tai Kanban, vaan ne vaihtelivat projekteittain. Esimerkiksi yrityksessä 1 valittiin projektin alussa kyseiseen projektiin sopivimmat toimintakäytännöt:

”Oikeastaan siinä projektin alussa mietittiin, mikä olis sopiva setti noita projektimenetelmiä, että tehtiin sillain parin viikon iteraatioissa sprinttejä ja siinä tavallaan yhen sprintin aikana menttiin vähän tällöisellä kanbanimaisella lähestymistavalla.”

– Sovelluskehittäjä

Yrityksen 2 haastateltujen projekteissa molemmissa hyödynnettiin Scrum-menetelmää. Haastatellun tuoteomistajan projektissa Scrum ei kuitenkaan ollut ainoa käytetty menetelmä vaan sitä käytettiin tutustumisvaiheena kehittäjille. Pisimpään yhdessä toiminut tiimi siirtyi Kanbaniin, kun yhteistyö oli alkanut sujumaan. Projektin pitkän keston seurauksena kehittäjien vaihtuvuus osassa kehitystiimeissä oli suurta, mikä on vaikuttanut valittuihin menetelmiin:

”Siin on sen verran ollut joillakin toimittajilla vaihtuvuutta resurssoinnissa että heidän kanssaan on jauhettu ihan klassisella Scrummilla. Mut siinä porukassa missä pysyvyys on ollut sama niin he ovat niinku Scrummista maturoituneet Kanbaniin. Se on aika luontevaa, kun ensin teet Scrummilla, kun se on enemmän tällöinen tutustumisvaihe, missä opitaan mitä kukakin osaa ja jossain vaiheessa pystyy sit maturoitumaan Kanban-malliin”

– Tuoteomistaja

Myös yrityksessä 3 painotettiin sopivimpien työkalujen ja menetelmien valintaa projektikohtaisesti. Taustalla on kuitenkin ketterien menetelmien tapa edetä pieni pala kerrallaan ja jatkuvasti tarkistaa kurssia näistä pienistä paloista saadun palautteen perusteella:

”Mä en usko että firmassa on niinku yhtä semmosta tiettyä kirjoitettua sanaa, jonka mukaan aina mennään, että toimintatapaan kuuluu se, että epäonnistutaan nopeasti eli yritetään saada mahdollisimman nopeasti jotain toimivaa aikaseksi. Nopeella iteraatiolla katotaan että mennäänkö oikeeseen suuntaan [...], mutta se ei edellytä minkään tietyn menetelmän käyttämistä.”

– Projektipäällikkö

Yrityksestä 3 haastateltu palvelusuunnittelija kertoi, että hänen projektissaan sovellettiin Scrumia Kanban-hengessä. Näitä menetelmiä ei kuitenkaan noudatettu pilkuntarkasti vaan työskentelytapoja muokattiin tarpeen vaatiessa:

”Aika soveltaen sen mukaan fiilispohjalta [tehdään], että mikä soveltuu tilanteeseen parhaiten. Meillä muutenkin ajatuksena on se, että testataan jotain työskentelytapaa. Jos se ei toimi, niin muutetaan sitä aika nopeasti matkan varrella.”

– Palvelusuunnittelija

UX-suunnittelija yrityksistä 3 kertoi, että hänen projektissaan edettiin iteratiivisesti, mutta mihinkään tiettyyn projektimenetelmään ei sitouduttu. Kehitystyö oli jaettu kehitysjaksoihin, joiden sisällä tehtiin jakson alussa päätetty kokonaisuus. Jokaisen kehitysjakson kesto arvioitiin aina tarpeen mukaan, sen sijaan että tietyn ajanjakson sisälle olisi pyritty sovittamaan sopiva määrä tehtäviä.

Haastateltujen esittelemät projektit poikkesivat siis toisistaan projektimenetelmien ja valittujen viestintävälineidenkin kannalta. Eroja oli jopa saman yrityksen sisällä

toteutetuissa projekteissa. Jokaisessa projektissa kuitenkin edettiin aina pieni pala kerrallaan sekä jatkuvasti toimintatapoja arvioiden ja tarpeen tullen niitä muuttaen.

5.2. Projekteissa käytetyt sähköiset työkalut

Haastateltujen projekteissa käytettiin laajaa kirjoa erilaisia sähköisiä työkaluja viestinnän tarpeisiin. Viestintävälineiden valinnat tehtiin aina projektikohtaisesti vaikka samoja työkaluja on löydettävissä useammasta eri projektista. Esimerkiksi vanhemman suunnittelijan projektissa ei ollut Skypeä käytössä ensimmäisen ohjelmistotoimittajan kanssa vaan siihen siirryttiin vasta ohjelmistotoimittajan vaihdoksen yhteydessä.

Osaan näistä työkaluista viitattiin vain niiden yleisellä nimityksellä, kuten puhelin ja sähköposti, ja osaan työkaluista viitattiin niiden nimellä. Esittelen seuraavaksi nimeltä mainittujen työkalujen ominaisuuksia. Olen järjestänyt esiteltävät työkalut niin, että esittelen ensimmäisenä korkeimman synkronisaatiotason viestintämediat ja etenen kohti asynkronisempia medioita.

5.2.1 Skype ja Lync

Skype ja Lync ovat molemmat VoIP-palveluita (Voice over Internet Protocol), jotka sisältävät myös joukon muita ominaisuuksia. Lync on nykyisin nimeltään Skype for Business. Skype sisältyy Microsoftin Office 365 -pakettiin ja Skype for Business kuuluu tiettyihin yrityksille suunnattuihin Microsoftin Office 365 -paketteihin.

Skype tarjoaa äänipuhelut ja videopuhelut Skypesta toiseen, mutta myös äänipuhelut matka- ja lankapuhelimiin. Kahdenkeskisten puheluiden lisäksi ryhmäpuhelut ja ryhmävideopuhelut ovat mahdollisia. Skypellä on myös mahdollista lähettää pikaviestejä yksittäisille ihmisille tai perustaa keskusteluryhmiä. Pikaviesteihin voi sisällyttää hymiöitä ja käyttäjä voi myös lähettää video- ja ääniviestejä. Skypellä voi lähettää tiedostoja ja jakaa näytön yhdelle tai useammalle käyttäjälle.

Skype for Business (entinen Lync) sisältää samat ominaisuudet kuin Skype, mutta sen kokoustoiminnallisuuksia on paranneltu. Esimerkiksi ryhmäkeskusteluihin voi osallistua useita satoja käyttäjiä ja jaettu video on HD-laatuista. Skype for Business tarjoaa myös integroinnin Microsoft Office-sovelluksiin.

5.2.2 Flowdock ja Slack

Flowdock ja Slack ovat pikaviestisovelluksia, jotka on suunniteltu erityisesti ryhmäkeskusteluja varten. Molemmat tarjoavat mahdollisuuden organisoida keskusteluryhmiä tiettyjen aihealueiden ympärille. Aiheille luodaan Flowdockissa ”virtauksia” (flow) ja Slack käyttää samasta asiasta termiä kanava (channel). Käytännössä nämä ovat keskusteluhuoneita, joihin käyttäjät voivat liittyä.

Molemmissa ohjelmissa on mahdollisuus kahdenkeskisiin keskusteluihin ja Slack tarjoaa jopa mahdollisuuden ääni- ja videopuheluihin. Sekä Slack että Flowdock tarjoavat hakutyökalun, jonka avulla voi helposti etsiä sisältöä vanhoista keskusteluista. Molemmissa sovelluksissa voi jakaa tiedostoja ja kumpaankin on mahdollista integroida muita sovelluksia, jolloin muissa sovelluksissa tapahtuvat muutokset tuottavat ilmoituksia pikaviestimeen.

5.2.3 Trello ja Jira

Trello ja Jira ovat projektinhallintaa varten kehitettyjä ohjelmistoja. Trello on suurelle yleisölle suunnattu työkalu, jota voi helposti käyttää erilaisten työ- ja harrasteprojektien organisointiin ja seurantaan. Jira on erityisesti ketterän ohjelmistokehityksen tarpeisiin suunnattu projektinhallinnan työkalu. Molemmat palvelut toimivat verkon yli, joten käyttäjillä on pääsy niihin mistä vain. Yrityksillä on myös mahdollisuus asentaa Jira omille palvelimilleen, mikäli he haluavat kontrolloida enemmän palvelun ylläpitoa ja palvelimen tietoturva.

Trello perustuu Kanban-taulun ajatukseen: siinä oma taulu projektille ja tauluun luodaan kortteja, joita voidaan siirtää luettelosta toiseen edistymisen kuvaamiseksi. Kortteihin voi lisätä muun muassa liitetiedostoja, määräaikoja, tarkistuslistoja ja nimilappuja. Toiminnallisuuksia voi laajentaa erilaisilla lisäosilla, joilla on mahdollista linkittää muiden sovellusten ominaisuuksia Trelloon. Trello mahdollistaa taulujen ja käyttäjän palveluun kutsumien kollegojen organisoimisen työryhmiin. Trelloon voi myös luoda kortteja ja kommentteja sähköpostin avulla.

