
KÄYTÄNNÖN KETTERÄT MENETELMÄT LAAJA- ALAISESSA OHJELMISTOKEHITYKSESSÄ



Ylemmän ammattikorkeakoulututkinnon opinnäytetyö

Teknologiaosaamisen johtaminen

Visamäki 9.3.2013

Mika Roima



VISAMÄKI

Teknologiaosaamisen johtaminen

Tekijä

Mika Roima

Vuosi 2013**Työn nimi**

Käytännön ketterät menetelmät laaja-alaisessa ohjelmistokehityksessä

TIIVISTELMÄ

Yrityksen tuotelinjan päätoimialana on ohjelmistokehitys matkaviestinnän verkkolaitteiden tarpeisiin. Käytössä on ollut tarkastellun ajanjakson aikana useita eri tyyppisiä ohjelmistokehityksen käytäntöjä. Perinteisestä vesiputousmallin mukaisesta ohjelmistokehityksestä on vähitellen siirrytty ketterien ohjelmistokehityksen menetelmien pariin. Muutoksella ei kuitenkaan saavutettu haettua hyötyä. Sen sijaan kehitysympäristön laajentuminen kansainvälisesti yhdessä uusien menetelmien ja ohjelmointikielien kanssa tuotti tuotekehitykselle ja ylläpidolle suuria ongelmia.

Tutkimuksen aikana yrityksessä ehdittiin kokea kaksi sisäistä organisaatiomuutosta. Yhteistyökumppanien osalta koettiin yhden osapuolen toiminnan laajentaminen ja toisen osapuolen osalta kaiken toiminnan irtisanominen, jonka vuoksi aluksi suunniteltua otantaa haastatteluissa supistettiin ainoastaan yrityksen sisäiseksi tutkimukseksi.

Tutkimuksen tavoitteena oli selvittää mikä ketterien menetelmien joukkoon kuuluvan scrum -menetelmän käyttöön ottamisessa epäonnistui. Samalla haluttiin kartoittaa tuotekehityksessä ilmeneviä haasteita ja hidasteita laaja-alaista toimintaa koordinoivan toimipisteen sisällä. Tutkimus perustui kokeneiden ohjelmistokehittäjien ja heitä työssään johtaneiden henkilöiden kokemuksiin ja havaintoihin.

Tutkimuksen voidaan katsoa saavuttaneen tavoitteensa. Esiin nousi lukuisia kohteita joiden osalta toimintaa on mahdollista parantaa. Yrityksen kannalta suurimpiin ongelma-kohtiin esitetään myös parannusehdotuksia tämän työn sisällä.

Avainsanat Ohjelmistokehitys, Ketterät menetelmät, Scrum, Käytäntö**Sivut**

68 s. + liitteet 3 s.

VISAMÄKI

Teknologiaosaamisen johtaminen

Author

Mika Roima

Year 2013

Subject of Master's thesis

Agile methodologies in practice

ABSTRACT

Product line inside the company develops software for the needs of mobile telecommunications. During the examination period there has been several different software development methodologies in use. From the traditional waterfall model the practice shifted slowly to the utilization of agile methodologies. This change did not achieve all the goals it was targetting for. Instead the extension of development environment internationally along with the new methodologies and new programming languages resulted problems for research and development and maintenance.

During this research the company initiated two internal organization changes. With collaborators there was big changes also. One collaborator was totally ramped down. Another collaborator started a whole new site for increased requirements from company. Due these changes in collaborator functionalities the originally planned research area was restricted to be only internal. The research was done inside the site controlling all development of product.

The goal for this research was to find out what went wrong in utilization of Scrum methodology. Mapping slowdowns and internal challenges and problem was another goal for this project. This research is based on experiences and notifications of experienced software developers and local management.

Research can be seen as success. Several items with improvement possibilities were raised up. Improvement ideas are introduced for the biggest problems from the company's point of view.

Keywords Software development, Agile methods, Scrum, Practice

Pages 68 p. + appendices 3 p.

SISÄLLYS

1	JOHDANTO.....	1
2	VIITEKEHYS	2
2.1.	Johtaminen	2
2.1.1.	Osaamisen johtaminen.....	3
2.1.2.	Henkilöstön johtaminen.....	5
2.1.3.	Johtamisen haasteet	10
2.2.	Ohjelmistokehityksen käytäntöjä	13
2.2.1.	Hallinnolliset käytännöt.....	13
2.2.2.	Tekniset perusteet	14
2.2.3.	Laadunvalvonta	14
2.3.	Ohjelmistokehityksen menetelmät yrityksessä	16
2.3.1.	Vesiputousmalli	16
2.3.2.	Ketterät menetelmät ja Scrum	17
2.4.	Ohjelmistokehityksen parhaat käytännöt	19
2.4.1.	Työympäristö.....	19
2.4.2.	Suunnittelu.....	20
2.4.3.	Toteutus	21
2.4.4.	Julkaisu	22
2.4.5.	Laatu ja klassiset virheet	22
2.4.6.	Ohjelmistokehitysprojektin johtaminen	25
3	TUTKIMUKSEN SUORITTAMINEN	28
3.1.	Tutkimuksen sisällön määrittely	28
3.1.1.	Tutkimusongelman määrittely.....	28
3.1.2.	Tutkimuskysymykset.....	28
3.1.3.	Tutkimusalueen rajaaminen.....	28
3.2.	Menetelmä 1: Nykytilanteen kartoittaminen.....	29
3.2.1.	Kehitysvaiheet	29
3.2.2.	Yrityksen toimintaympäristö	32
3.3.	Menetelmä 2: Haastattelututkimus.....	37
3.3.1.	Haastattelujen aihealueet	37
3.3.2.	Haastattelut tapahtumina	38
3.3.3.	Tulosten käsittely.....	39
4	TULOKSET	40
4.1.	Tulosten käsittely	40
4.2.	Haastatteluosuuden tulosten yhteenveto	40
4.2.1.	Yhteinen ymmärrys	40
4.2.2.	Ryhmät	40
4.2.3.	Muutokset	41
4.2.4.	Resurssien kohdentaminen	42
4.2.5.	Kommunikaatio	42
4.2.6.	Hukkaresurssit	43
4.2.7.	Työkalut.....	45
4.2.8.	Tuotetietämys	45
4.2.9.	Osaaminen	46
4.2.10.	Riippuvuudet	47

4.2.11. Demot	47
4.2.12. Tekninen velka	48
4.2.13. Esimiehet ja johtaminen	49
4.2.14. Näkyvyys	49
4.2.15. Asiakashyöty	50
4.2.16. Toimintatavat.....	50
4.2.17. Päähidasteita	51
4.3. Kollektiivinen näkemys muutoksista	52
5 POHDINTA.....	57
5.1. Reliabiliteetin ja validiteetin arviointi.....	57
5.2. Menetelmien soveltuvuus.....	58
6 KONTRIBUUTIO	60
6.1. Henkilöstön motivointi.....	60
6.2. Ennustettavuuden parantaminen	61
6.3. Iteraation sisällön määrittäminen	62
6.4. Työkalujen ajanmukaistaminen.....	62
6.5. Tulevia kehityskohteita	62
LÄHTEET	64
LÄHTEET (ANON) (LUOTTAMUKSELLINEN).....	68
Liite 1	Ketterän ohjelmistokehityksen julistus (Agile manifest)
Liite 2	Yhteenveto ketteristä ohjelmistokehitysmenetelmistä
Liite 3	Vikakorjausproseduuri

1 JOHDANTO

Tämän työn tilaaja on tietoliikenteen alalla ohjelmistoja ja palveluratkaisuja tuottava yritys, jota kutsutaan tässä työssä vain yritykseksi anonymiteetin säilyttämiseksi voimakkaasti kilpaillulla toimialalla. Tähän ratkaisuun päädyttiin yrityksen vaatimuksesta. Tästä syystä myös yrityksen sisäiseen materiaaliin viittaavat osat (anon) on poistettu julkisesta työstä.

Tässä työssä on tarkoitus keskittyä tutkimaan ohjelmiston kehitysprosessia ja siihen liittyviä työvaiheita. Tavoitteena on kartoittaa kehitysprosessin vaiheita joissa ketterien ohjelmistokehitysmenetelmien mukaiset toimintatavat eivät ole toimineet yrityksessä tai joissa on esiintynyt muunlaisia ongelmia. Työn sisältö keskittyy yksittäisen tuotelinjan toimintaan, sisältäen suppealla tasolla koko tuotteen elinkaaren alkaen ominaisuuksien määrittelystä ja päättyen jälkikäteen asiakkaille tarjottavaan tuotetukeen. Pääkohteena työssä ovat kuitenkin tuotantoputken keskellä olevat olennaiset kohdat; ohjelmointityö, tuotetun ohjelmakoodin testaaminen ja myöhempi, julkaisun jälkeinen ylläpito.

Yrityksessä on viime vuosina siirrytty käyttämään ketteriä kehitysmenetelmiä. Käytäntö on kuitenkin osoittanut etteivät menetelmät välttämättä sovellu suoraan oppikirjamallin mukaisesti käyttöön. Prosessia on toistuvasti muokattu sekä organisaatiotasolla että pienemmässä mittakaavassa tuotelinjan sisällä, mutta ongelmat ovat säilyneet esiintyvien oireiden vain muuttaessa muotoaan.

Esille selvästi nousseita ongelmakohtia ketterien menetelmien teorian mukaisessa toiminnassa ovat olleet muun muassa kehitysesurssien lukumäärä, resurssien hajanaisuus maailmanlaajuisesti, ohjeistettujen toimintamallien jatkuva muuttuminen, organisaation jatkuvat muutokset sekä eri työvaiheiden synkronointi toisiinsa ja kokonaisprosessiin. Näiden aihepiirien seurauksena onkin havaittu usealla eri osa-alueella selkeitä puutteita, selkeimpinä esiin nousevina oireina laatuongelmat ja aikataulujen pitävyys.

Viitekehityksen laatimisen jälkeen tässä opinnäytetyössä määritetään tutkimusongelma, sen ratkaisuun vaadittavat tutkimuskysymykset sekä rajataan tutkimusalue. Varsinainen tutkimus suoritettiin kaksiosaisena, joista ensimmäinen perustui havainnointiin ja empiirisiin kokemuksiin ja jälkimmäinen teemamuotoiseen haastattelututkimukseen. Näiden tulosten perusteella laadittiin sitten yritykselle toimenpide-ehdotuksia toiminnan tehostamiseksi.

2 VIITEKEHYS

Tuotteen tämänhetkisen historian voidaan katsoa alkaneen kevättalvella 2007. Tuolloin yrityksessä siirryttiin virallisesti vesiputousmallista ketterien kehitysmenetelmien mukaiseen toimintatapaan. Uudeksi teoreettiseksi taustamalliksi yrityksessä valittiin ketteristä menetelmistä Scrum, jonka erään vaikuttajan, Craig Larmanin, oppeihin uudet työskentelytavat perustuivat.

Ketterät kehitysmenetelmät terminä esiteltiin vuonna 2001 julkaisussa Manifesto for Agile Software Development (Agile Alliance, 2001a). Tämän jälkeen menetelmiä lisääviä ja täydentäviä teoksia on julkaistu vuosittain.

Tässä työssä käytetyt viittaukset ketteriin menetelmiin perustuvat pääsääntöisesti Craig Larmanin teoksiin sekä pidettyihin luentoihin ja opetuksiin. Muiden ketterien menetelmien asiantuntijoiden ja kehittäjien (Schwaber, Ambler jne) teoksia on työssä käytetty lähinnä lisämateriaalina laajentamassa tai selkeyttämässä aiempia näkemyksiä. Lisäksi on haettu toimintamalleja ketterien kehitysmenetelmien ulkopuolelta (esim. McConnell).

Yrityksen siirtyessä ketterien kehitysmenetelmien pariin ja Scrum-menetelmän käyttöön, osallistui myös Craig Larman tähän prosessiin kouluttajan roolissa. Myöhemmin hän yhdessä Bas Vodden kanssa julkaisi kaksi teosta (Larman & Vodde, 2008, 2010) jotka osittain perustuvat näihin kokemuksiin Scrum -menetelmän käyttöön ottamisessa yrityksessä.

Erilaisista ketteristä kehitysmenetelmistä on myös tehty useita opinnäytetöitä, joissa kuitenkin pääasiassa tutkitaan niiden käyttöönottamista sekä sisältöä, perehtymättä syvemmin siihen kuinka hyvin kyseiset toimintamallit käytännössä soveltuvat käytettäväksi laajalaisissa ohjelmistokehitysprojekteissa.