Jirassa voi hallinnoida projektin työtehtäviä joko Kanbanilla tai Scrumilla. Jira tarjoaa työkalut projektien suunnitteluun ja seuraamiseen. Siinä voi kirjata käyttäjätarinoita ja luoda tehtäväkortteja, joita on mahdollista linkittää käyttäjätarinoihin. Korteille voi määrittää kuvauksen lisäksi esimerkiksi kriittisyysasteen, määräajan, tekijän ja työmääräarvion. Kortteja voi kommentoida ja korttiin voi lisätä käyttäjiä, joille lähetetään sähköpostia joka kerta, kun kortin tiedot

päivittyvät. Jirassa pystyy hallinnoimaan projektin aikataulua ja voi seurata reaaliajassa työn edistymistä erilaisten kuvaajien avulla.

5.2.4 Confluence

Confluence on wiki-tyylinen ryhmäohjelma Jiran valmistajalta. Käyttäjät voivat luoda sivuja ja tuottaa sisältöä niille ja muut käyttäjät voivat puolestaan kommentoida tuota sisältöä. Confluence on integroitu Jiraan, joten sillä on mahdollista suunnitella tiimien töitä ja saada tilanteet päivittymään automaattisesti Jirasta halutuille sisältsivuille. Confluence tarjoaa työkalut valitun sisällön rajoittamiseen vain tietyille käyttäjille, dokumenttien hallinnan työkaluja, luodun sisällön versiointia sekä joukon erilaisia lisäosia. Lisäosat tarjoavat esimerkiksi integraatioita eri sovelluksiin, kuten kalentereihin ja sähköpostiohjelmiin, sekä työkaluja kyselyiden, kaavioiden ja erilaisten tulostaulujen tuottamiseen.

5.2.5 Google Drive ja SharePoint

Google Drive ja SharePoint ovat pilvipalveluita, joiden ensisijainen tarkoitus on mahdollistaa tiedostojen jakaminen helposti. Molemmat mahdollistavat tietyn käyttäjäryhmän kesken jaettujen kansioden luomisen ja niihin tallennettujen tiedostojen jakamisen tuolle käyttäjäryhmälle.

5.2.6 Käytettyjen viestintämedioiden ilmaisuvoimaisuus

Taulukossa 14 on asetettu esitellyt viestintämediat niiden edustamaan median tyyppiin (vrt. taulukko 4) sekä annettu linkki, josta löytyy lisätietoa näistä projekteissa käytetyistä viestintämedioista. Skype ja Lync listautuvat sekä videokonferenssin että pikaviestimen alle, koska niissä on mahdollisuus molempiin.

Viestintämedian tyyppi	Projekteissa käytetty viestintämedia	URL
<i>Videokonferenssi</i>	Skype	https://www.skype.com/fi/
	Lync	https://products.office.com/fi-fi/skype-for-business/online-meetings
<i>Pikaviestin</i>	Skype	https://www.skype.com/fi/
	Lync	https://products.office.com/fi-fi/skype-for-business/online-meetings
	Flowdock	https://www.flowdock.com/features
	Slack	https://slack.com/
<i>Asynkroninen ryhmäohjelma</i>	Trello	https://trello.com/
	Jira	https://www.atlassian.com/software/jira
	Confluence	https://www.atlassian.com/software/confluence
	Google Drive	https://www.google.com/drive/
	SharePoint	https://products.office.com/fi-fi/sharepoint/collaboration

Taulukko 14: Käytettyjen viestintämedioiden sijoittuminen viestintämedioiden tyyppiin

Versionhallintaohjelmat voisi myös lukea asynkronisiksi ryhmäohjelmiksi ja vaikka niitä projekteissa käytettiin, ei asiakkailla ollut halua tarkastella tuotettua koodia. Tästä syystä versionhallintaa ei ole mainittu näissä listoissa.

5.3. Projektien ajallisen keston vaikutus viestintään

Haastateltujen valitsema projekti olivat kestoltaan hyvin erimittaisia ja lähes jokainen esitellyistä projekteista oli edelleen kesken. Lyhimmän aikaa käynnissä ollut projekti oli yrityksen 3 UX-suunnittelijan esittelemä kuusi kuukautta haastattelun aikaan jatkunut projekti, jossa tarkoitus oli aluksi päivittää asiakkaan nykyinen järjestelmä ja pitemmällä aikavälillä ”räjäyttää koko alla oleva järjestelmä” ja tuottaa tilalle paremmin asiakkaan tarpeita vastaava järjestelmä ilman vanhan järjestelmän luomia rajoitteita. Muista projekteista poiketen, tässä projektissa ei ollut määriteltynä lopullista päättämispäivää:

”[Kyseessä] on oikeasti iteratiivinen projekti, eli sitä tehdään pätkissä ja sitten [suunnitellaan] jatkoa. Eli vois aatella, että en tiedä tarkalleen millon se loppuu ajallisesti eli tarkoitus on tehdä yks ohjelmisto valmiiksi. Oletettavasti se valmistuu ens vuoden aikana.”

– UX-suunnittelija

Myös kahden muun yrityksessä 3 työskentelevän haastatellun projektissa oli lähdetty korvaamaan vanhaa järjestelmää. Haastatellun palvelusuunnittelijan esittelemä projekti oli edelleen käynnissä ja oli jatkunut haastatteluhetkellä 10 kuukautta. Haastateltu kertoi, että liikkeellä oltiin isolla seitsemän hengen kehitystiimillä, koska tässä projektissa pitää saada asiakkaalle nopeasti näkyvää tulosta aikaiseksi. Tarkoituksena on korvata asiakkaan nykyinen järjestelmä perustuksista asti uudelleen kehitetyllä järjestelmällä. Yhden suurimmista haasteista muodosti asiakkaan aikaisemmat tietojärjestelmät. Niistä paljastui aikatauluun vaikuttavia yllätyksiä siinä vaiheessa, kun uuteen kehitettävään järjestelmään oli tarkoitus tuoda tietoja vanhoista järjestelmistä. Haastateltu palvelusuunnittelija epäili projektin jatkuvan vielä ainakin vuoden verran ja totesi kyseessä olevan siinä mielessä *”massiivinen, iso projekti”*. Hän myös totesi olleensa mukana alusta asti, josta arveli olevan hyötyä asiakasviestinnässä:

”Mä oon ollu projektissa ihan alusta asti mukana eli oon ollu myymässä sitä projektia asiakkaalle eli tavallaan siinä mielessä on ollu hyvät lähtökohdat asiakasviestinnälle ja ennakko-odotusten hallinnalle.”

– Palvelusuunnittelija

Yrityksestä 3 haastatellun projektipäällikön esittelemä projekti oli kestänyt kolme vuotta. Toinen yritys oli jo yrittänyt toimittaa samaa sisältöä asiakkaalle, mutta huonolla menestyksellä. Asiakkaan alkuperäinen toive edellisen toimittajan kuluttaman ajan vuoksi olisi ollut saada järjestelmä käyttöön jo ensimmäisen vuoden jälkeen, joten kaksi viimeistä vuotta projektista tehtiin kovan paineen alla. Suurimmaksi syyksi aikataulun venähtämiselle haastateltava arvioi sisällön määrittelyn *”fraktaalimaisuuden”*: pintapuolisesti tarkasteltuna määrittelyt vaikuttivat yksinkertaisilta ja helpoilta toteuttaa, mutta yksityiskohtiin mentäessä havaittiin, että yhtä käyttöliittymän näkymää käsittelevän listan *”yksi ranskalainen viiva tarkoittikin itse asiassa kahdeksaa uutta näytettä, joita ei oltu tunnistettu aikaisemmin”*. Kiireestä syntynyt paine muutti myös tiimin tapaa kirjata kehitystehtäviä:

”Loppuvaiheessa alettiin sinne [Jiraan] mallintaan myöskin niitä toiminnallisia tehtäviä että siinä vaiheessa kun alettiin olemaan tarpeeks myöhässä. Ohjausryhmä ja sillä tasolla ruvettiin haluamaan enemmän semmosta niinkuin pillistä kiinni -meininkiä, että jonkinlainen kontrollin tunne siellä, että kun ne näkee mitkä nämä numerot on niin sitten on helpompi olla, kun se epävarmuus poistuu. Eihän se [epävarmuus] mihinkään poistunu mutta ehkä se auttoi siihen tuskaan.”

– Projektipäällikkö

Pisimmät projektit olivat yrityksen 2 haastatelluilla. Toiseksi pisin projekti oli vanhemman suunnittelijan esittelemä projekti, joka oli kestänyt ”useamman vuoden” ja haastateltava oli ollut mukana kaksi vuotta. Kehitystiimi oli jouduttu vaihtamaan kesken projektin ja aikataulupaineiden vuoksi uusi tiimi ei voinut aloittaa puhtaalta pöydältä.