2.1. Johtaminen

Yrityksen toiminnan tuloksen ja ulkoisen imagon kannalta oleellinen asia on yrityksen henkilöstöjohtaminen ja sen tehokkuus. Asia on oleellinen, olipa kyseessä sitten paikallinen tai maailmanlaajuinen yritys. Usein tällä alueella voidaan osoittaa puutteita; eräänä syynä voi olla resurssien puute, eivätkä saneeraukset, tehostamiset tai irtisanomiset yleensä ainakaan helpota tilannetta.

Ulkoisen toimintaympäristön muutokset viime vuosina vaikuttavat yritysten toimintaan voimakkaasti. Taloudellinen ja poliittinen maailmantilanne, turvallisuustekijät sekä tekniikan kehitys asettavat omat haasteensa yrityksille ja henkilöstöjohtamiselle.

2.1.1. Osaamisen johtaminen

Ketteristä menetelmistä Scrum sisältää myös projektinhallinnan työkaluja joka asettaakin erityisiä vaatimuksia johtamiselle. Projektitason johtamisen ohella myös osaamisen johtaminen asiantuntijaorganisaatiossa nousee tärkeään rooliin. Organisaation toiminnan perustana onkin jatkuva oppiminen ja sen mukanaan tuomat haasteet ja mahdollisuudet.

Asiantuntijaorganisaatio

Suuri osa Suomessa tehtävästä työstä on asiantuntijatyötä, joko tietotyötä tai luovaa työtä. Korkeatasoisen koulutuksen tuloksena syntyy asiantuntijoita, mutta asiantuntijan muuttamiseksi huipputekijäksi tarvitaan muutakin; osaamisensa johtamista tai johtamista osaamiselle.

Tärkeitä tekijöitä henkilötasolla ovat aloitteellisuus, kyky kohdata haasteet ja kyky kohdentaa ajankäyttö ja työpanos oikein. Oleellista on myös taito pitää itsensä ajan tasalla kaikin tavoin. Asiantuntijan onkin osattava johtaa ensisijaisesti itseään. Oleellista oman itsensä johtamisessa onkin työnteon asiasisällön ohella kyetä vastaamaan seuraaviin kysymyksiin (Klinge, 2012):

- Mihin työpanos tulee suunnata?
- Mikä on oleellista organisaation menestyksen kannalta?
- Kenen kanssa kannattaa hakeutua yhteistyöhön?
- Mitä tulee kehittää omassa toiminnassa?

Kun asiantuntija Klengen mukaan ymmärtää yllä mainitut kysymykset ja niiden sisällön, hänen ajatusmaailmansa on näiltä osin lähellä organisaation johdon ajatusmaailmaa - ja hän kykenee siten peilaamaan omaa rooliaan kokonaisuuden kannalta. Kuitenkin on sanottu, että ydinosaamiseen keskittyminen kaventaa osaamista ja voi tehdä siitä ongelmallista. Kamenskyn (2008) mukaan osaamisen käsittely ydinosaamisen kautta on kuitenkin hyödyllistä:

- Ydinosaamisen tarkastelu tuo esiin strategisesti tärkeimmät osaamiset.
- Osoittaa osaamisen yhteyden yrityksen liiketoimintaan ja koko toimintaympäristöön.
- Ydinosaaminen on yrityksen kaikkein arvokkainta osaamista koska sen kopiointi on vaikeaa, jos ei mahdotonta.



Kuva 1. Tiimioppiminen (Helakorpi, 2012)

Tiimioppiminen yhdistää organisaation osaamisen yksilön osaamiseen. Kaikki oppiminen tapahtuu yksilöiden ja organisaation välisessä tilassa oheisen kuvan mukaisesti (Helakorpi, 2012). Organisaation kannalta keskeinen tehtävä on osaamisen johtaminen. Osaaminen voi olla paitsi ulospäin havaittavaa tietoa myös piilevää, ns. hiljaista tietoa. Organisaatiotieto on yleensä tietoa jonka täsmällinen esiintyminen tulee ilmi lähinnä toiminnan kautta.

Ryhmätoiminta on organisaation kannalta eräs tapa jaotella toimintoja osaamisen tai toiminnan funktion kautta. Tällöin muodostuu erityyppisiä osastoja, joiden henkilömäärä vaihtelee organisaation toiminnan mukaan.

Hyvä ja huono johtaminen

Hyvä johtaminen on jatkuvaluonteista, koko organisaation toiminnan kattavaa toimintaa. Perusteina hyvälle johtamiselle ovat yrityksen arvot, politiikka ja toimintaperiaatteet. Tarkemmin jaoteltuna hyvään johtamiseen kuuluvia osa-alueita ovat tavoitteiden asettaminen, toiminnan suunnittelu, mittareiden määrittäminen, kehityksen seuraaminen sekä reagointi toiminnassa tapahtuviin muutoksiin. Tärkeitä elementtejä tässä ovat johdon ja tekijöiden välinen vuorovaikutus ja tiedottaminen. Hyvin tehtynä kaikki nämä tekijät ovat itsessään työhyvinvointia ja motivaatiota lisääviä tekijöitä.

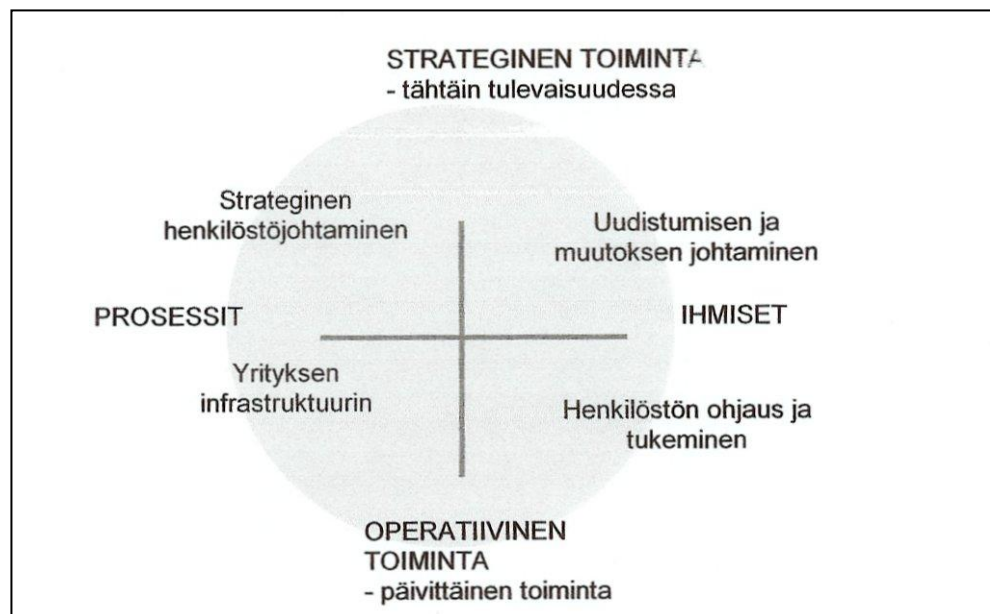
Tampereen yliopiston johtamistieteiden laitoksen tekemässä tutkimuksessa pelolla johtamisesta huonoa johtamista määritellään seuraavilla tavoin (Kokko, 2010). Huono johtaminen on usein huonoa käytöstä. Huono johtaja voi käyttää valta-asemaansa väärin ja olla kunnioittamatta alaisiaan. Tämän vaikutus taas näkyy suoraan työmotivaatiossa ja sitä kautta tuottavuudessa. Pahimmillaan huono johtaminen voi jopa johtaa työkyvyn menettämiseen. Huonon johtajan merkkejä ovat erilaiset valta-aseman väärinkäytännöt, etäisyys, kunnioituksen puute muita kohtaan, sekä puutteellinen kyky herättää luottamusta.

Usean työtätekevän mielestä oma esimies on pelkästään haitaksi työnteolle. Erään selityksen tälle ajatukselle tarjoaa David Rock, neuroleadership -termin kehittäjä (Talouselämä, 2012) joka on myös uranuurtaja neurotieteiden soveltamisessa johtamisen kehittämiseen, muutoshankkeisiin ja valmentamiseen. Hänen mukaansa evoluution tuloksena ihmisiä kiinnostavia sosiaalisia tekijöitä ovat status, ennustettavuus, vaikutusmahdollisuudet, yhteenkuuluvuus ja oikeudenmukaisuus. Perinteiseen auktoriteettiin nojaava esimies lisääkin epävarmuutta ympärillään pienentämällä näitä tekijöitä suhteellisesti ympärillään olevilta työntekijöiltä. Viime aikoina on myös levinnyt käytäntö jossa työntekijöitä asetetaan järjestykseen suoritustensa perusteella. Käytäntö onkin omiaan hyökkäämään kaikkia viittä sosiaalisuuden osa-aluetta vastaan. Hyvä johtaminen sisältää panostamista myös näihin epäsuoriin tekijöihin jotka auttavat pidemmän päälle saamaan aikaan enemmän.

2.1.2. Henkilöstön johtaminen

Henkilöstöjohtajana toimiva henkilö on yhtä aikaa näkijä ja tekijä. Ymmärrys ihmisistä sekä liiketoiminnasta ovat tarpeellisia taitoja. Tästä johtuen henkilöstöjohtamisen alueella vaadittava osaamisalue on laaja-alainen; tärkeinä osa-alueina ovat henkilöstöön liittyvät toiminnot kuten rekrytointi, kehittäminen ja palkitseminen. Lisäksi on myös hallittava työsuojelukysymykset, työsuhteasiat sekä henkilöstösuunnittelu (Schmidt & Vanhala, 2010).

Seuraavassa kuviossa Ulrich (1997) jäsentää henkilöstöjohtamisen kentän neljään alueeseen. Jakoperusteena hän käyttää aikajännettä sekä toiminnan kohdetta.



Kuva 2. Henkilöstöjohtamisen roolit ja tehtävät (Ulrich 1997).

Aikajänne kuvaa joko pitkää, strategisen toiminnan aikaväliä ja toisaalta lyhyen aikavälin päivittäistä operatiivista toimintaa. Henkilöstöjohtamisen pitää huomioida molemmat tarkasteluvälit. Toiminnan kohteen ääripää ovat henkilöstöjohtamisen prosesseissa tai huolehtia ihmisistä. Yllä esitetyn mallin avulla voidaan hahmottaa neljä erityyppistä henkilöstöjohtamisen roolia (Ulrich, 1997):

1. strategien henkilöstöjohtaminen
2. yrityksen struktuurin johtaminen
3. henkilöstön ohjaus ja tukeminen
4. uudistamisen ja muutoksen johtaminen

Henkilöstön rekrytointi ja valinta

Henkilöstön rekrytointiprosessi vaatii omanlaistaan osaamista. Tämän osaamisalueen helpottamiseksi onkin laadittu erilaisia malleja (Työterveyslaitos, 2010), joita hyväksikäyttämällä voidaan prosessia helpottaa.

1. Tehtävänkuvaus ja valintakriteerien määrittäminen
2. Hakuilmoitus ja hankintakanavien valinta
3. Hakuaika ja tiedusteluihin vastaaminen
4. Hakemusten käsittely
5. Haastattelut
6. Soveltuvuusarviointi
7. Muut luotettavuutta lisäävät käytännöt
8. Valintapäätös

Usein rekrytointiprosessi loppuu päätöksen tekoon, työsopimuksen laatimiseen ja allekirjoittamiseen mutta on hyvä huomioida ettei prosessi ole kokonaisuudessaan siinä. Listan jatkoksi tuleekin lisätä vielä:

9. Perehdyttäminen ja sen varmistaminen

Lisäksi loppupäätelmänä ja loogisena jatkoa rekrytointiprosessiin onkin huomioitava että esimiehen jatkuva toiminta henkilöstön ja työaikojen kehittämiseksi on tarpeen henkilökunnan pitkäaikaisen sitoutumisen varmistamiseksi.

Henkilöstön kehittäminen

Henkilöstön kehittäminen on oleellinen osa henkilöstöhallintoa, ja tavoitteiden on tuettava yrityksen perustehtävää ja toimintastrategiaa. Henkilöstön kehittämisellä tarkoitetaan myös omalta osaltaan työnantajan vastuuta henkilöstöstä.

Oheisessa taulukossa on esitetty esimerkinomaisesti Lahden ammattikorkeakoulun julkaisun (Pihkala, 2007) mukaan tärkeimmät henkilöstöjohtamiseen liittyvät toteuttamiskohteet ja eräitä keinoja niihin.