”Tässä on ollut tosiaan yks toimittaja jo aikasemmin, joka on jouduttu vaihtamaan ja nyt tämä uus toimittaja on joutunut lähtemään siitä vanhan pohjasta niinkun siihen rakentamaan ja täs on aikatauluhaasteet, että tosiaan tää pitäis huomenna saada tuotantoon.”

– Vanhempi suunnittelija

Uudella kehitystiimillä oli siis rajallinen aika koota ymmärrys projektista ja kehitettävän ohjelmistotuotteen toimialueesta kokonaisuutena. Aikataulupaineet rajoittivat myös ohjelmistotoimittajan viestintämahdollisuuksia asiakkaan suuntaan:

”Toimittaja varmaan toivois aktiivisempaa tai tiiviimpää kanssakäymistä asiakkaan kanssa ja [tuoteomistajan] ja loppukäyttäjien kanssa, mutta johtuen tästä aikataulusta niin ei oo mahdollisuutta.”

– Vanhempi suunnittelija

Pisin projekti oli yrityksestä 2 haastatellun tuoteomistajan esittelemä projekti, joka oli haastatteluhetkellä jatkunut jo neljä vuotta ja tulisi jatkumaan vielä seuraavat kaksi vuotta. Projektissa kehitettävä järjestelmä on laajuudeltaan muiden haastateltujen projekteihin verrattuna suurin ja tämä on yksi syy sille, miksi järjestelmä on pilkottu kolmeen osakokonaisuuteen kolmelle eri kehitystiimille. Projektissa oli asetettu kiinteitä päivämääriä eri osien toimittamiselle. Tämä aiheutti tilanteita, joissa käyttäjän kannalta vielä keskeneräisiä kokonaisuuksia otettiin tuotantokäyttöön, jotta kokonaisuuteen kuuluva pienempi osa tulisi aikataulun mukaisesti käyttöön. Tämä edellytti avointa

viestintää asiakkaan ja loppukäyttäjien suuntaan, että he ymmärsivät, miksi näin tehtiin ja mistä tällaisessa ketterässä kehityksessä on kyse.

Haastattelujen perusteella projektien ajallisella kestolla on vaikutusta projektien viestintään ja valittuihin menetelmiin. Pitkän projektin aikana kehitystiimi oppii tuntemaan toistensa työskentelytavat ja työskentelyä on mahdollista virtaviivaistaa, mikäli tiimin jäsenet pysyvät jokseenkin samoina. Ennakoidusta aikataulusta poikkeaminen aiheutti asiakkaan vaatimuksesta viestinnän lisäämistä yhdessä projektissa, kun taas toisessa projektissa koettiin, ettei tiiviimmälle viestinnälle ollut kiireen vuoksi aikaa.

5.4. Asiakkaan tavoitettavuus

Asiakkaan osallistuminen projekteihin oli hyvin vaihtelevaa. Osassa projekteista asiakas tarjosi omia kehittäjiään projektin tarpeisiin, yhdessä projektissa asiakas osallistui päivittäin videoneuvotteluihin ja molemmat yrityksen 2 haastateltavat toimivat projektissa asiakkaan edustajina kehittäjien suuntaan.

Yrityksen 3 projektipäällikön esittelemässä projektissa asiakkaalta osallistui yksi asiakkaan oma kehittäjä. Asiakkaan kanssa oltiin päivittäin yhteydessä projektiin liittyvistä teknisistä asioista, sillä järjestelmän määrittely oli hyvin pintapuolinen ja ”fraktaalinen”. Järjestelmää päivitettiin testiympäristöön jatkuvan integraation hengessä ja asiakas seurasi aktiivisesti mukana:

”Hyvin tiheästi niitä katottiin ja asiakas sitten testas niitä myöskin samassa tahdissa, että sieltä tuli kyllä hyvin nopeasti se palauteki sitten aina.”

– Projektipäällikkö

Asiakkaalta osallistui eritasoisia toimijoita vaihtelevalla intensiteetillä. Esimerkiksi asiakkaan oma ohjelmistokehittäjä oli mukana päivittäin kehitystyössä tiimin osana, kun taas ohjausryhmä kokoontui kuukauden välein keskustelemaan budjetista ja projektin etenemisestä. Asiakkaan tuoteomistaja vastasi toimialan ja järjestelmän teknisen toiminnan yhteen saattamisesta. Projektipäällikkö oli yhteydessä tuoteomistajaan keskimäärin useamman kerran päivässä ja valtaosa viestinnästä hoidettiin Skypen välityksellä. Kasvokkain tuoteomistajan kanssa tavattiin useimmiten tilanteissa, joissa käsiteltävä asia oli vielä suhteellisen jäsentymätöntä ”höttöä”.

Yrityksen 1 sovelluskehittäjän projektissa kehitettävää järjestelmää työstettiin kahden viikon sprinteissä ja asiakkaan kanssa tavattiin kasvokkain sprinttien välissä seuraavan kehitysjakson töiden suunnittelun parissa. Viestintään käytettiin pääasiassa sähköpostia ja puhelimeen turvauduttiin ainoastaan kiireellisissä tapauksissa, joissa tarvittiin asiakkaalta tietoja työn saamiseksi eteenpäin. Haastateltu sovelluskehittäjä arvioi, että asiakas osallistui projektiin sopivan aktiivisesti lopputuloksen kannalta. Hänen arvionsa mukaan yhteistyö oli toimivaa:

”Asiakkaan kanssa oltiin suht tiiviissä yhteistyössä että hekin auttoivat sitten suunnittelussa ja jonkin verran jopa kehitystyössä.”

– *Sovelluskehittäjä*

Yrityksestä 2 haastateltu vanhempi suunnittelija kertoi edustavansa projektissa asiakasta kehitysjaksojen aikana. Haastateltava toimi kehittäjien ensisijaisena tietolähteenä ja otti yhteyttä asiakkaaseen niissä tilanteissa, kun ei itse pystynyt tarjoamaan vastauksia. Tällä tavoin asiakkaan ei itse tarvinnut olla koko ajan vastaamassa kysymyksiin ja seuraamassa kehitystyön edistymistä.

Ensimmäisen ohjelmistotoimittajan kanssa ei ollut käytössä yhteisiä Skype-kanavia, joissa myös asiakas olisi ollut läsnä, mikä osaltaan vaikeutti viestintää. Uuden toimittajan kanssa nämä viestintäkanavat kuitenkin ovat olemassa. Vaikka viestintä pääasiassa tapahtuikin haastatellun välityksellä, asiakas kuitenkin osallistui kehitysjaksojen välisiin arviointitapaamisiin. Niinpä ohjelmistotoimittajan suora kosketuspinta asiakkaaseen oli lähinnä sprinttipalaverieissa.

Yrityksessä 2 työskentelevän haastatellun tuoteomistajan projektissa asiakas osallistui kohtalaisen aktiivisesti projektiin, mutta aikatauluhaasteiden vuoksi jokapäiväinen viestintä oli annettu haastatellun tuoteomistajan vastuulle. Haastateltu mainitsi, ettei asiakkaalla esimerkiksi ollut omien sanojensa mukaan aikaa seurata keskusteluja pikaviestimissä. Kiireellisissä tapauksissa asiakkaan sai parhaiten kiinni puhelimella, ja sähköpostilla vastauksen sai parissa kolmessa päivässä. Yrityksen 2 haastateltu tuoteomistaja kertoi suosivansa kasvokkain tapaamisia:

”Niin paljon kuin sitä kasvotusten puhumista saa vaan aikaseksi niin paljon sitä pitää myös ottaa koska se on kaikista paras. Jää pois se tulkinnanvaraisuus kun ihminen pystyy näkemään toisen naaman.”

– *Tuoteomistaja*

Tässä projektissa kasvokkain tapaamiset onnistuivat keskimäärin kahden viikon välein. Haastateltu tuoteomistaja koki, että asiakkaan kiireiden vuoksi osa vähemmän kriittisistä kysymyksistä oli hyvä säästää näihin kasvokkain tapaamisiin.

Yrityksestä 3 haastateltu palvelusuunnittelija kertoi, että asiakkaan kanssa viestittiin päivittäin lyhyillä videoneuvotteluilla, joissa käsiteltiin projektin etenemistä ja vastaan tulleita ongelmia. Kehittäjillä oli myös mahdollisuus kysyä asiakkaalta lisätietoja tarvitsemiinsa asioihin näissä tapaamisissa. Tästä huolimatta haastateltu palvelusuunnittelija arvioi, ettei asiakas pystynyt osallistumaan tarpeeksi aktiivisesti projektiin. Alussa asiakkaan puolelta oli avainhenkilönä vain markkinointipäällikkö, mikä vaikeutti teknisistä asioista keskustelua. Kun projektiin saatiin asiakkaalta myös tekninen päällikkö mukaan, projektissa saatiin parempi käsitys asiakkaan omista järjestelmistä.

Yrityksen 3 UX-suunnittelijan projektissa asiakkaalla ei ollut nimettynä varsinaista tuoteomistajaa, sillä asiakkaan yrityskulttuuriin kuului eräänlainen ”kollektiivinen omistajuus”. Projektiin oli kuitenkin nimetty asiakkaalta avainhenkilö, johon ensisijaisesti oltiin projektiin liittyvissä asioissa yhteydessä. Asiakkaan kanssa tavattiin fyysisesti kehitysvaiheiden alussa ja lopussa sekä tilanteissa, joissa suunnitelmia jouduttiin kesken vaiheen selvästi muuttamaan. Haastateltava kertoi muutoin olevansa noin kerran viikossa yhteydessä asiakkaaseen ”tavalla tai toisella”:

”Tyypillisesti joko [sähköpostilla] tai saattaa olla ihan vaan sellanen [epämuodollinen puhelu], missä kerron, missä mennään, onko isoja poikkeamia [suunnitelmaan] ja niin pois päin. Eli tässä ei oikeastaan [ole] tarvetta näissä vaiheiden sisällä sieltä asiakkaankaan kannalta seurata päivätasolla, se ei ole heille oleellista. Lähinnä se, jos tulee poikkeamia siihen isoon [suunnitelmaan].”

– UX-suunnittelija

Jokaisessa projektissa moniselitteisiä ja vaikeasti hahmotettavia asioita käsiteltiin mieluiten kasvokkain. Nopeaa reagointia vaativissa asioissa haastatellut turvautuivat puhelimeen, mutta muutoin asiakkaan suuntaan käytettiin viestintämedioita, jotka eivät vaatineet asiakkaalta aktiivista läsnäoloa viestintävälineen parissa. Palvelusuunnittelija yrityksestä 3 mainitsi, että projektinhallinnan ja siihen liittyvän viestinnän keskittäminen yhteen työkaluun helpotti sekä kehitystiimin omaa työtä että asiakkaan kykyä seurata projektin edistymistä.

5.5. Asiakasviestinnästä tehtyjä havainnot

Viestinnällä on selvästikin iso merkitys projektien onnistumiseen. Sitä käytetään niin kehitystehtävien määrittämiseen kuin myös asiakkaan odotusten kontrollointiin. Viestintävälineiden valinta vaikuttaa myös projektiin osallistuvien kykyyn tunnistaa sosiaalisia rakenteita asiakkaan organisaatiossa ja rakentaa yhteistä tietopohjaa epävirallisella viestinnällä.

Vaikka ketterissä projekteissa pyritään tuottamaan toimivia ohjelmistotuotteita nopeasti, kannattaa yrityksen 3 projektipäällikön mukaan käyttää alussa aikaa todellisten tarpeiden selvittämiseen. Hänen mukaansa tällä tavoin voidaan keskittyä tekemään juuri sitä, mitä asiakas tarvitsee, eikä käyttää aikaa ylimääräisiin töihin:

”Alkuvaiheessa ei pidä todellakaan rynnätä tekemään nopeasti, kun näyttää siltä, että asiakas tietää mitä haluaa vaan se pitää syvästi myöskin tietää siellä kehitysporukan puolella, mitä asiakas haluaa ja se on sen kehitysporukan vastuulla ymmärtää se [tarve], että se ei kolminkertaistu se toimitusaikataulu suhteessa alkuperäiseen tavoitteeseen”

– Projektipäällikkö

Myös UX-suunnittelija yrityksestä 3 huomautti, että asiakkaalta aluksi saatu tieto asiakkaan tarpeista voi olla puutteellista. On täysin mahdollista, että projektiin osallistuvilla avainhenkilöillä ei ole kattavaa kuvaa loppukäyttäjien työtavoista ja tarpeista:

”He eivät itsekään sisäisesti keskustele riittävästi. Kun siellä on useamman ihmisen kanssa tekemisissä niin saakin vähän erilaista näkökulmaa.”

– UX-suunnittelija

Sovelluskehittäjä yrityksestä 1 muistutti, että pelkkä asiakkaan tarpeisiin vastaaminen ei yksinään riitä tuottamaan asiakastytyväisyyttä. On tärkeää luoda asiakkaalle realistinen kuva siitä, mitä kehitystiimi pystyy tietyn ajan sisällä toimittamaan. Tämä edellyttää varovaisuutta aika-arvioissa ja niiden viestinnässä:

”Pitäisi aina luvata vähän vähemmän ja toimittaa vähän enemmän. Asiakas yllättyy silloin positiivisesti.”

– Sovelluskehittäjä

Yksi onnistuneen viestinnän avaimista on tunnistaa valtarakenteita asiakkaan organisaatiossa. Tuoteomistajan roolissa toiminut yrityksen 2 haastateltu tähdensi, että

on tärkeää osata tulkita, millaisia ristiriitoja asiakasorganisaation sisällä on ja tunnistaa mitkä ovat herkkiä aiheita. Myös valitun viestintämedian rajoitteet on otettava huomioon väärinkäsitysten välttämiseksi:

”Esimerkiksi mä lähetin erään sähköpostin missä mä kommentoin erään asian ja laitoin hymiön sinne perään. Aiheutti ihan käsittämättömän [myrskyn]. Siinä ajateltiin, että mä suoranaisesti [vinoilen] sille asiakkaalle.”

– Tuoteomistaja

Hyväksi havaitut menetelmät eivät aina jää pelkästään kehitystiimin työkalupakkiin. Asiakkaan osallistuva rooli ketterissä menetelmissä tarjoaa mahdollisuuden molemmiin suuntaiseen vuorovaikutukseen ja asiakas voi oppia kehitystiimin toimintatavoista samaan tapaan kuin kehitystiimi voi oppia asiakkaan. Palvelusuunnittelijan projektissa kävi juuri näin, kun projektissa hyväksi havaitut viestintätavat päättyivät lopulta myös asiakkaan yleisemmiksi käytännöiksi. Scrum-menetelmässä joka päivä pidetään hyvin lyhyt daily scrum -hetki, jolloin jokainen kertoo nopeasti, mitä edellisenä päivänä sai aikaan ja mitä aikoo tämän päivän aikana tehdä. Tällainen lyhyt tilannekatsaus auttaa asiakasviestinnässä, mutta myös sisäisissä projekteissa:

”Nyt sit rantautunut heidän talon sisällekkin tää daily-menetelmä, että he siellä sisäisestikin [tarkastelevat], mitä kaikki tekevät talossa. [Se] parantaa työhyvinvointia ja kommunikointia siinä mielessä, että siellä ei kaikki käy vuorotellen pitkin päivää kysymässä, että mitä sä teet. Keskitetysti saadaan joka päivä IT-väki keskustelemaan aiheista ja koko muu päivä on sitten rauhoitettu muulle työnteolle.”

– Palvelusuunnittelija

On selvää, että mikäli kaikki aika käytetään keskusteluun, ei aikaa jää mihinkään muuhun. Lyhyet tilannekatsaukset auttavat kuitenkin kehitystiimejä sekä asiakkaita pysymään paremmin selvillä valitusta suunnasta ja edistymisen tahdista. Vaikuttaa myös siltä, että käyttämällä projektin alussa enemmän aikaa myös epäviralliseen viestintään, voidaan helpottaa yhteistyötä asiakkaan ja kehitystiimin välillä luomalla yhteistä tietopohjaa asiakkaan toimintatavoista ja sisäisistä sosiaalisista rakenteista.

6. Pohdinta

Tämän tutkielman alussa esitin kysymyksen: *miten ketterien menetelmien projekteissa ylläpidetään yhteistä tietopohjaa asiakkaan ja kehitystiimin välillä*. Haastatteluilla pyrin kartoittamaan erilaisia asiakkaan ja kehitystiimin välisiä viestintätilanteita sekä eri tilanteissa käytettyjä viestintävälineitä. Tässä luvussa vertailen havaintojani luvussa 2 esiteltyihin taustateorioihin ja muutamaan ohjelmistoprojektien viestintää käsittelevään tutkimukseen.

6.1. Projektien viestintätilanteet ja -välineet

Haastatteluissa korostui viestintäkäytäntöjen tapauskohtaisuus. Projektimenetelmiä sovellettiin ja räätälöitiin tarpeiden mukaan tapauskohtaisesti eikä yrityksissä ollut määritelty valmista joukkoa sähköisiä työkaluja projektien käyttöön. Haastateltujen esittelemät projektit olivat hyvin erilaisia, mutta kaikissa niissä toistui samantyyppisiä viestintätilanteita.

Taulukossa 14 kokoaan yhteen projekteissa esiintyneitä viestintätarpeita ja niihin käytettyjä viestintämedioita. Siitä on nähtävissä, että suunnitteluun liittyvissä tehtävissä hyödynnettiin pääasiassa matalan synkronisaation viestintämedioita. Tämä käy yhteen median synkronisaation teorian kanssa, jonka mukaan matalan synkronisaatiotason viestintämediat sopivat tilanteisiin, joissa viestin vastaanottaja tarvitsee aikaa viestin sisältämän tiedon tarkasteluun ja sisäistämiseen [Fernando *et al.* 2011].