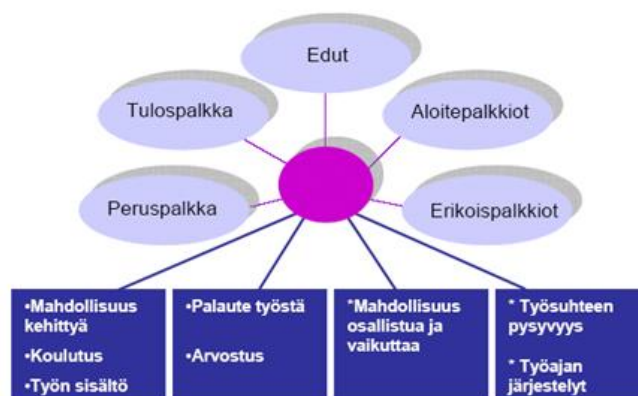
Taulukko 1. Henkilöstöjohtamisen toteuttamiskohteita ja keinoja (Pihkala, 2007)

Kohde	Toteuttamistapa
1. Työn mahdollistaminen	työntekijöiden työnteon kannalta tarpeellisten asioiden tarjoaminen
2. Tunnustus	hyvän suorituksen huomiointi ja osoittaminen
3. Palkitseminen	hyvän suorituksen palkitseminen ja siihen kannustaminen
4. Arvostelu	huomauttaminen huonosta suorituksesta tai sääntöjen rikkomisesta
5. Huomioon ottaminen	avoin, tukeva ja ystävällinen suhtautuminen alaisiin
6. Yksilöllisyyden huomiointi	kunkin alaisen kohtelevminen yksilönä
7. Päätöksiin osallistaminen	alaisten mielipiteiden huomioon ottaminen päätöksissä
8. Delegointi	alaisten omien päätösten salliminen esim. työtavoista
9. Yhteistyön mahdollistaminen	yhteistyön ja ystävällisyyden edistäminen alaisten keskuudessa
10. Konfliktin johtaminen	auttaminen alaisten välillä esiintyvien konfliktien purkamisessa
11. Roolien kirkastaminen	työryhmien sääntöjen, työnkuvausten ja tehtävien valvonta
12. Tavoitteen asettaminen	tavoitteellisuuden ja yksilöllisten tavoitteiden korostaminen
13. Suunnittelu	suunnitelmallisuuden seuraaminen ja korostaminen
14. Tiedottaminen	suunnitelmien ja työhön liittyvien tietojen välittäminen työntekijöille
15. Koordinointi	koordinoinnin seuranta, korostaminen ja parantaminen
16. Innostaminen	työn arvostuksen ja merkityksen korostaminen
17. Aloitteellisuus	uusien toimintatapojen ideointi ja käyttöön kannustaminen
18. Visioiden välittäminen	oman ja organisaation vision esittäminen työntekijöille

Henkilöstön palkkaus ja palkitseminen

Oleellinen osa työnteon käsitteestä on palkkaus - työn määrittely sisältää siitä vastavuoroisesti saatavan hyödykkeen, palkan. Tämän nykyaikaisessa yhteiskunnassa itsestään selvän asian lisäksi voidaan henkilöstöä palkita eri tavoin.

Aalto-yliopiston perustieteiden korkeakoulun julkaisun (palkitsemisen tutkimusohjelma) mukaan palkitsemistapojen kokonaisuus muodostuu aineettomista ja aineellisista palkitsemistavoista oheisen kaavakuvan mukaisesti:



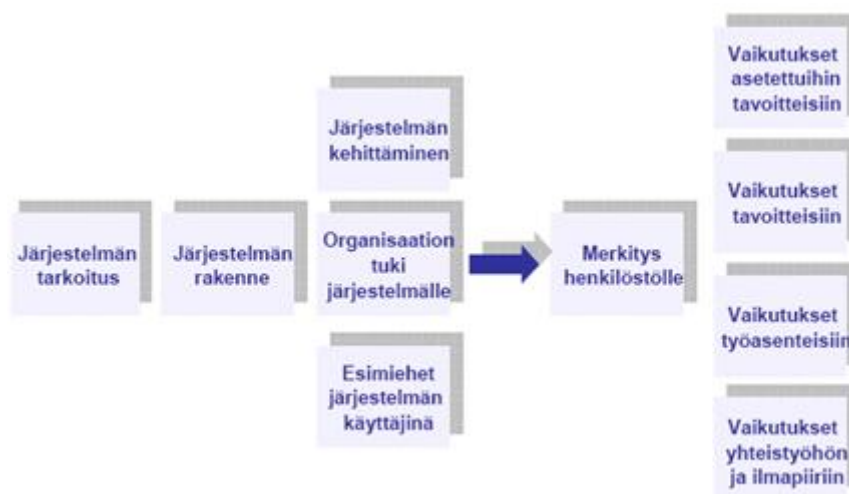
Kuva 3. Palkitsemisen kokonaisuus (Vartiainen, Hakonen & Hulkko 1998)

Palkitseminen voi olla aineellista tai aineetonta. Erilaisia aineellisia palkitsemisen tapoja ovat esimerkiksi peruspalkka, tulospalkkiot, erikoispalkkiot, optiot tai edut. Aineettomia palkitsemistapoja taas edustavat työn sisältö, koulutus- ja kehittymismahdollisuudet, mahdollisuus vaikuttaa ja osallistua päätöksentekoon tai erilaiset työajan järjestelyt. Tutkimusohjelman mukaan kaikki palkitsemistavat ovat

merkityksellisiä työntekijän kannalta, ja siksi yritysten kannattaa niitä tarjota.

Eri palkitsemistavoilla tavoitellaan erilaisia asioita. Työntekijän kannalta kokonaisuus on usein merkittävämpi kuin joku yksittäinen järjestelmän osa-alue hänen arvioidessaan työpaikkansa hyvyttä. Mikään yksittäinen palkitsemistapa ei kuitenkaan ole yksinään ratkaisu yritysmaailman palkkauksen ja palkitsemisen haasteisiin vaan kokonaisuus muodostuu kaikista käytetyistä palkkaus ja palkitsemistavoista sekä niihin liittyvistä prosesseista yhdessä.

Aalto-yliopiston perustieteiden korkeakoulussa on laadittu useisiin eri teorioihin ja tutkimusperinteisiin pohjautuva toimivuusmalli palkitsemiseen.



Kuva 4. Palkitsemisjärjestelmän osa-alueet (Hakonen, Hulkko & Ylikorkala 2004)

Malli kehitettiin alunperin tulospalkkauksen toimivuuden tutkimukseen, mutta se soveltuu hyvin myös muihin palkitsemisen järjestelmiin. Mallissa oletetaan että organisaatio voi vaikuttaa palkitsemisen toimivuuteen kehittämällä sen tarkoitusta, rakennetta ja toteuttamista.

Työsuojelukysymykset

STM:n (Sosiaali- ja terveysministeriö, 2012) mukaan työsuojelun tarkoituksena on ehkäistä, vähentää ja poistaa työstä ja työoloista aiheutuvia tapaturmia, ammattitauteja sekä fyysistä että psyykkistä kuormitusta.

Työsuojelu- ja hyvinvointipolitiikalla halutaan muun muassa:

- Vähentää sairauksista aiheutuvia poissaoloja
- Vähentää pysyvän työkyvyttömyyden riskiä
- Kehittää työoloja
- Ylläpitää työ- ja toimintakykyä

Suomessa työsuojelusta vastaa työnantaja. Työpaikoilla toimii lain velvoittamana työsuojelupäällikkö sekä työsuojeluvaltuutettuja. Työterveyshenkilöstö taas vastaa työhyvinvoinnista ja käytännön tekijöistä yhdessä yrityksen henkilöstöhallinnon kanssa.

Työsuhdeasioiden hoitaminen

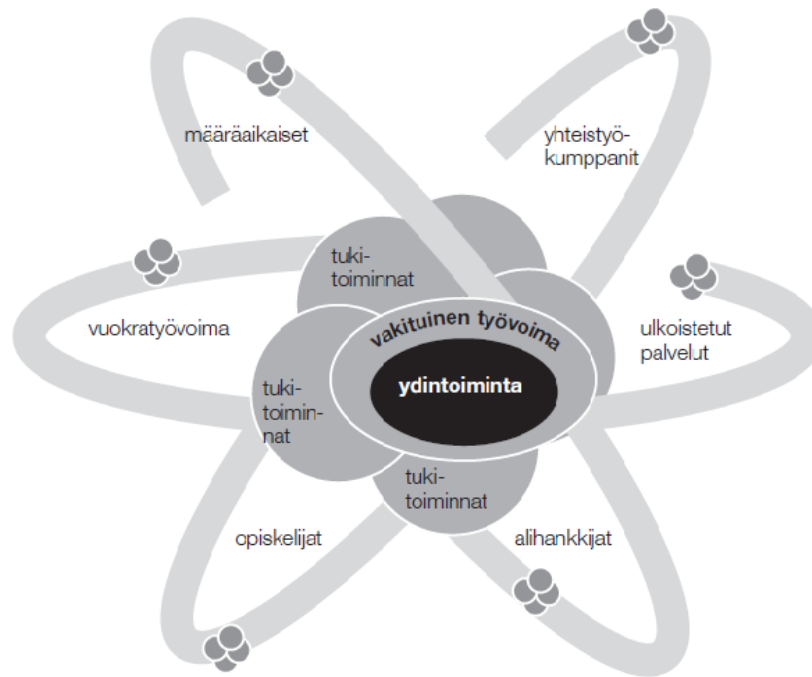
Työsuojeluhallinnon määritelmän (Työsuojeluhallinto, 2012) mukaisesti työsuhteasioilla tarkoitetaan työsuhteen ehtoja koskevien lainsäädännösten ja sopimusmääräysten sekä niihin perustuvien toimenpiteiden ja työkäytäntöjen kokonaisuutta. Tärkeimmät työsuhteisiin liittyvät asiat sisältyvät lakeihin; työsopimus- työaika- ja vuosilomalakeihin.

Työsuhteen perusteina ovat kaksi asiaa; vastuu ja luottamus. Vastuulla tarkoitetaan sitoutumista toimimaan kuten on sovittu, sekä luottamuksella sopimusten ja lupauksen pitämistä. Perusedellytyksiä tähän työpaikoilla ovat sopimusehtojen huolellinen suunnittelu sekä niihin perehdyttäminen, vastuualueiden ja valtuuksien määrittelemine sekä itsenäisen toimintatavan, ammatillisen työotteen sekä yhteistyön hallitseminen. Tuloksellista yhteistyötä tukee esimiehen kyselevä, ohjaava, vastuullistava sekä tunnustusta antava ote, joka motivoi työntekijöitä omia ratkaisuja hakevaan malliin. Tällä on havaittava vaikutus työmotivaatioon ja sen kautta välillisesti tulokseen ja tuloksellisuuteen.

Työsuhdeasioissa yrityksen puolelta tapahtuva ehtojen laiminlyönti tai työntekijän oikeuksien jatkuva toteutumattomuus voivat aiheuttaa epäluottamusta, joka taas alentaa motivaatio- ja suoritustasoa, sekä voi johtaa ristiriitoihin, vastuuttomaan käyttäytymiseen ja työn tuloksen huononemiseen. Työntekijän puolelta laiminlyönnit taas johtavat nopeastikin luottamuksen menettämiseen ja vaarantavat sitä kautta työsuhteen jatkuvuuden.

Henkilöstösuunnittelu

Perinteiset rakenteet työyhteisössä ovat muuttuneet 1990-luvulta alkaen. Pyramidi- ja matriisiorganisaatioista on siirrytty kohti joustavampia organisaatorakenteita. Nykyaikaisen yrityksen henkilöstön hajautumista voidaankin kuvata oheisen kuvion mukaisesti (Pihkala, 2007).



Kuva 5. Verkostoitunut organisaatio (Pihkala 2007)

Pihkalan (2007) mukaan rakenteen ytimessä ovat vastuun kantavat ydinosat; yrityksen omat asiantuntijat. Heidän osaamisensa muodostaa perustan yrityksen toiminnan tehokkuudelle ja rutiinien sujuvuudelle - tämä on oleellista yrityksen kilpailukyvyllä ja sen kehittymiselle. Henkilöstösuunnittelun kannalta oleellista on hahmottaa mitkä roolit voidaan sijoittaa ulkokehälle ja mitkä toiminnot tarkempine osaamisineen on syytä pitää yrityksen hallinnassa.