	Vaati- musten ja tarpeiden määrittely	Kehitys- tehtävien määrittely	Muutok- sista viestintä	Lisä- tietojen pyytä- minen	Edisty- misen seuranta	Tiimi- hengen luominen
<i>Kasvokkain</i>	×	×	×	×	×	×
<i>Skype / Lync</i>			×	×	×	
<i>Puhelin</i>			×	×	×	×
<i>Slack / Flowdock</i>				×		×
<i>Sähköposti</i>	×			×	×	
<i>Trello / Jira</i>	×	×			×	
<i>Confluence</i>	×					
<i>Google Drive / SharePoint</i>	×	×				

Taulukko 14: Viestintätarpeet ja niissä käytetyt viestintämediat

Korkean synkronisaation viestintämedioita käytettiin tilanteissa, joissa tarkennettiin olemassa olevaa tietoa tai viestittiin poikkeamista suunnitelmaan. Median synkronisaation teoriassa mainittiin, että korkean synkronisaatiotason viestintämediat sopivat tilanteisiin, joissa halutaan hakea yhteisymmärrystä jo olemassa olevasta informaatiosta [Fernando *et al.* 2011]. Jo olemassa oleviin vaatimusmäärittelyihin liittyvät keskustelut ovat luonteeltaan juuri tällaisia yhteisymmärrystä hakevia tilanteita.

Yhteisen tietopohjan luominen tapahtuu juuri näissä yhteisymmärrystä hakevissa tilanteissa, joissa keskustelukumppaneiden on mahdollista reagoida nopeasti vastapuolen kommentteihin, pyytää tarkennuksia epäselviin asioihin ja korjata väärinymmärryksiä. Keskustelut tapahtuivat yleensä kehitysjaksojen työtehtävien suunnitteluvaiheessa ja tavoitteena oli muodostaa vaatimusmäärittelyistä kehitystehtäviä kehitysjaksoon. Toisin sanoen olemassa olevasta informaatiosta jalostettiin yhteisymmärrys seuraavaksi kehitettävistä ominaisuuksista.

Kohdassa 5.5 nostin esille tuoteomistajan kommentin siitä, miten asiakas ymmärsi väärin sähköpostin sävyn. Kommentti paljastaa, miten tietyt viestintämediat ovat heikompia sosiaalisten vihjeiden, kuten huumorin, välittämisessä.

Taulukosta 14 paljastuu myös, että projekteissa hoidettiin kasvokkain tavatessa kaikkia viestintätarpeita vähintäänkin jollain tasolla. On kuitenkin huomioitava, että kasvokkain tavattaessa ei luotettu pelkästään suulliseen viestintään vaan esimerkiksi vaatimuksia ja kehitystehtäviä määriteltäessä asioita kirjattiin muistiin esimerkiksi Trelloon tai Jiraan. Kuten Dennis ja Valacich [1999] totesivat, valitun viestintämedian ilmaisuvoimaisuus riippuu usein siitä, millaisilla tavoilla viestintämediaa käytetään.

Kasvokkain tapahtuvan viestinnän yhteydessä ilmaisuvoimaisuutta voi lisätä käyttämällä erilaisia aputyökaluja, kuten kaavioiden piirtämistä taululle tai paperille. Haastateltujen projektien yhteydessä apukeinona oli käsiteltävien asioiden kirjaaminen projektinhallintajärjestelmään tai wikiin.

Vaatimusmäärittelyyn ja tarpeiden kartoitukseen liittyviä keskusteluja käytiin projekteissa läpi ensisijaisesti kasvokkaisissa tapaamisissa. Yrityksessä 3 työskentelevä projektipäällikkö mainitsi suosivansa kasvokkain tapaamisia tilanteissa, joissa käsiteltävä asia oli hyvin jäsentymätöntä. Myös yrityksen 2 haastateltu tuoteomistaja mainitsi tulkinnanvaraisuuden poistuvan viestinnästä kasvokkain tavattaessa. Sen sijaan näiden keskustelujen tuloksia kirjattiin matalan synkronisaation viestintämedioihin, kuten projektinhallintatyökalut Trello ja Jira.

Vaatimuksiin ja tarpeisiin liittyvissä kysymyksissä puhelimeen turvauduttiin projekteissa silloin, kun vastauksia tarvittiin nopeasti kehitystyön jatkamiseksi. Mikäli tiedonpuute ei aiheuttanut estettä töiden jatkumiselle, käytettiin projektin sähköisistä työkaluista riippuen joko pikaviestimiä tai sähköpostia. Tällä tavoin asiakkaan oma työpäivä keskeytyi odottamatta vain silloin, kun siihen todella oli tarve ja muissa tilanteissa asiakas pystyi vastaamaan kehitystiimin kysymyksiin hänelle itselle parhaiten sopivassa välissä. Yleisesti puhelimeen turvauduttiin projekteissa ensisijaisesti tilanteissa, joissa asiakkaalta tarvittiin nopeaa reagointia. Käsiteltävä aihe ei välttämättä ollut näissä tilanteissa korkean synkronisaatiotason viestintämedialle sopivia, mutta ilman pikaista reagointia työt olisivat seisoneet.

Modi ja muut [2013] toteavat tapaustutkimuksessaan, että epävirallisella viestinnällä hajautetuissa ketterissä projekteissa on suuri merkitys yhteisen tietopohjan muodostumisessa. Haastatteluissa esitellyissä projekteissa epävirallista viestintää hoidettiin kasvokkain ja esimerkiksi puhelimella projektin edistyksen raportoinnin yhteydessä. Tuoteomistajan projektissa jopa järjestettiin koko projektitiimille epävirallisia tapaamisia, kuten keilausilta, jolloin ihmiset pystyivät tutustumaan toisiinsa ja juttelemaan epävirallisesti projektiin liittyvistä asioista.

Tällaisten epävirallisten tapaamisten taustalla on tiimin jäsenten yhteistyön sujuvoittaminen. Fernando ja muut [2011] viittasivat tällaiseen sosiaaliseen kanssakäymiseen median synkronisaation teorian kolmannessa ulottuvuudessa, johon sisältyi tiimin hyvinvointiin ja jäsenten tukeen liittyvät tehtävät ja toiminnot. Kuten tuoteomistaja totesi, pitempään yhdessä töitä tehnyt kehitystiimi pystyi siirtymään Kanban-menetelmään, koska jäsenet tunsivat toisensa ja toistensa työskentelytavat.

Median synkronisaation teoria suositteli vaihtelevaan erilaisten viestintämedioitten välillä viestintätarpeiden mukaan [Fernando *et al.* 2011].

Haastateltujen projekteissa turvauduttiinkin eri viestintämedioihin tarpeesta riippuen. Avoimen lähdekoodin projekteja tutkittaessa havaittiin kuitenkin, että sähköpostilistat yhdistettynä ajoittaisiin kasvokkain tapaamisiin riittivät hyvin monenlaisiin viestinnällisiin tarpeisiin [Gutwin *et al.* 2004].

Viestinnän rajoittaminen vain muutamaankin välineeseen helpottaa viestihistorian ylläpitämistä, jolloin on helpompaa löytää aiemmin keskustelun kohteena olevat asiat. Haittapuolena viestintävälineiden rajoittamisessa on ainakin median synkronisaation teorian pohjalta kasvanut väärinymmärryksen riski tai viestinnän tehokkuuden heikkeneminen. Lee ja muut [2006] huomauttavat että varsinkin globaalisti hajautetuissa ketterissä projekteissa on tärkeää tuottaa kattavaa dokumentaatiota koodin lisäksi, koska ketterien projektien suosima epämuodollinen viestintä saattaa olla tehottomampaa kulttuurierojen ja kielierojen vuoksi. Tämä täytyy ottaa huomioon myös projekteissa, joissa on kansainvälisiä asiakkaita.

Projekteissa valitut viestintämediat ja niiden käyttö vastasi suurimmaksi osaksi median synkronisaation teorian esittämiä oletuksia. Muutamassa haastattelussa nousi esille, että asiakkailla meni hetki omaksua uusien viestintävälineiden käyttö ja että viestintävälineitä harkittaessa asiakkaiden aiempi kokemus sovelluksista vaikutti valintoihin. Myös Fernando ja muut [2011] huomasivat saman ilmiön.

6.2. Projekteissa ilmenneet aikatauluongelmat

Haastateltujen esittelemissä projekteissa ilmeni aikatauluongelmia tilanteissa, joissa kehitystiimi joutui jatkamaan projektia toisen tiimin epäonnistuttua kehitettävän ohjelmistotuotteen toimittamisessa. Asiakkaan näkökulmasta projekti oli näissä tilanteissa jo jatkunut pidemmän aikaa, eikä kehitystyötä oltu valmiita aloittamaan alusta. Uuden tiimin oletettiin jatkavan edellisen työtä, joten alkuperäinen aikataulu määräaikaan siirtyi uudelle tiimille.

Haastatteluissa nousi esille kaksi täysin vastakkaista viestintätapaa vastauksena edellisen ohjelmistotoimittajan epäonnistumisen tuottamiin aikatauluhaasteisiin. Vanhemman suunnittelijan projektissa koettiin, ettei kiireen vuoksi aikaa voitu käyttää asiakkaan ja kehitystiimin väliseen viestintään niin paljon kuin kehittäjät olisivat toivoneet. Projektipäällikön esittelemässä projektissa asiakas taas ryhtyi vaatimaan tiiviimpää viestintää erityisesti projektin edistymiseen liittyen, josta ei kuitenkaan projektipäällikön arvion mukaan ollut mitään todellista hyötyä projektin onnistumisen kannalta.

Toinen haastatteluista esiin noussut aikatauluja venyttävä tekijä oli liian ylimalkaiset määrittelyt. Yrityksen 3 UX-suunnittelijan esittelemässä projektissa yhdestä sivuhuomautuksesta käyttöliittymäkuvassa saattoi paisua useamman uuden näkymän kokonaisuus. Yksinkertaisiksi oletetuista asioista paljastui vasta toteutusvaiheessa niiden todellinen laajuus. Niinpä haastateltu kehotti välttämään kehitystyöhön ryntäämistä ennen kuin kehittäjille varmasti on selvinnyt, mitä asiakas todella tarvitsee. Tässä mielessä yksinkertaisten toiminnallisten prototyyppien käyttäminen voisi edistää keskustelua paremmin kuin pelkät käyttöliittymäkuvat, auttaa välttämään ominaisuuksien paisumista ja keskittymään olennaisimpaan toiminnallisuuteen.

Palvelusuunnittelijan projektissa työtä hidasti alussa se, että asiakkaalta osallistui projektiin vain markkinointipäällikkö, jolla ei ollut ymmärrystä asiakkaan omien järjestelmien teknisistä ratkaisuista. Työtahti parani, kun asiakas sai irrotettua projektin käyttöön teknisen asiantuntijan, joka pystyi vastaamaan kehittäjien kysymyksiin. Myös UX-suunnittelija totesi, ettei yhden avainhenkilön näkemys tarpeesta välttämättä edusta koko organisaation näkemystä.

Asiakkaalta projektiin osallistuvan avainhenkilön näkemys kehitettävän ohjelmiston tarpeista ei välttämättä vastaa koko organisaation näkemystä. Mikäli asiakkaan organisaation sisällä ei keskustella riittävästi, ei asiakkaan avainhenkilö välttämättä ole edes tietoinen kaikista tarpeista. Varsinkin tilanteissa, joissa loppukäyttäjää ei ole asiakas, tarpeiden selvittäminen vaatii erityistä huomiota.

6.3. Pohdintaa menetelmästä

Valintani tutkia viestintämenetelmiä ketterissä projekteissa tapaustutkimuksen keinoin ei ole ainutlaatuinen. Modi ja muut [2013] tutkivat tapaustutkimuksella yhteisen tietopohjan muodostamista globaalisti hajautetussa ketterässä kehityksessä. Isaacs ja Tang [1993] tutkivat tapaustutkimuksen avulla videopuheluiden hyödyntämistä maantieteellisesti hajautetussa yhteistyössä ja havaitsivat tällä tavalla rikkaamman viestintämedian sujuvoittavan viestintää verrattuna pelkkään äänipuheluun. He kuitenkin huomauttivat hyödyn olevan täysin riippuvainen videopuhelun äänen viiveettömyydestä. Yagüe ja muut [2016] ovat tehneet eksploratiivisen tutkimuksen tapaustutkimuksella viestinnästä globaalisti hajautetussa ketterässä kehityksessä. Toisin sanoen, tapaustutkimus on hyväksytty menetelmä tällaiseen tutkimukseen.

Haastattelut eivät olleet ainoa tapa, jolla aineistoa olisi voinut kerätä tapaustutkimusta varten. Olisin voinut kerätä tietoa projekteista laatimalla kyselyn, jonka olisin lähettänyt moneen yritykseen. Tämä olisi saattanut tarjota suuremman määrän tapauksia tutkittavaksi, mutta näkökulma olisi rajoittunut ennalta määriteltyjen asioiden tarkasteluun. Haastatteluissa pystyin tarttumaan myös itse haastattelutilanteessa esiin nousseisiin uusiin aiheisiin.

Toinen vaihtoehto olisi voinut olla mennä tarkkailemaan yhteen yritykseen asiakasviestintää. Tämä olisi kuitenkin vienyt paljon aikaa eikä useamman tapauksen tarkastelu olisi ollut mahdollista. Myös omien aikataulujeni sovittaminen tällaisen tarkkailun mahdollistamiseksi olisi ollut vaikeaa. Ongelmaksi olisi saattanut muodostua myös tietosuojasyyt, mikäli olisin tehnyt tarkkailuni yrityksen tiloissa. Myöskään lokitietojen tarkastelu ei olisi tietosuojasyistä ollut realistinen vaihtoehto. Haastattelujen avulla haastateltavat pystyivät itse rajoittamaan, mitä kertoivat ja olemaan paljastamatta arkaluontoisia asioita.

Järvinen ja Järvinen [1996, 102-103] muistuttavat, että haastateltavat saattavat kokea haastattelukysymykset tungetteleviksi tai epäillä tutkimusta yrityksen johdon teettämäksi valvontamenettelyksi, jolloin he saattavat jättää kertomatta osan asioista. Itse en havainnut haastattelutilanteissa tällaista varautuneisuutta, mutta sen mahdollisuus on silti olemassa. Selkeämpi ongelma oli kuitenkin se, että haastatteluja tehdessäni en vielä pystynyt täysin arvioimaan, mitkä seikat tulisivat olemaan tärkeitä ja mistä olisi pitänyt kysellä tarkemmin. Yksi ratkaisu olisi voinut olla lyhyt jatkohaastattelu sen jälkeen, kun varsinaiset haastattelut oli tehty ja alustavasti käsitelty.

Kokemattomuuteni haastattelijana vaikutti varmasti myös keräämäni haastatteluaineiston laatuun. Onnekseni haastateltavat olivat hyvin avoimia ja sain usein pitkiäkin vastauksia suhteellisen yksinkertaisiin kysymyksiin. Strukturoitu haastattelu olisi ollut avointa haastattelua helpompi tapa kerätä tietoa, mutta saamani aineisto olisi ollut hyvin paljon rajoittuneempi. Koin muistiinpanojen kirjaamisen haastattelutilanteessa vaikeaksi, joten päädyin hyvin pian luottamaan pelkkään haastattelun äänittämiseen. Karkea haastattelun aikana koostettu lista esiin nousseista kiinnostavista asioista olisi helpottanut aineiston käsittelyvaiheessa.

Haastateltavien rekrytoiminen ei ollut ongelmatonta. Siitä huolimatta, että haastateltuja rekrytoitiin henkilökohtaisesti kohdistetuilla sähköpostiviesteillä, lopulta vain muutama vastasi kutsuun. On myös huomioitava, että sain vain yhden ihmisen näkökulman jokaisesta projektista, joten haastattelut eivät välttämättä tarjonneet täydellistä kuvaa asiakasviestinnästä näissä projekteissa.

Useampi haastattelu olisi varmasti tarjonnut enemmän yhteisiä asioita eri projektien välillä, mutta poikkeuksiakin olisi todennäköisesti myös tullut enemmän vastaan. Useampi haastattelu samaan projektiin osallistuneilta ihmisiltä olisi antanut kattavamman kuvan projektin viestinnästä. Olisi ollut myös mielenkiintoista saada haastatteluja pitkin samaa projektia, jolloin olisi selvinnyt, muuttuiko ihmisten näkemys viestinnästä projektin aikana.

Asiakkaan näkökulma jäi tässä tutkielmassa melko lailla tarkastelun ulkopuolelle vaikka osa haastatelluista toimikin asiakkaan edustajina. Viestinnän onnistumisen arvioinnissa olisi auttanut asiakkaan näkemys siitä, miten projektit heidän mielestään onnistuivat.

7. Yhteenveto

Tässä tutkielmassa olen tarkastellut usean tapauksen tapaustutkimuksella, miten ketterien menetelmien projekteissa ylläpidetään yhteistä tietopohjaa asiakkaan ja kehitystiimin välillä. Ketterät menetelmät painottavat tehokasta viestintää ja yhteinen tietopohja puolestaan on viestintää sujuvoittava vuorovaikutuksen prosessi. Niinpä olikin luontevaa lähteä tarkastelemaan viestintää yhteisen tietopohjan näkökulmasta.

Yhteisen tietopohjan muodostumiseen vaikuttaa oleellisesti käytetyn viestintämedian ilmaisuvoimaisuus. Niinpä eri viestintämedioiden arviointiin valikoitui median synkronisaation teoria, joka on vastaus vanhemman median ilmaisuvoimaisuuden teorian kohtaamaan kritiikkiin. Esittelemäni kolme yleisintä ketterän ohjelmistokehityksen menetelmää esiintyivät haastateltujen projekteissa muokattuina versioina. Yleisesti voidaan todeta, ettei haastateltujen projekteissa oltu menetelmäuskollisia vaan menetelmistä valikoitiin parhaiten tarpeita vastaavat palat käyttöön.