Hajautetussa toimintamallissa ei yrityksen tasolla tehty tarkastelu enää riitä varmistamaan tuloksellista johtamista. Laadun ja hyvän johtamisen varmistamiseksi on mentävä operatiiviselle tasolle; henkilöstösuhteen sisällä tapahtuviin perehdyttämiseen, motivointiin, ohjaukseen, palkitsemiseen sekä työsuorituksen arviointiin tulee panostaa yrityksen toiminnan kaikilla tasoilla. Samoin työntekijöiden mahdollisuudet oman itsensä, oman työnsä sekä työympäristönsä kehittämiseen tulee huomioida.

2.1.3. Johtamisen haasteet

Nykypäivän henkilöstöjohtamisen haasteista tärkeimpinä (Lehto-Kannisto, 2012) nousevat esiin ikärakenteen muuttumisen seurauksena ikäjohtaminen, globalisaation ja muiden rakenteellisten muutosten tuloksena syntyvä ja voimakkaana vaikuttava muutosjohtamisen tarve sekä henkilöstön työhyvinvoinnista huolehtiminen.

Ikäjohtaminen

Ikäjohtamisella tarkoitetaan työn johtamista ottamalla huomioon erilaiset johtamistarpeet ihmisen elämänkulun ja voimavarojen näkökulmasta. Johtamistarpeet ovat erilaiset nuorilla ja ikääntyvillä.

TTL:n (Työterveyslaitos, 2012) mukaan ikäjohtamisen haasteita ovat mm seuraavat:

- Huomioidaanko eri ikäisten ja eri elämäntilanteessa olevien ihmisten erilaiset tarpeet työelämässä?
- Suunnitellaanko työt yksilöllisesti, erilaiset lähtökohdat, voimavarat ja tarpeet huomioon ottaen?
- Onko ikääntyville palautumisaikaa fyysisistä toiminnoista?
- Onko henkilöstöllä vaikutusvaltaa omaan työhönsä ja työvuoroihinsa?
- Työskentelevätkö erilaiset henkilöstöryhmät yhdessä?
- Huomioidaanko osaamisen kehittämisessä erilaiset lähtökohdat, eri ikäisten vahvuudet ja toisaalta haasteet?

Hyvin toteutettuna ikäjohtaminen hyödyttää kaikkia osapuolia, kaikki osapuolet jaksavat työssä pidempään, organisaatiot pystyvät hyödyntämään kaikkia resurssejaan tehokkaammin ja tuottavuus siten paranee ja yhteiskunta hyöttyy pidemmistä ja yhtenäisemmistä työurista.

Työpaikoilla ikäjohtamisen työkaluksi soveltuu muun muassa:

- Tilanteen arviointi (ikä rakenne, työnjako, koulutukseen hakeutuminen)
- Kehittämisteeman valinta (esim. asennemuokkaus tai työajat)
- Toteutetaan ikäjohtamisen toimenpiteet (organisaatiolle räätälöity kokonaisuus)
- Aikaansaannosten arviointi

Onnistuneella ikäjohtamisella voidaan myös mahdollistaa ja helpottaa vanhempien työntekijöiden tietotaidon siirtämistä nuoremmille työntekijöille. Ikäjohtamisella voidaan siis hakea myös sukupolvenvaihdesta.

Jos ikäjohtamisen merkitystä ei ymmärretä, voivat menetykset työyhteisölle olla suuriakin ja korvaamatonta tietoa voi mennä hukkaan kun ikääntyvät työntekijät jäävät eläkkeille. Ikäjohtamisen periaatteiden unohtaminen työpaikalla voi myös helposti johtaa ikäsyryntään, joka on työsyryntärikos.

Muutosjohtaminen

TTL:n artikkeli muutosjohtamisesta (Työterveyslaitos, 2010) esittää että muutosjohtamisen idea voidaan esittää kysymyksenä - onko muutos hallittavissa vai ei? Muutoksen hallinnan peruskysymyksiä ovat:

- Miksi muutos toteutetaan?
- Onko muutos mielekäs?
- Miten muutokseen voi vaikuttaa?

Muutosjohtamisen ydin mahdollistaa näihin kysymyksiin vastaamisen, ja sitä kautta muutoksen sisällön ja toteutustavan samanaikaisen hallinnan. Muutos siis pyritään ottamaan hallintaan löytämällä sille perusteita ja perusteluita, sekä ymmärtämällä sen merkitys ja etsimällä keinoja vaikuttaa sen suuntaan, nopeuteen ja toteutumiseen.

Muutosjohtamisen haasteita ovat muuttuva ympäristö ja työntekijöiden muuttuvat arvot, sekä niistä syntyvät että niihin vaikuttavat paikalliset ja yleismaailmalliset ilmiöt. Paikallisella tasolla kyseessä voi olla esimerkiksi kustannussäästöistä aiheutuvat toiminnan tehostamisprosessit tai maailmanlaajuisesti megatrendinä tunnettu globalisaatio jossa muutos ilmenee yksinkertaisesti työpaikkojen alasajona. Kaikkeaa tätä on kyettävä johtamaan hallitusti.

Työhyvinvointi

Moni yritys kokee avaintekijänä eri ikäisten ihmisten menestyksekkään johtamisen. Eri ikäiset, eri taustaiset ja erilaisia arvoja kannattavat ihmiset sekä näiden ihmisten väliset suhteet luovat oman haasteensa henkilöstötyöhön.

Työhyvinvoinnin ongelmat voivat ilmetä sekä yksilö- että ryhmätasolla. Yksilötasolla se näkyy yleensä työssä jaksamisen ongelmina, kun taas muilla tasoilla esiintyminen ilmenee yleensä konfliktitilanteina ja osapuolten tai jopa ulkopuolisten tahojen asiattomana kohteluna. Työterveyslaitoksen (2010) mukaan työhyvinvoinnin ongelmat voidaan jakaa kahteen pääryhmään, työyhteisön ristiriitoihin ja työkuormituksen hallintaan.

TTL:n mukaan konfliktit ja ristiriitatilanteet ovat parhaimmillaan viestintuojia jotka kertovat mistä kannattaisi keskustella ja mitä kehittää. Näiden viestien havaitseminen ja tulkitseminen ovat usein haasteellisia toteutettavia. Kuitenkin jos viestit jätetään huomiotta, ne voivat johtaa konfliktin henkilöitymiseen tai laajenemiseen, jonka jälkeen niiden selvittäminen muuttuu vieläkin haasteellisemmaksi.

Henkilökohtaisella tasolla kohtuullinen työkuormitus edistää hyvinvoinnin ja työn sujumisen edellytyksiä. Pitkittynyt muutostila työkuormituksessa taas voi johtaa uhkan muodostumiseen työntekijän terveydelle, jaksamiselle ja sitä kautta työkyvylle. Työssä jaksamiseen vaikuttavat sekä henkiseen toimintakykyyn että fyysiseen toimintakykyyn vaikuttavat tekijät.

Henkisen toimintakyvyn ylläpito ja kuormituksen hallinta ovat siis tärkeitä sekä yrityksen että henkilön itsensä kannalta. Kohtuullinen ja tilapäinen stressitila mahdollistaa tehokkaan oppimisen ja hyvän työpanoksen luomisen, mutta pitkittynyt tai voimakas stressitila voi johtaa erilaisiin terveysongelmiin, työuupumukseen, masennukseen tai jopa työkyvyttömyyteen.

Fyysisellä puolella toimintakykyyn vaikuttavat lihaksien ja luuston rasitus tai jopa rasittamattomuus. Tavallisimpia fyysisiä kuormitustekijöitä ovatkin raskas ruumiillinen työ, taakkojen käsittely, staattiset työasennot, jatkuva istuminen sekä jatkuvasti toistuvat liikkeet. Haasteena onkin siis löytää tasapaino eri tekijöiden välillä.

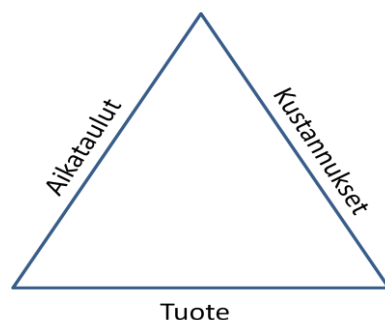
2.2. Ohjelmistokehityksen käytäntöjä

Paras tulos ohjelmistotuotannossa saavutetaan nojaamalla ennalta määriteltuihin perusteisiin. Erilaisia käytäntöjä eri tarkoituksiin on lukuisia, samoin eri organisaatioilla on erilaisia tapoja soveltaa niitä omiin tarkoituksiinsa. Yhteistä parhaiten onnistuneilla ohjelmistoprojekteilla on se, että onnistutaan käytännön toteutustapojen valinnassa (Hetzel 1993).

Yrityksessä käyttöön valitusta ketterästä lähestymistavasta huolimatta onnistunut kehitystyö vaatii todellisten kehittämisperusteiden tunnistamisen. Nämä perusteet ovat hallinnolliset, tekniset ja laadunvalvonnalliset käytännöt (McConnell, 2002).

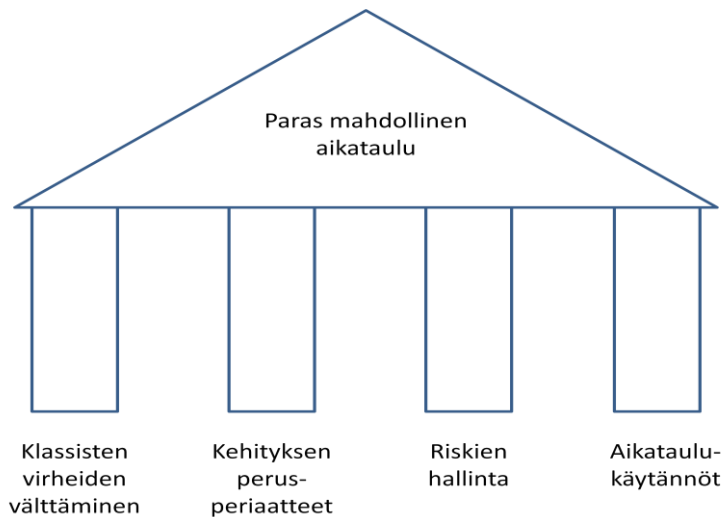
2.2.1. Hallinnolliset käytännöt

Hallinnolla käsitetään kolmiota aikataulu-kustannus-tuote. Perustana tässä määritellään tuotteen koko, joka sisältää myös toiminnallisuudet ja muut tuotteelle ominaiset piirteet. Suunnitelmallisiin kustannuksiin lasketaan mahdollisimman kattavasti henkilöresurssien tarve, työkalujen tarve, laitteistotarpeet sekä mahdolliset muut, tiedossa olevat kulut.



Kuva 6: Hallinnon perusteet (McConnell, 2002)

Tämän jälkeen voidaan etsiä tuotteen kokoon sopivat kehitysresurssit, sekä laaditaan lopuksi aikataulu-arvio tuotteen luomiselle.



Kuva 7. Nopean ohjelmistokehityksen peruspilarit (McConnell, 2002)

Näiden alustavien osien jälkeen valvotaan että tuote pysyy ennalta määrättyjen aikataulu-, kustannus ja laaturajojen sisällä. Lisäksi pitkän aikavälin kehitykselle oleellista on kaikenlaisen oleellisen mittausdatan kerääminen ja tilastoiminen projektien ajalta.

2.2.2. Tekniset perusteet

Oleellisen tärkeä peruste tekniselle työlle on tehokas vaatimusten kerääminen ja hallinta. Historiallisesti yli puolet epäonnistuneista ohjelmistoprojekteista ovat epäonnistuneet riittämättömästä vaatimusten hallinnasta johtuen (Kitson & Masters, 1993). Teknisiin perusteisiin kuuluvat myös huolellisesti tehty kattava suunnitelma tuotteesta, tuotteen rakentaminen sekä kokoonpanon hallinta. Rakentaminen sisältää koodauskäytännöt, dokumentointi, varmistukset, virheidenhallinnan käytännöt, koontiohjeistukset sekä testaaminen eri vaiheineen. Kokoonpanon hallinta, SCM (en. Software Configuration Management) on oleellinen osa mitä tahansa laajempaa ohjelmistohanketta. Se on myös eräs laadunvalvonnan tavoista.

2.2.3. Laadunvalvonta






Jos ohjelmistossa on liikaa virheellisyyksiä, voivat kehittäjät käyttää ohjelmistokokonaisuuden korjaamiseen enemmän aikaa kuin uuden toiminnallisuuden kehittämiseen. On todistettu (McConnell, 2002), että organisaation tuotekehitys toimii tehokkaammin jos virheitä ei päästetä lopulliseen tuotteeseen. Virheiden aikainen torjuminen perustuukin laadunvalvontaan panostamiseen ensimmäisestä suunnitteluaihiosta lähtien.