Haastatteluissa selvisi, että viestintätavat vaihtelevat projekteittain jopa saman yrityksen sisällä. Muun muassa epäonnistunut viestintä oli kahdessa projektissa johtanut ohjelmistotoimittajan vaihtamiseen ja vaihdoksesta aiheutuneet aikatauluongelmat siirtyivät uudelle kehitystiimille. Huolimatta laajasta valikoimasta erilaisia sähköisiä viestintävälineitä, projekteissa suosittiin kasvokkaisia tapaamisia aina, kun se vain oli mahdollista.

Valitut viestintämediat tukivat pääasiassa median synkronisaation teorian oletuksia: yhteisymmärrystä hakevissa viestintätarpeissa hyödynnettiin korkean synkronisaation viestintämedioita ja informaatiota ilmaisevia asioita viestittiin matalan synkronisaation medioilla. Yhteistä tietopohjaa tukevaa ja luovaa viestintää hoidettiin kasvokkain sekä korkean synkronisaation viestintämedioilla kuten puhelimella ja videopuheluilla. Lisätietoja epäselviin tilanteisiin saatettiin pyytää kuitenkin myös matalan synkronisaation viestintämedioilla silloin, kun asiakkaan koettiin olevan kiireinen ja tarvittua lisätietoa ei pidetty kriittisenä. Viestintämedioita hyödynnettiin siis melko joustavasti.

Projektin sujuvan etenemisen kannalta oli oleellista, että kehittäjillä oli mahdollisuus olla yhteydessä asiakkaan edustajaan, jolla oli tiedossa oleellinen tieto. Ongelmia ilmeni esimerkiksi, kun asiakkaalta projektiin osallistui vain henkilö, jolla ei ollut tietoa yrityksen teknisistä ratkaisuista. Esiin nousi myös näkemys siitä, että on tärkeää saada asiakkaan puolelta useamman ihmisen näkemys tarpeista.

Viestintä oli keskeisessä asemassa, kun haastateltavilta kysyttiin hyväksi havaittuja menetelmiä ketterissä projekteissa. Asiakkaan odotukset kannattaa pitää kurissa olemalla hyvin varovainen lupaillessaan, mitä kaikkea kehitystiimi pystyy tietyn ajan sisällä toimittamaan. Kehitystyöhön ei myöskään kannata säännätä ennen kuin kaikille on täysin selvää, mitä ollaan tekemässä. Toiminnallisista prototyypeistä voisi tässä keskustelussa olla hyötyä. Viestinnässä on syytä huomioida, ketkä ovat viestin vastaanottajia ja miten he mahdollisesti viestiin reagoivat. Onkin esitetty, että projektipäälliköillä on oltava hyvät sosiaaliset taidot tehokkaan viestinnän ja yhteistyön varmistamiseksi [Alzoubi *et al.* 2014].

Tulevaisuudessa olisi mielenkiintoista nähdä kvantitatiivinen tutkimus, jossa tarkasteltaisiin millaisia työkaluja yleisesti median synkronisaation teorian tunnistamisessa viestintätarpeissa käytetään. Asiakkaan näkökulman sisällyttäminen voisi tarjota kattavamman kuvan kuin tässä tutkielmassa oli mahdollista saada.

Myös pitkittäistutkimus yhden projektin viestintään voisi paljastaa, muuttuvatko viestintätavat ja valittujen viestintämedioiden käyttötavat projektin aikana. Tässä tutkielmassa ilmeni, että kehitystiimin on mahdollista siirtyä työtavasta toiseen, kun tiimin jäsenet oppivat tuntemaan paremmin toisiaan. On perusteltua olettaa, että myös viestintätavat saattavat kehittyä tällä tavalla.

Lähdeluettelo

- Nik Nailah Binti Abdullah, Shinichi Honiden, Helen Sharp, Bashar Nuseibeh, and David Notkin. 2011. Communication patterns of agile requirements engineering. In: *Proceedings of the 1st Workshop on Agile Requirements Engineering (AREW '11)*, Article 1, 4 pages. DOI: 10.1145/2068783.2068784
- Yehia Ibrahim Alzoubi, Asif Qumer Gill and Ahmed Al-Ani. Empirical studies of geographically distributed agile development communication challenges: A systematic review. *Information & Management* 53, 1, (January 2016), 22-37, DOI: 10.1016/j.im.2015.08.003
- Ritu Arora and Sanjay Goel. 2012. Collaboration in software development: a spotlight. In: *Proceedings of the CUBE International Information Technology Conference (CUBE '12)*, 391-396. DOI: 10.1145/2381716.2381789
- Zornitza Bakalova and Maya Daneva. 2011. A comparative case study on clients participation in a 'traditional' and in an Agile software company. In: *Proc. of the 12th International Conference on Product Focused Software Development and Process Improvement (Profes '11)*, 74-80. DOI: 10.1145/2181101.2181118
- Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland and Dave Thomas. 2001. Manifesto for agile software development. <http://www.agilemanifesto.org/>
- Hugh Beyer. 2010. User-centered agile methods. John M. Carroll (eds.), *Synthesis lectures on human-centered informatics*, lecture 10, Morgan & Claypool Publishers, 71 pages. DOI: 10.2200/S00286ED1V01Y201002HCI010
- Pernille Bjørn, Morten Esbensen, Rasmus Eskild Jensen and Stina Matthiesen. 2014. Does distance still matter? Revisiting the CSCW fundamentals on distributed collaboration. *ACM Trans. Comput.-Hum. Interact.* 21, 5, Article 27 (November 2014), 26 pages. DOI: 10.1145/2670534

- Thomas Chau and Frank Maurer. 2004. Knowledge sharing in agile software teams. Wolfgang Lenski (eds.), *Logic versus Approximation, Lecture Notes in Computer Science* 3075, 173-183. DOI: 10.1007/978-3-540-25967-1_12
- James O. Coplien and Gertrud Bjørnvig. 2010. *Lean Architecture: for Agile Software Development*. Wiley.
- Alan R. Dennis and Joe S. Valacich. 1999. Rethinking media richness: towards a theory of media synchronicity. In: *Proc. of the 32nd Annual Hawaii International Conference on Systems Sciences (HICSS-32)*, 10 pages. DOI: 10.1109/HICSS.1999.772701
- Jari Eskola ja Jaana Vastamäki. 2010. Teemahaastattelu: opit ja opetukset. Teoksessa: Juhani Aaltola ja Raine Valli (toim.), *Ikkunoita tutkimusmetodeihin I*. PS-kustannus, 26-44.
- Biyagamage Agra Junius Fernando, Tracy Hall and Anthony Fitzpatrick. 2011. The impact of media selection on stakeholder communication in agile global software development: a preliminary industrial case study. In: *Proc. of the 49th SIGMIS annual conference on Computer personnel research (SIGMIS-CPR 2011)*, 131-139. DOI: 10.1145/1982143.1982177
- Brian Fitzgerald, Mariusz Musiał and Klaas-Jan Stol. 2014. Evidence-based decision making in lean software project management. In: *Companion Proceedings of the 36th International Conference on Software Engineering (ICSE Companion 2014)*, 93-102. DOI: 10.1145/2591062.2591190
- Angela Graveline, Cheryl Geisler and Michael Danchak. 2000. Teaming together apart: emergent patterns of media use in collaboration at a distance. In: *Proc. of IEEE professional communication society international professional communication conference and Proceedings of the 18th annual ACM international conference on Computer documentation: technology & teamwork (IPCC/SIGDOC 2000)*, 381-393.
- Carl Gutwin, Reagan Penner and Kevin Schneider. 2004. Group awareness in distributed software development. In: *Proc. of the 2004 ACM conference on Computer supported cooperative work (CSCW 2004)*, 72-81. DOI: 10.1145/1031607.1031621
- James D. Herbsleb, Audris Mockus, Thomas A. Finholt and Rebecca E. Grinter. 2000. Distance, dependencies, and delay in a global collaboration. In: *Proc. of the 2000 ACM conference on Computer supported cooperative work (CSCW '00)*, 319-328. DOI: 10.1145/358916.359003