Useassa tapauksessa ohjelmistoprojektin laatuongelmat ilmenevät koodissa joka joudutaan tekemään uudelleen sen sijaan että olisi alunperin uhrattu siihen resursseja. Seurauksena voidaan joutua analysoimaan koodia useasti, tekemään itse ohjelmointityö uudelleen, testaamaan uusi koodi, sekä siitä muokattu koodi moneen kertaan sekä pahimmillaan

vieläkin palaamaan sen pariin myöhemmin. Yleensä syynä tähän ovat aikataulupaineet, jotka pakottavat käyttämään oikoteitä vaikka kehittäjä tietäisi hyvin ettei ratkaisu ole paras mahdollinen.

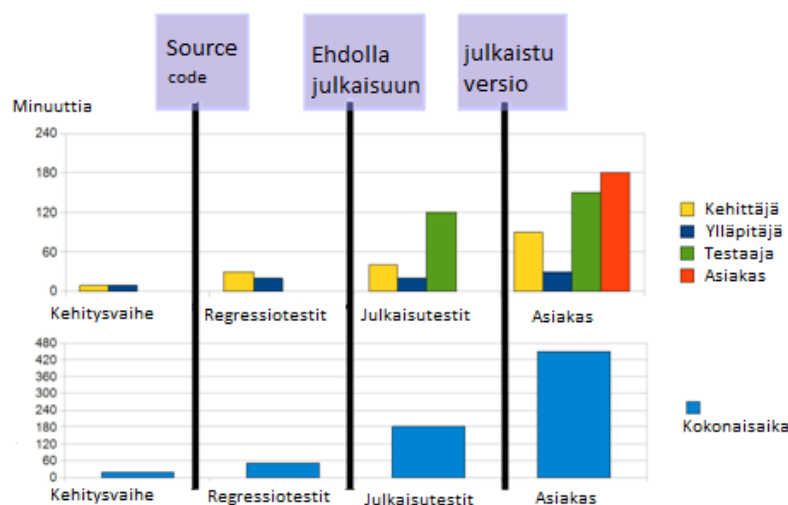
Monessa organisaatiossa vikatasot ovat sellaisia että ne aiheuttavat tarpeettomia venymiä aikatauluihin. Huonon laadun onkin todettu (Jones, 1994) olevan yleisin syy aikataulujen venymiseen. Riittämättömään laadunvarmistukseen on syyllystynyt 60% yrityksistä (Kitson & Masters, 1993). On laskettu että 40-50% ohjelmiston kokonaishinnasta muodostuu vikakorjauksista ja siihen liittyvistä muista kustannuksista (Jones 1986, Boehm 1987 etc). Muistisääntönä tai ohjeena voidaankin käyttää vertausta; tunnin ennakointi säästää 3-10 tuntia myöhemmin tehtävää työtä (Jones 1994). Rahallinen kokonaisarvo jälkikäteen tehtävälle työlle on arvioitu olevan 50-200 kertaa alkuperäisen työn hinta (Boehm & Papaccio, 1988). Yrityksen sisäisessä selvityksessä (Anon 4) on myös tutkittu kustannusrakennetta ohjelmistovirheen havaitsemisen eri vaiheissa oheisen kuvan mukaisesti.

Taulukko 2. Kustannusrakenne (Anon 4)

Vian löytyminen	Oma prosessin vaihe	Seuraava prosessin vaihe	Myöhempi prosessin vaihe	Ennen asiakasta	Asiakkaan löytämä
Hinta	 \$1	 \$10	 \$100	 \$1000	 \$10000
Vaikutus	Minimaalinen	Pieni viive	Työn uudelleen aikatauluttaminen	Huomattava uudelleentekeminen	Takuumaksut Ylläpitokustannukset
			<u>Uudelleentekeminen</u>	Toimituksen viivästyminen	Maine
				Ylimääräinen tutkintatyö	Markkinaosuus

Suorana vertauksena kustannusrakenteeseen on samassa tutkimuksessa havaittu ajankäytön suora korrelaatio tapaukseen sitoutuneiden osapuolten välillä sekä ohjelmistokehityksen käynnissä olevan vaiheen osalta.

Taulukko 3. Ajankäyttö (Anon 4)



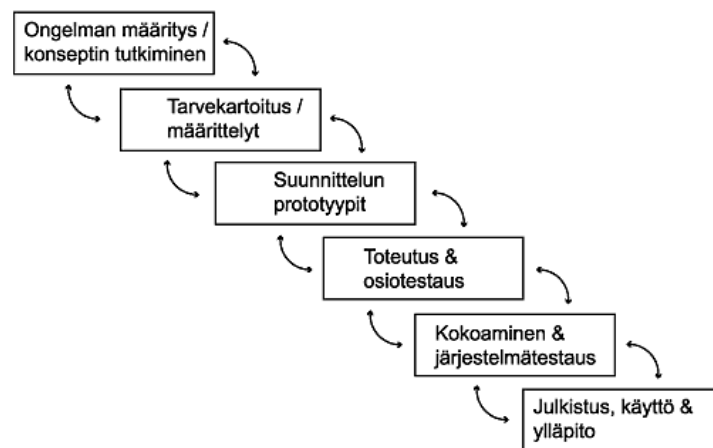
2.3. Ohjelmistokehityksen menetelmät yrityksessä

Yrityksessä on vertailujakson kuluessa käytetty pääasiassa kahta erilaista toiminnallista viitekehystä. Nämä ovat ohjelmistokehityksen vesiputousmalli ja ketteriin ohjelmistokehitysmenetelmiin kuuluva Scrum-menetelmä. Lisäksi näiden välisenä siirtymä kautena on käytössä ollut erilaisia yhdistelmiä näistä työskentelytavoista. Lisäksi yrityksessä on sovellettu mallien ja teorioiden ulkopuolisia toimintatapoja esimerkiksi viankorjauksen yhteydessä, periaatteellisen ja ohjeistetun toimintatavan pysyessä muuttumattomana taustalla. Seuraavissa kappaleissa onkin hieman avattu näitä päämalleja.

2.3.1. Vesiputousmalli

Ensimmäinen vesiputousmallin teorian mukainen esitys pidettiin Herbert D. Beningtonin toimesta kesäkuussa 1956. Esityksessä käsiteltiin ohjelmiston tuotantoa SAGE-projektilla (Semi-Automatic Ground Environment), joka toimi 1958-1983.

Vesiputousmallin terminä katsotaan saaneen nimensä Winston W. Roycen (1929-1995) vuonna 1970 kirjoittamasta artikkelista jossa hän kuvaili toimimattoman ohjelmistokehitysmallin. Mallin juuret ovat tuotanto- ja valmistusprosesseissa joissa oli käytössä selkeästi määritellyt työvaiheet. Tämä työskentelymalli otettiin käyttöön pienin muutoksin myös ohjelmistotalle.



Kuva 8. Vesiputousmallin rakenne (Mielonen & Hintikka, 1998)

Vesiputousmallissa ohjelmointityö on jaettu osa-alueisiin, ja nämä vaiheet käydään läpi yksi kerrallaan; seuraavaan vaiheeseen siirrytään vasta edellisen vaiheen valmistuttua. Vesiputousmallilla onkin selkeät etunsa:

- Suunnitelmallisuus

On osoitettu (McConnell, 2002), että tuotanto- tai ylläpitovaiheessa löytynyt ja korjattu virhe ohjelmistokoodissa on 50-200 kertaa kalliimpi korjata kuin suunnittelun yhteydessä. Tarkka suunnittelu auttaa myös

integraatiovaiheessa koska malli on pakottanut tekemään kaiken ennalta suunnitellulla tavalla.

- Dokumentoinnin laatu

Kaikki työvaiheet on suunniteltu mahdollisimman pitkälle ja kaikki tieto on saatavilla kirjallisessa muodossa. Tämä helpottaa uusien henkilöiden liittymistä kehitystyöhön tai ylläpitoa koska kaikki asiaan liittyvä tieto on saatavilla.

- Seurattavuus

Koska jokainen työvaihe on oma kokonaisuutensa, on projektin seuraaminen kokonaisvaltaisesti helppoa.

Vesiputousmallia pidetään huonona pääasiassa siksi että täydellistä suunnittelua pidetään mahdottomana. Aina on voitava palata aikaisempaan työvaiheeseen. Asiakasvaatimusten eläminen ja jatkuva muuttuminen on myös vaikuttanut vesiputousmallin hylkäämiseen monessa tapauksessa.

2.3.2. Ketterät menetelmät ja Scrum

Ketterät kehitysmenetelmät terminä esiteltiin vuonna 2001 julkaisussa Manifesto for Agile Software Development (Agile Alliance, 2001a), joka syntyi ohjelmistokehittäjien tapaamisen tuloksena. Ketterän kehityksen julistus (Liite 1) sisälsi kirjallisessa muodossa uudenlaisen, ohjelmistolähtöisen kehitysprosessin periaatteet (Ambler, 2002). Sillä siis tarkoitetaan joukkoa ohjelmistotuotannossa käytettäviä menetelmiä eikä niinkään minkäänlaista kehitysprosessia itsessään.

Julistus sisälsi neljä ketterien kehitysmetodien pääajatusta:

1. Henkilöt ja kanssakäyminen on tärkeämpää kuin työkalut ja prosessit.
2. Toimiva ohjelmisto on tärkeämpää kuin kattava dokumentaatio
3. Asiakasyhteistyö on tärkeämpää kuin sopimusneuvottelut
4. Muutokseen vastaaminen on tärkeämpää kuin suunnitelmassa pysyminen

Julistuksen yhteydessä julkaistiin ketterien menetelmien periaatteita (Agile Alliance, 2001b), jotka pätevät edelleen:

- Jatkuviin julkaisuihin perustuva asiakastyytyväisyys
- Muuttuvien vaatimusten toivottaminen tervetulleiksi ja niiden kääntäminen asiakkaan hyödyksi
- Toimivan ohjelmiston nopea ja säännöllinen toimittaminen
- Läheinen yhteistyö eri osapuolten välillä
- Itsenäiset, hyvin toimivat kehitysryhmät
- Kasvokkainen keskusteluyhteys on tehokkain kommunikaatioväline
- Kehitysmittarina toimiva ohjelmisto

- Jatkuva ja tasaisesti etenevä kehitys
- Jatkuva keskittyminen teknisiin yksityiskohtiin ja hyvään suunnitteluun
- Prosessin yksinkertaistaminen hylkäämällä tuottamattomat työvaiheet
- Paras osaaminen on asiantuntijoilla itsellään
- Tehdyn jatkuva seuranta ja vaadittavien muutosten tekeminen

Kehitystyötä pyritään ennakoimaan ja riskejä minimoimaan jakamalla kehitysprosessi iteraatioihin, viikon tai useamman mittaisiin osiin. Yhden iteraation aikana ryhmällä on vaatimukset, resurssit, osaaminen ja laitteisto mitä vaatimusten toteuttaminen annetun ajanjakson sisällä vaatii. Jakson lopussa arvioidaan toteutunut suoriutuma ja päätetään seuraavan iteraation sisältö.

Ketterissä menetelmissä prosessin työvaiheiden suunnittelulle ei anneta yhtä suurta painoarvoa kuin perinteisemmissä ohjelmistokehityksen tyyleissä. Sen sijaan suunnittelu on jatkuvampaa ja sallii suunnitelmien ja vaatimusten muutokset hyvinkin lyhyellä varoitusaajalla. Jyväskylän ammattikorkeakoulun opinnäytetyössä (Kainulainen, 2008) perehdyttiin erilaisten ketterien ohjelmistokehitysmenetelmien eroihin ja niiden soveltuvuuteen erilaisten ohjelmistoprojektien osalta siirryttäessä vesiputousmallin käytöstä ketterien kehitysmenetelmien käyttöön. Tiivistelmä eroista löytyy liitteestä 2.

Scrum itsessään on vanhempi kuin varsinaiset ketterät menetelmät terminä. Scrumin idea esiteltiin ensimmäisen kerran 1986 Hirotaka Takeuchin ja Ikujiro Nonakan toimesta (Takeuchi & Nonaka, 1986). Siinä kuvattiin uudenlainen holistinen lähestymistapa tuotekehitykseen. Mallissa yksi monitaitoinen ryhmä suorittaa vahvasti keskenään lomittuneet työvaiheet koko kehitysprosessin läpi. Nimi Scrum muodostuikin viittauksesta rugby-joukkueen aloitusmuodostelmaan ja myöhemmin jäi käyttöön samankaltaisesta merkityksestä johtuen - molemmat ovat nopeita, itseohjautuvia ja tilanteeseen nopeasti sopeutuvia. Varsinainen Scrum-viitekehys ohjelmistokehitykseen syntyi myöhemmin, pääkehittäjinään mm. Jeff Sutherland, Ken Schwaber, John Scumniotales ja Jeff McKenna.

Suorittavan portaan osalta Scrumin suurin ero muihin ketteriin menetelmiin löytyy etenemän seurannasta. Käytäntöihin kuuluu päivittäinen, lyhyt tapaaminen jossa jokainen osallistuja tiedottaa muille mitä on tehnyt, mitä aikoo tehdä ja mitä ongelmia on kohdannut. Lisäksi Scrum sisältää erilaisia ennustettavuuden ja seurattavuuden malleja projektin johtamisen ja hallinnan tarpeisiin.

Scrum menetelmä itsessään ei tuo ratkaisua mahdollisiin kehitysympäristön tai käytäntöjen ongelmiin. Sen sijaan menetelmä nostaa ne esille (Larman & Vodde, 2008). Ongelmien noustessa esiin tulee usein uhrata aikaa tuotteen kehittämisen kustannuksella ja keskittyä parantamaan kyseisiä kohteita. Usein tässä yhteydessä tehdään se virhe että käydään tosin muuttamaan scrum-käytäntöjä sen sijaan että mietittäisiin miten omia toimintatapoja voitaisiin muuttaa.

2.4. Ohjelmistokehityksen parhaat käytännöt

Tutkimuksen aihepiirin mukaisia avainkohtia on avattu seuraavissa kappaleissa. Aihepiirit on jaoteltu tutkimuksen rakenteen mukaisesti. Asioihin perehdytään sekä teorialmallien että vakiintuneiden parhaiden käytäntöjen mukaisesti.

2.4.1. Työympäristö

Työskentelytila ketterien metodien mukaisessa toiminnassa on selkeästi määritelty. Robert Martinin (2002) mukaan ainoa mahdollinen työtila on avoin tila jossa ryhmän jäsenet työskentelevät. Läheinen yhdessä työskentely sallii avoimen kommunikaation ryhmän jäsenten välillä, ja jokaisen ryhmän jäsenen onkin siten tiedettävä toisten jäsenten senhetkinen tilanne. Tilaa voitaisiin olettaa meluisaksi ja häiritseväksi, mutta Michiganin yliopistossa on tutkittu sopivan työtilan tehostavan toimintaa kaksinkertaiseksi. Rajoituksena vaaditaan ulkopuolisen häiriön sulkemista tilan ulkopuolelle.

Vastaavasti vanhempi tutkija Valtteri Hongisto toteaa TOTI-hankkeen loppuraportissa (Työterveyslaitos, 2012), että melu ja työrauhan puute ovat suoraan yhteydessä siihen, kuinka avotoimistotila toimii asiantuntijayhteisössä. Haittoja (mm. keskittymisvaikeudet, virheiden syntyminen, väsymys) voidaan vähentää huomattavasti panostamalla akustiseen suunnitteluun tiloissa. Myös tilojen ahtaus ja visuaaliset häiriötekijät nostetaan esille samassa tutkimuksessa.

Ryhmät ketterien kehitysmenetelmien parissa ovat avainasemassa. Scott Amblerin (2002) mukaan yksittäisen ryhmän jäsenen tärkein ominaisuus on ryhmätyötaito. Muina tärkeinä ominaisuuksina hän mainitsee kommunikaatiotaidot ja käytännönläheisyyden. Paras ryhmä muodostuu hänen mukaansa sopivassa suhteessa sekoitettuna tietyn osa-alueen asiantuntijoista ja yleisosaajista. Ryhmien perustyyppiä on kolmenlaisia (McConnell, 2002), ongelmanratkaisuryhmiä, luovia ryhmiä ja suorittavia ryhmiä. Nämä perustyyppit pätevät kaikenlaiseen laajempaan ohjelmistokehitykseen jota tehdään ryhmissä.

Hyvin toimivan ryhmän luominen on haastava tehtävä, mutta se on myös avainasemassa projektien onnistumisessa. Scott Ambler antaa myös ohjeet (2002) kuinka ryhmän saa toimimaan tosielämässä;

- Projektin edustajan aktiivisuus on avainasemassa
- Ryhmätyön tärkeyttä ei voi korostaa liikaa
- Uusien tapojen kokeiluun tulee rohkaista
- Jos ryhmä ei toimi, sen kokoonpanoa on muutettava viivyttämättä
- Uuden oppimiselle on löydettävä aikaa

Ryhmien luominen ja kokoonpanon muovaaminen ovat aikaa vieviä projekteja. Uuden kokoonpanon seurauksena on ryhmäytymiseksi kutsuttu ilmiö (Kopakkala, 2005), jonka alkuvaiheissa (muodostuminen, kuohunta) ryhmän ei voida olettaa toimivan tehokkaasti. Vasta ryhmäytymisen

myöhemmissä (yhdenmukaisuus, yhteisöllisyys) vaiheissa voidaan odottaa maksimaalisen tehokasta tuotosta.

Ketterien kehitysmenetelmien mukaan (Larman & Vodde, 2008) tulee ryhmien muodostuksessa välttää muodostamasta johonkin komponenttiin tai muuten yhteen asiaan keskittyviä ryhmiä. Sen sijaan tulee pyrkiä kokonaisuuksia kehittäviin ryhmiin. Näin yhden ominaisuuden kokonaisvaltainen hallinta pysyy rajatun ryhmän hallinnassa ja tällä vältetään rajoitetumpien kohdealueiden muodostamat ongelmat. Ryhmän toiminta sisältää koko ominaisuuden elinkaaren, suunnittelun eri vaiheista käytännön toteutuksen kautta testauksen viimeistelyyn. Mallin mukaisesti ominaisuuden kehittämisen vaatima byrokratia ja koordinoitavuus minimoidaan.

Muita toimivan ryhmän merkkejä ovat ryhmän pitkäikäisyys, yhteinen vastuunkanto, yksinkertainen organisaatorakenne sekä omatoimisuus. Yleiseksi virheeksi nimitetäänkin tässä yhteydessä (Larman & Vodde, 2008) liiallinen projektinhallinta, koska ryhmien sisäinen koordinoitavuus on yksinkertaista. Samassa yhteydessä tähän esitettiin ratkaisuksi manageriportausta leikkaamista kahdella kolmanneksella, joka osin sallii ja osin pakottaa ryhmän ottamaan vastuuta ja kontrolloimaan omia tekemisiään. Ryhmät eivät pääse tavoiteltuun omatoimiseen tilaan niin kauan kuin johtava taso pysyy entisenlaisena. Samoin matriisiorganisaatiosta, jossa työntekijä raportoi useammalle kuin yhdelle johtotason henkilölle, tulisi päästä täysin eroon.

2.4.2. Suunnittelu

Hyvin suunniteltu ohjelmistoprojekti on jaettavissa kolmeen vaiheittaiseen osa-alueeseen (McConnell, 2002). Nämä ovat tuotteen koon arviointi, tähän arvioon perustuva työmääräarvio, sekä työmääräarvioon perustuva aikataulun laatiminen. Parhaiden projektien määrittelyssä taas Bill Hetzel (1993) määritteli tärkeimmiksi toiminnan vaiheiksi seuraavat:

- Arvioinnit ja aikataulujen laatiminen
- Tarvittavan organisaation ja resurssien määrittäminen
- Riskien hallinta
- Strategiset päätökset (mm. ominaisuudet ja alihankinta)

Varsinaisen ohjelmistoprojektin aloittamiseen Scrum-menetelmää käyttäen vaaditaan näiden lisäksi lista tuotteen ominaisuuksista tarkasti määriteltynä ja tärkeysjärjestykseen asetettuna. Tämä ns. tuoteloki (en. product backlog) on tuotteen suunnittelun ja toteutuksen ajan tärkein seurannan muoto joka elää koko projektin ajan. Tästä listasta saadaan milloin tahansa kattava ja täydellinen katsaus koko projektin tilasta. Tuotelokin alla seuraava taso on julkaisuloki (en. release backlog), jota tuotteen omistaja pitää ajan tasalla. Usein tuntematon tai unohtuva käytäntö on tuotteen omistajan velvollisuus määrittää bisnes-arvo jokaiselle julkaisulokin kohdalle ja viestittää se varsinaista kehitystyötä tekeville ryhmille. Listan kohteiden priorisointi tapahtuu pohjautuen tähän bisnes-arvoon sekä arvioituun työmäärään (Larman & Vodde, 2008).

Iteraation alussa järjestetään kaksivaiheinen suunnittelupalaveri. Ensimmäisessä osassa tuotteen omistaja ja ryhmät katselmoivat tärkeimmät kohteet tuoteloki -listalta. Samalla määritetään valmistumisen vaatimukset kyseisen iteraation tuotokselle, ns. tehdyn määritelmä (en. Definition of Done). Ensimmäisessä suunnittelupalaverin osassa keskitytään siis ymmärtämään mitä halutaan. Tuotteen omistajan läsnäoloa ei vaadita tämän jälkeen, mutta hänen on oltava tavoitettavissa myös toisen vaiheen palaverin aikana. Tässä toisen vaiheen suunnittelupalaverissa keskitytään määrittelemään se, miten asetetut tavoitteet saavutetaan. Tässä kohtaa ohjelmistokehitystä tekevät ryhmät määrittelevät itsenäisesti sen, mitä he tulevat tekemään seuraavan iteraation aikana. Määrittelyssä huomioidaan alunperin suunnitellut tekemiset jotka tuotteen omistajan kanssa suunniteltiin ensimmäisessä suunnittelupalaverissa, mutta vaatimusten tarkentuessa voidaan osia siirtää myöhempiin iteraatioihin. Ryhmän omien päätösten pohjalta sitten laaditaan iteraation tehtävälista (en. sprint backlog), jossa listataan kyseisen jakson aikana tehtäväksi sovitut käyttäjätarinat (en. user story).

2.4.3. Toteutus

Käytännössä iteraation alettua, ryhmät ottavat käyttöön päivittäisen scrum -tapaamisen. Tämä tapahtuu päivittäin samaan aikaan, ja kaikki ryhmän jäsenet osallistuvat tapaamiseen. Tarkoituksena on pitää lyhyt tilannekatsaus siihen, mitä muut ryhmän jäsenet ovat tehneet, mitä tehdään seuraavaksi ja tuoda esiin mahdolliset ongelmat. Ongelman noustessa esiin on scrum masterin vastuulla ratkaista se. Tilaisuuden aikana ei myöskään keskustella, mutta sen jälkeen voidaan pitää erillinen keskustelutilaisuus.

Ryhmän jäsenet pitävät iteraation tehtävälistan ajantasaisena päivittäin. Näiden päivitysten perusteella joku ryhmän jäsenistä laatii ns. sprint burndown chart -kaavion, jossa näkyy iteraation etenemä graafisessa, nollaa lähestyvässä ja jäljellä olevan työmäärän esittävässä muodossa.

Iteraation lopussa katselmoidaan (sprint review) tuotos tuotteen omistajan kanssa. Tämä katselmus usein nimetään virheellisesti demoksi joka ei kuitenkaan ole itse päätarkoitus. Sen sijaan tässä nousee esiin scrum menetelmän avainkohta; tutki ja muokkaa (en. inspect and adapt; Larman & Vodde 2008). Tässä annetaan siis tuotteen omistajalle mahdollisuus päivittää product backlogia vastaamaan todellista tilannetta.

Sprint review -tilaisuuden jälkeen järjestetään vielä sprint retrospective, jossa keskitytään tutkimaan ja muokkaamaan itse prosessia. Valitettavan usein tämäkin vaihe jätetään pois käytöstä koska se ei tuota suoraan mitään; se on kuitenkin pääväline mahdollisten kehitys- ja tehostamiskohteiden havaitsemisessa organisaation sisäisessä toiminnassa. Retrospective -tilaisuuden jälkeen sprintin tekemistä on vielä jäljellä release backlog -listauksen sekä release burndown -kaavion päivittäminen. Näiden jälkeen voidaan aloittaa valmistautuminen seuraavaan iteraatioon, eli aloittaa uusi sprint.