- Elke Hochmüller. 2011. The requirements engineer as a liaison officer in agile software development. In: *Proc. of the 1st Workshop on Agile Requirements Engineering (AREW '11)*, Article 2 , 4 pages. DOI: 10.1145/2068783.2068785
- Helena Holmström Olsson and Jan Bosch. 2015. Towards continuous validation of customer value. In: *Scientific Workshop Proceedings of the XP2015 (XP '15 workshops)*, Article 3, 4 pages. DOI: 10.1145/2764979.2764982
- Markus Hummel, Christoph Rosenkranz and Roland Holten. 2013. The role of communication in agile systems development. *Business & Information Systems Engineering* 5, 5 (October 2013), 343-355. DOI: 10.1007/s12599-013-0282-4
- Henri Hämäläinen. 2016. Tehokas Kanban vai kasa lappuja seinällä? *Contribyte*, 15.12.2016, <http://contribyte.fi/2016/12/15/tehokas-kanban-vai-kasa-lappuja-seinalla/>
- Irum Inayat, Siti Salwah Salim, Sabrina Marczak, Maya Daneva and Shahaboddin Shamshirband. 2015. A systematic literature review on agile requirements engineering practices and challenges. *Computers in Human Behavior* 51, Part B, (October 2015), 915-929. DOI: 10.1016/j.chb.2014.10.046
- Ellen A. Isaacs and John C. Tang. 1993. What video can and can't do for collaboration: a case study. In: *Proceedings of the first ACM international conference on Multimedia (MULTIMEDIA '93)*, 199-206. DOI:10.1145/166266.166289
- Pertti Järvinen ja Annikki Järvinen. 1996. *Tutkimustyön metodeista*. Opinpaja Oy.
- Mikko Korkala, Minna Pikkarainen and Kieran Conboy. 2010. A case study of customer communication in globally distributed software product development. In: *Proceedings of the 11th International Conference on Product Focused Software (PROFES '10)*, 43-46. DOI: 10.1145/1961258.1961269
- Gwanhoo Lee, William DeLone and Alberto Espinosa. 2006. Ambidextrous coping strategies in globally distributed software development projects. *Communications of the ACM*, 49 (10), 35-40. DOI: 10.1145/1164394.1164417
- Robert C. Martin. 2003. *Agile Software Development, Principles, Patterns, and Practices*. Pearson Education.
- Gurpreet Singh Matharu, Anju Mishra, Harmeet Singh and Priyanka Upadhyay. 2015. Empirical study of agile software development methodologies: A comparative analysis. *SIGSOFT Softw. Eng. Notes* 40, 1 (February 2015), 1-6. DOI: 10.1145/2693208.2693233
- Sunila Modi, Pamela Abbott and Steve Counsell. 2013. Negotiating common ground in distributed agile development: A case study perspective. *IEEE 8th International*

Conference on Global Software Engineering, Bari, 80-89.
DOI: 10.1109/ICGSE.2013.18

- Tuomas Niinimäki, Arttu Piri, Peitsa Hynninen and Casper Lassenius. 2009. Studying communication in agile software development: a research framework and pilot study. In: *Proceedings of the ICMI-MLMI '09 Workshop on Multimodal Sensor-Based Systems and Mobile Phones for Social Computing* (ICMI-MLMI '09), Article 5, 4 pages. DOI: 10.1145/1641389.1641394
- Gary M. Olson and Judith S. Olson. 2000. Distance matters. *Human-Computer Interaction* 15, 2 (September 2000), 139-178. DOI: 10.1207/S15327051HCI1523_4
- Ronald E. Rice, John D'Ambra and Elizabeth More. 1998. Cross-cultural comparison of organizational media evaluation and choice. *Journal of Communication*, 48, 3, 3-26. DOI: 10.1111/j.1460-2466.1998.tb02757.x
- Winston W. Royce. 1987. Managing the development of large software systems: concepts and techniques. In: *Proceedings of the 9th international conference on Software Engineering* (ICSE '87), 328-338.
- Maria Saarela-Kinnunen ja Jari Eskola. 2010. Tapaus ja tutkimus = tapaustutkimus? Teoksessa: Juhani Aaltola ja Raine Valli (toim.), *Ikkunoita tutkimusmetodeihin I*. PS-kustannus, 189-199.
- Renuka Sindhgatta, Bikram Sengupta and Subhajit Datta. 2011. Coping with distance: an empirical study of communication on the jazz platform. In *Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion (OOPSLA '11)*, 155-162. DOI: 10.1145/2048147.2048190
- Kai Stapel and Kurt Schneider. 2012. Managing knowledge on communication and information flow in global software projects. *Expert Systems* 31, 3, 234-252. DOI: 10.1111/exsy.649
- Ken Whitaker. 2010. *Principles of Software Development Leadership: Applying Project Management Principles to Agile Software Development*. Course Technology PTR.
- Agustin Yagüe, Juan Garbajosa, Jessica Díaz, Eloy González. 2016. An exploratory study in communication in agile global software development. *Computer Standards & Interfaces*, 48, 184-197. DOI: 10.1016/j.csi.2016.06.002

Liite A – Esitietolomake

Tämän lomakkeen tarkoitus on kerätä taustatietoja haastattelua varten, jotta itse haastattelussa voimme keskittyä tutkimukseni kannalta keskeisiin asioihin. Haastattelua varten toivoisin, että palauttaisit mieleesi ainakin yhden ketterän projektin, joka mielestäsi on opettanut eniten viestinnästä asiakkaan kanssa.

Nimesi tai nimikirjaimesi (käytän tätä tietoa vain yhdistääkseni lomakkeen vastaukset oikeaan haastatteluun)

Toteutatko projekteja yksityiselle sektorille, julkiselle sektorille vai molemmille?

- Yksityiselle
- Julkiselle
- Molemmille

Kokemuksesi ohjelmistoprojekteista

	ei lainkaan kokemusta	alle vuosi	1-3 vuotta	3-5 vuotta	yli 5 vuotta kokemusta
Projektinhallintatehtävät	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tarpeiden ja vaatimusten määrittelytehtävät	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
UX- ja UI-suunnittelutehtävät	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän suunnittelutehtävät	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Kehitystehtävät	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Kokemuksesi ketteristä menetelmistä

- alle 3 vuotta,
- 3-6 vuotta,
- 6-9 vuotta
- enemmän kuin 9 vuotta

Missä rooleissa olet toiminut ketterissä projekteissa?

Mikä on nykyinen työroolisi?

Liite B – Haastatteluskripti

1. Tervehtiminen, nopea kertaus suostumuslomakkeen sisällöstä ja sen allekirjoitus, muistutus vapaaehtoisuudesta.

2. Nopea läpikäynti haastateltavan valitseman projektin taustoista.

- Kuinka pitkä projekti oli?
- Kuinka suuri kehitystiimi?
- Työskentelivätkö kaikki tiimin jäsenet samassa paikassa?
- Edellyttikö erityistä toimialaosaamista?
- Oliko valittu kehitysmenetelmä jokin tietty (XP, Scrum, Kanban, etc) vai jokin tarkoitusta varten räätälöity?
- Liittyikö tähän projektiin jotain erityisiä rajoitteita tai haasteita (esimerkiksi tiukka aikataulu, laaja tuleva käyttäjäpohja, lainsäädännön asettamia vaatimuksia tms.)?
- Kuinka merkittävä tämä projekti oli asiakkaalle? Oliko projekti asiakkaalle budjetiltaan iso panostus, oliko tuleva käyttötarkoitus asiakkaan ydinliiketoiminnan kannalta tärkeä tai oliko projektin lopputuloksella mahdollisesti vaikutusta asiakkaan imagoon?
- Mitä opitte viestinnästä asiakkaan kanssa tässä projektissa?

3. Asiakkaan osallistuminen.

- Osallistuiko asiakas suoraan projektiin vai edustiko joku asiakkaan organisaation ulkopuolinen asiakasta projektissa?
- Mikäli asiakkaan edustaja ei jatkuvasti ollut läsnä, millä eri tavoin asiakkaaseen pidettiin yhteyttä? (jos sähköisiä työkaluja, mitä ominaisuuksia niissä oli ja mitä niistä hyödynnettiin)
- Oliko kaikki viestintä asiakkaan/asiakkaan edustajan kanssa muodollista/virallista vai oliko asiakkaan kanssa yhteisiä epävirallisempia tilaisuuksia/viestintäkanavia?
- Oliko asiakkaalla pääsy kaikkiin sähköisiin työkaluihin?
- Pystyikö asiakas osallistumaan tarpeeksi aktiivisesti?

4. Vaatimukset ja muutosten hallinta.

- Kun kehitettävän ohjelmiston vaatimuksia kirjattiin, ketkä kaikki osallistuivat suunnittelutyöhön?
- Millä tavoin pidettiin kirjaa kehitettävistä ominaisuuksista?
- Miten työmääriä arvioitiin ja miten niistä pidettiin kirjaa?
- Miten vaatimusten prioriteeteista päätettiin?
- Jos vaatimukseen liittyen heräsi kehityksen aikana kysymyksiä, mistä vastauksia lähdettiin hakemaan? Otettiin asiakkaaseen yhteyttä vai oliko tarvittavaa tietoa koostettu jonnekin?
- Millä tavoin asiakas pystyi seuraamaan projektin edistymistä?
- Oliko projektilla ennalta määrättyjä ”tarkistuspisteitä”?
- Miten mahdollisista muutostarpeista viestittiin asiakkaan ja kehitystiimin välillä?
- Miten muutokset kirjattiin ylös?