2.4.4. Julkaisu

Ideaalitilanteessa tuote on julkaisuvalmis jokaisen iteraation päätteeksi. Tällöin oletetaan ettei lisätyötä vaadita. Lisätyöksi lasketaan kaikki tehtävät jotka olisi pitänyt saada valmiiksi jakson kuluessa, mutta jotka jostain syystä ovat jääneet tekemättä. Näitä tehtäviä voivat olla esimerkiksi jonkun osan testaaminen tai dokumentointi. Voidaan myös ottaa tavoitetilaksi se, että kaikki työvaiheet ovat valmiit jokaisen iteraation päätteeksi. Tosin tällöin varsinaisen kehitystyön määrä jakson kuluessa jää jo suunnitelmallisesti vähäisemmäksi. Yleisestä suurten organisaatioiden kankeudesta sekä heikoista käytännöistä johtuen joudutaan usein lisäämään ns. julkaisuiteraatio (en. release sprint), jossa viimeistellään kaikki keskeneräiset työtehtävät ja osa-alueet. Tämän julkaisuiteraation tarpeellisuus osoittaa kuitenkin heikkoutta jossain kohtaa käytettyjä kehitysmenetelmiä tai käytäntöjä. Ideaalitilanteessa tätä ei tarvita (Larman & Vodde, 2008).

Yleisesti esitetty kysymys iteratiivisessä kehitystyössä on se miten voidaan tehdä pitkän aikavälin arvioita ja suunnitelmia. Tässä tuleekin esiin vain kaksi eri tapausta (Larman & Vodde, 2008); Täysin uusi tuote sekä vanha tuote. Uuden tuotteen osalta suunnitteluun voidaan käyttää jopa viikkoja. Vanhan tuotteen osalta ei mitään erityisiä vaatimuksia pitäisi tulla esiin koska tehtävälistaa on päivitetty jokaisen jakson yhteydessä osoittamaan tuotteen sen hetkistä tilannetta. Varsinaista tuotteen tiekarttaa ja muita laadittuja aikatauluja muokataan jatkuvasti tähän arvioon perustuen.

Julkaisu voi myös perustua ajankohtaan valmiusasteen sijaan. Tällöin on tuotteen omistajan vastuulla määrittää toteutettava sisältö. Tärkeimmiksi on tällöin nostettava muiden mahdollisten tuotteiden toiminnan esittämät vaatimukset oman, sisäisen toiminnan sijaan.

2.4.5. Laatu ja klassiset virheet

Ohjelmistojen kehittäjät saavuttavat yleisesti itsekunnioitusta tuotoksistaan. Korkealaatuisen tuotteen tekeminen motivoi kehittäjiä voimakkaasti, eikä tätä tule estää johtamisen keinoin. Jos johtotason henkilöt vaativat tinkimään laadusta aikataulujen kustannuksella, viedään kehittäjiltä tuolloin ylpeys laadukkaasta tuotoksesta sekä omistajuus tuotteesta tai sen osasta, koska päätös laadun leikkaamisesta tulee muualta.

Teoksessa Ohjelmistotuotannon hallinta (McConnell, 2002) listataan yleisiä laatuun vaikuttavia virheitä jotka voivat toteutua riippumatta siitä mitä käytännön menetelmiä yrityksessä käytetään. Listattuja pääryhmiä ovat ihmiset, prosessi, tuote ja teknologia. Mainittujen ryhmien alla listatut virheet tunnetaan klassisten virheiden nimellä.

Ihmiset

Ylivoimaisesti tärkein ihmisiin ja henkilöresursseihin liittyvä tekijä on motivaatio (Boehm 1981). Motivaatio-ongelmat vaikuttavat tuottavuuteen ja laatuun enemmän kuin mikään muu tekijä. Muita henkilöstön osalta vaikuttavia tekijöitä ovat henkilöstön osaaminen sekä hallitsemattomat ongelmatyöntekijät. Lisäksi mainitaan klassisimmaksi klassisista virheistä ihmisten lisääminen jo myöhässä olevaan projektiin; usein kokemattomien henkilöiden tuominen projektiin kuluttaa vanhojen työntekijöiden työaika enemmän kuin mitä uudet henkilöresurssit voivat antaa projektin kuluessa. Myös johdon, asiakkaiden sekä työntekijöiden itsensä epärealistiset odotukset voivat aiheuttaa ongelmia, Standish Groupin tekemässä tutkimuksessa (1994) olikin realistiset odotukset listattu ohjelmistoprojektin onnistumiseen vaikuttavien viiden tärkeimmän tekijän joukkoon. Tärkeimpänä löydöksenä tutkimus nimesi asiakkaiden ja loppukäyttäjien osallistumisen projektin alkuvaiheen suunnitteluun.

Prosessi

Virheet prosessissa hukkaavat yleisesti ihmisten taitoja ja aiheuttavat turhaa työtä, ja sitä kautta hidastavat prosessin etenemistä. Yleisesti aikataulut nähdään liian optimistisesti, ja sitä kautta kohdataan paineita lyhentää projektin kriittisiä suunnittelu- ja aloitusosia (McConnell, 2002). Pitkällä tähtäimellä tämä painostaa kehittäjiä ja siten voi heikentää motivaatiota ja tuottavuutta sitä kautta. Projektin riskien kartoittamiseen, hallintaan ja eliminointiin käytetyllä ajalla voidaan estää vakavammat viivästyksiset myöhemmissä vaiheissa.

Jos yrityksellä on kiire, voidaan kokonaisuuksia tai paloja siirtää alihankkijan toimeksi. Alihankkijat kuitenkin joskus toimittavat tuotantonsa myöhässä tai täyttämättä sille asetettuja laadullisia tai toiminnallisia vaatimuksia (Boehm, 1981). Tästä syystä alihankkijan käyttö on hallittava tarkasti jotta projekti ei hidastu nopeutumisen sijaan. Aikataulujen muodostuessa ongelmaksi, projektisuunnitelma saatetaan hylätä. Tästä voi seurata siirtyminen tilaan, jossa ongelmana ei ole suunnitelman hylkääminen vaan siirtymä koodaa-ja-korjaa -tilaan projektin muuttuessa osin tai pahimmillaan kokonaan hallitsemattomaksi.

Aikatauluongelmien kasaantuessa tilanne johtaa helposti entistä pahempaan suuntaan. Laadunvalvonnasta voidaan tinkiä, suunnittelussa ja suunnitelmien katselmoinnissa oikoa, tinkiä ohjelmiston toiminnoista tai muita, myöhemmin moninkertaisina kostautuvia tekijöitä (McConnell, 2002). Kireät aikataulut voivat johtaa aikataulun puitteissa tehtyyn, mutta tietoisesti heikotasoiseksi jääneeseen koodiin. Charles Lecht varoitti tästä jo neljä vuosikymmentä takaperin (Lecht, 1967); Ohjelmistokehittäjä pyrkii aina kaikin keinoin saavuttamaan annetut tavoitteet annetun aikarajan puitteissa. Jos aika on vähissä, saatetaan käyttää oikopolkuja -kovakoodattuja toimintoja, copy&paste-koodia, testauksessa oikomista, tehdään ylitöitä tai käytetään jotain muita laatua heikentäviä tapoja (Scwaber, 2007), jonka tuloksena koodin laatu kärsii. Tästä seuraa

kumuloituvia ongelmia, joiden korjaaminen on sitä työläämpää, mitä pidemmältä ajalta kertymää on päässyt syntymään. Tämänkaltaisen laatuongelma tunnetaan myös nimeltä tekninen velka.

Tuote

Lopulliseen tuotteeseen liittyviä ongelmia voidaan listata lähinnä ominaisuuksien osalta. Asetetaan enemmän vaatimuksia kuin tuote oikeasti tarvitsee tai lisätään vaatimuksia kesken projektin jolloin ominaisuuksien prosentuaalinen osuus vastaa suoraan projektin vaatiman ajan prosentuaalista kasvua (Jones 1994).

Teknologia

Teknologiaan liittyvät tekijät eivät vaikuta suoraan tuotteeseen, vaan lisäävät epäsuorasti ajankäyttöä ja mahdollisesti kustannuksia. Saatetaan luottaa liikaa ohjelmointikieleen tietämättä kuinka hyvin se soveltuu haluttuun tarkoitukseen tai kehitysympäristöön (McConnell, 2002). Saatetaan yliarvioida uusien välineiden tai metodien tuoma säästö, tai muutetaan käytettyjä välineitä tai ohjeistuksia kesken projektin.

Nykyaikaisessa, ketterässä kehitystyössä on myös käytettävä tekniikan kehityksen mahdollistamia, erilaisia tapoja laadun parantamiseen. Eräs tärkeä osa-alue on jatkuvan testaamisen periaate, ns. continuous integration, joka määrittelee (Fowler, 2006) siten että jokainen kehittäjä integroi säännöllisesti tuotoksensa koodikantaan, kuitenkin vähintään kerran päivässä. Tallennetun ohjelmakoodin perusteella kokonaisuudesta luodaan useita julkaisuja päivittäin. Nämä julkaisut testataan automaattisesti niin nopeasti kuin mahdollista. Tiheä testausrytmi puolestaan sallii mahdollisesti julkaisun hajottavan toiminnan kiinnisaamisen nopeasti ja siten lyhentää korjaamiseen vaadittavaa aikaa. Tiheästi ja säännöllisesti tapahtuva automaattinen testaus säästää aikaa myös muilla tasoilla; se pienentää integraation sisältämää riskiä, pienentää huonon laadun riskiä, helpottaa ja nopeuttaa virheen paikallistamista, helpottaa kokonaisuuden seuranta, mahdollisesti parantaa moraalia tuotteen toiminnan näkyessä sekä parantaa asiakassuhteita, jos kehitystyön etenemää päätetään tiedottaa tälle sektorille (McConnell, 2002).

Jatkuvassa koosteen tekemisessä on myös riskinsä. Kehittäjät voivat laatia pikakorjauksia jotka saavat välijulkaisun toimimaan halutusti, mutta jotka vaativat uutta työtä myöhemmässä vaiheessa. Tähän kuluu helposti enemmän aikaa kuin jos ensimmäisessä vaiheessa olisi kehittäjälle annettu mahdollisuus lopulliseen ratkaisuun. Samoin automaattisen testausvaiheen on yleensä todettu pitenevän ja testiaineiston kasvavan projektin edetessä (McConnell, 2002). Tässä vaiheessa unohtuu usein aineiston säännöllinen tarkastelu ja vain merkityksellisen aineiston mukana pitäminen.

2.4.6. Ohjelmistokehityksen johtaminen

Ohjelmistokehityksen johtaminen perustuu samoihin periaatteisiin kuin yleisempi johtaminen. Johdettavana on ihmisiä ja tekniikkaa. Ohjelmistoprojektin johtamisen haastavuus kasvaa välittömästi tilanteessa jossa edes osa tuotteen toiminnasta siirretään oman suoran hallinnan ulkopuolelle.

Motivaatio

Tuottavuustutkimuksissa on havaittu että motivaatio on suurin ihmisen tuottavuuteen vaikuttava yksittäinen tekijä. Motivaatio on kuitenkin tekijä, jonka mittaaminen on vaikeaa joten kehityksen mittareina sen syrjäyttävät usein vähäarvoisemmat, mutta helpommin mitattavat tekijät. Huomattavan harvalla organisaatiolla onkin tehokkaita ohjelmia henkilöstön motivointiin. Heikko henkilöstö ohjelma tai motivointiyritys yleensä tuottaa enemmän haittaa kuin hyötyä (McConnell 2002). Huomattavaa on myös se, että yleisesti johtotason henkilöstöä motivoivat erilaiset tekijät kuin suorittavan portaan henkilöstöä. Tästä johtuen motivointitavat eivät usein toimi kuten niiden ajatellaan toimivan. Tehokkaita motivointituloksia saadaan yleensä aikaan tärkeimpien motivaatiotekijöiden käyttämisellä. Näitä ovat:

Saavutukset. Helppo tapa motivoida on mahdollistaa sellaisen työn tekeminen mistä kehittäjä nauttii. Työn laatu motivoi yleisesti teknisesti suuntautunutta ohjelmistokehittäjää paremmin kuin aikataulut (McConnell 2002).

Omaehtaisuus. Jokainen tekee mieluummin töitä saavuttaakseen oman tavoitteensa kuin jonkun toisen tavoitteet.

Tavoitteet. Jos tavoitteet ja aikataulut on asetettu alunalkaen epärealistisiksi, niiden tavoittelukin voi tuntua turhalta. Samoin liian monien tavoitteiden tai prioriteettien asettaminen on osoittautunut projektille haitalliseksi (Larson & LaFasto 1989).

Kasvumahdollisuudet. Ohjelmistoala muuttuu jatkuvasti, ja tämän johdosta on työntekijän syytä pitää osaamisensa ajan tasalla.

Itse työ. Sisäinen motivaatio tulee kolmesta lähteestä; työ on tunnettava merkitykselliseksi, on koettava vastuuta tuloksista sekä tunnettava työn todelliset seuraukset (Hackman & Oldham, 1980).

Motivaatiota vähentävät tekijät ovat vähintään yhtä tärkeässä roolissa kuin sitä lisäävät tekijät. Näitä tekijöitä voivatkin olla esimerkiksi työtilojen riittämätön valaistus, huono lämpötila ja heikko ilmasto, riittämättömät pöytätilat, keskittymiseen tarvittavan työrauhan puute tai vähemmän modernit työvälineet. Muita motivaatiota voimakkaasti heikentäviä tekijöitä ovat mm. hallinnon manipulointi, liiallinen aikataulupaine, arvostuksen ja palkitsemisen laiminlyönti, johdon liiallinen puuttuminen

käytännön tekemiseen, kehittäjiä unohtaminen heitä koskevien päätösten tekemisestä, tuotannon esteet tai huono laatu (McConnell 2002).

Tekninen johtaminen

Tekninen johtaminen soveltuu paremmin kehittäjille kuin varsinaisille johtotehtävissä toimiville henkilöille. Nämä päälliköt johtavat jo muita ja teknisiin yksityiskohtiin paneutuminen voikin tuntua askeleelle taaksepäin. Sen sijaan kehittäjä voikin tuntea saavuttaneensa askeleen eteenpäin urallaan, jos hänet nimitetään johtamaan teknisesti jotain osa-aluetta projektissa.

Tätä voidaan soveltaa myös laajemmin. Kaikille mahdollisille teknisille osa-alueille voidaan nimetä tekninen johtaja, ja uusimpia projektin työntekijöitä lukuun ottamatta muut voidaan nimetä mentoreiksi jollekin tietylle osa-alueelle (McConnell 2002).

Ulkoistaminen

Ulkoistamisella tarkoitetaan käytäntöä, jossa ulkopuoliselle taholle maksetaan tuotteen kehittämisestä tai ylläpidosta sen sijaan että se tehtäisiin yrityksen omia resursseja käyttäen (McConnell, 2002). Päättavoitteet ulkoistamisessa ovat aikataulujen nopeuttaminen ja kustannussäästöt. Perinteinen ohjelmistoprojektien hallinta nimeää ulkoistamiseen sekä suuria etuja että suuria riskejä; projektin aikataulu voi lyhentyä merkittävästikin, mutta samalla aikatauluriskit kasvavat. Pitkällä tähtäimellä riskit pienenevät ja ulkoistamisen onnistumismahdollisuus kasvaa merkittävästi. Samalla tosin ulkoistamisen pääriskit nousevat esiin; osaaminen ja asiantuntemus yrityksen sisällä menetetään, tulevan kehittämisen hallinta voidaan menettää, luottamuksellisten tietojen suhteen joudutaan tekemään kompromisseja sekä edistymisen näkyvyys ja suora kontrolli hämärtyvät.

Ulkoistamisen onnistumiseksi projektia on käsiteltävä kuin se olisi yrityksen oma, sisäinen projekti. Projektille on laadittava hallintasuunnitelma riskinhallintoineen, sekä sopimusteknisiin yksityiskohtiin on osattava panostaa erityisen voimakkaasti (Marciniak & Reifer, 1990). Lisäksi ulkopuolisen tahon tehdessä kehitystyötä tulee yrityksen omaan vaatimusten määrittelyyn panostaa moninkertaisesti siihen nähden mitä sisäinen tuotekehitys vaatii - ulkopuolinen tekijä tuottaa sen mitä alunperin sovitaan, kaikesta muusta veloitetaan erikseen. Ulkoistamisen yhteydessä olisi myös aina muistettava vaihtoehtoinen toimintatapa, ns. B-suunnitelma, ja varattava myös sekä mahdollisuus että valmiudet sen toteuttamiseen.

Ulkoistamisen onnistumisen avainkohdat ovatkin

- Kumppanin huolellinen valinta
- Sopimus itsessään

- Projektin hallintaan panostaminen
- Sähköiseen kommunikaation ja neuvotteluihin panostaminen
- Tulevat suunnitelmat

Viime vuosina trendinä on ollut kehitystyön siirtäminen edullisten kehityskustannusten maihin, joilla yleisesti tarkoitetaan ns. kehittyviä talouksia. Nämä valtiot sijaitsevat yleensä täysin eri aikavyöhykkeillä, ja meillä itsestään selvät asiat, kuten tietoverkot, puhelinyhteydet ja muut vastaavat tekniset tekijät voivatkin olla ongelmallisia. Aikaerosta johtuen tuleekin kommunikaatioon panostaa erityisen voimakkaasti, koska yleisesti työajat eivät satu samanaikaisiksi eri osapuolilla, ja onkin hyvä sopia sopimusneuvotteluissakin jo alusta alkaen kumpi osapuoli joustaa esimerkiksi palaveriajankohdissa.

Steve McConnell mainitsee seuraavat kohdat teoksessaan Ohjelmistotuotannon Hallinta (2002), joihin tulisi vastata miettiessään tuotannon ulkoistamista. Jos kyllä-vastauksia on enemmän kuin ei-vastauksia, ei tuotannon ulkoistamisessa välttämättä ole järkeä.

Taulukko 4. Ulkoistamisen järkevyys

Onko teknisen osaamisen ylläpito tärkeää yritykselle?
Antaako sovellus tällä hetkellä kilpailuetua?
Onko todennäköistä palata sovelluksen pariin yrityksen sisällä?
Aiheutuuko sovelluksen pariin palaaminen kustannuksia myöhemmin?
Sisältääkö sovellus kilpailukyvyn kannalta salaista tietoa?
Sisältääkö sovellus itse kehitettyä toiminnallisuutta?
Myytkö tuotetta joka perustuu itse kehitettyyn osaan sovellusta?
Onko organisaation kehityskyky parempi kuin kilpailijoilla?
Saatko tuotteen markkinoille kilpailijoita nopeammin?
Onko ohjelmistosi parempi kuin kilpailijoiden?

Viimeisimpien tutkimusten mukaan (esim. 3T, 2012) tulee edullisten kustannusten maiden massatuotantoon ohjelmistotekniikan alalla suhtautua varauksella. Alkuperäisten kustannusten näyttäessä murto-osalta suomalaisen koodauksen hinnasta, lopulliset kustannukset nousevat helposti kolmannelle suuremmaksi johtuen eri kehitysvaiheissa esiin tulevista virheistä ja niiden aiheuttamasta lisätyöstä. Ulkoistuskumppanin toimintaa tuleekin seurata jatkuvasti - ja jos näyttää siltä ettei tavoiteltuun vaikutukseen päästä, onkin parempi pitää projekti yrityksen omissa käsissä (McConnell, 2002).

3 TUTKIMUKSEN SUORITTAMINEN

3.1. Tutkimuksen sisällön määrittely

Tuotteen kehityksen hallinta keskittyy yhdelle toimipisteelle, josta käsin kokonaisuutta yritetään hallita kokeneen ydinjoukon toimesta. Ohjelmointi-, testaus- ja ylläpitotyötä tapahtuu yrityksen oman toimipisteen henkilöstön lisäksi useamman alihankkijayrityksen toimesta. Samanaikaisen kehityksen alla on kaksi eri tuotetta sekä useampi ohjelmistoversio näistä tuotteista. Tuotekehitystä tapahtuu paikallisen toimipisteen lisäksi eri yhteistyökumppanien kanssa myös muualla Suomessa, Kiinassa, Intiassa sekä Filippiineillä.

Tutkimusosuus voidaan jakaa kahteen pääalueeseen. Ensimmäisenä osiona on kokemusperäinen olosuhteiden nykytilanteen kartoittaminen. Toisena varsinainen ongelmien ja hidasteiden kartoittaminen teemahaastatteluiden muodossa. Tutkimus päätettiin toteuttaa perustuen kokeneen tuotekehitysryhmän osaamiseen ja kokemuksiin toteutuneesta työskentelystä.

3.1.1. Tutkimusongelman määrittely

Yrityksessä on pyritty pitämään kehitysprosessit koko laajuudessaan ketterien kehitysmenetelmien, ja niistä erityisesti tarkemmin Scrumin mukaisena. Tässä on törmätty jatkuvasti erilaisiin ongelmiin, eikä käytäntöjä olla saatu toimimaan yrityksessä. On todettu vastaan tulevan asioita ja tapoja, jotka ovat yrityksessä toteutuessaan ketterien menetelmien periaatteiden vastaisia tai muuten soveltumattomia tarkoitukseensa. Tässä tutkimuksessa yritetään kaivaa tämänkaltaisia tekijöitä esille.

Tästä lähtöasetelmasta ollaan saatu johdettua tiivistetysti tämän työn tutkimusongelma; Mitä ongelmakohtia ja hidasteita tuotekehityksen kokonaisprosessista on löydettävissä.

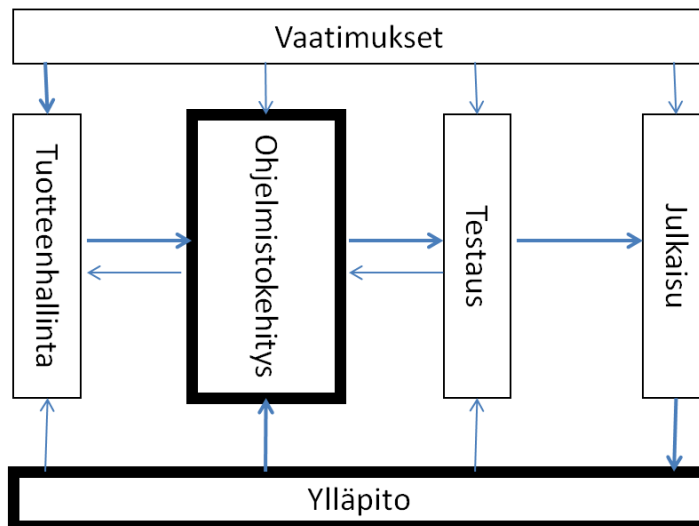
3.1.2. Tutkimuskysymykset

Tutkimusongelman määrittelyn jälkeen voidaan esittää kysymyksiä joihin tämän työn suorittamisen ohessa pyritään vastaamaan.

- Mitä ongelmia nousee toistuvasti esiin työskentelyn yhteydessä?
- Miten voidaan tehostaa päivittäistä työskentelyä?

3.1.3. Tutkimusalueen rajaaminen

Tutkimuksen sisältö on rajoitettu koskemaan vain osaa tuotekehityksen koko alueesta yksittäisen tuotteen kehityksen sisällä. Tuotekehitysprosessi itsessään muodostaa kokonaisuuden, jonka tarkka tutkiminen tässä työssä on mahdotonta toiminnan laajuuden vuoksi.



Kuva 9. Tutkimusalueen rajaaminen

Tutkittava alue rajattiin tässä työssä koskemaan ainoastaan varsinaista ohjelmointityötä sekä siihen kiinteästi liittyvää testausta. Aikaisemman vaiheen tutkimustyö, sekä varsinainen suunnittelutyö jätettiin tutkinnan ulkopuolelle. Myös myöhemmät testausvaiheet jätettiin tässä tarkastelun ulkopuolelle. Lisäksi mukaan on otettu ylläpito jonka sovittaminen yhteen uutta tuottavien resurssien toimintaan on osoittautunut ongelmalliseksi. Käytännössä ylläpidolliset tehtävät tekee sama henkilöstö kuin varsinaisen tuotekehityksenkin.

3.2. Menetelmä 1: Nykytilanteen kartoittaminen

Tutkimuksen yrityksen nykytilannetta kuvaava osuus pohjautuu pääasiassa empiirisiin havaintoihin päivittäisen työskentelyn yhteydessä. Osuudessa on myös huomioitu ns. käytäväkeskusteluissa tai muissa epävirallisissa yhteyksissä käytyjä keskusteluja sekä kommentteja ja mielipiteitä.

3.2.1. Kehitysvaiheet

Viimeisen kymmenen vuoden ajalta on yrityksen toiminnassa voitu erottaa neljä selvästi toisistaan erottuvaa vaihetta toimintaperiaatteiden ja käytäntöjen osalta. Näiden vaiheiden selkeiden muutoskohtien lisäksi eri vaiheiden sisällä on tapahtunut lukuisia erilaisia muutoksia. Pahimmillaan organisaatiomuutoksia ja ohjeistuksien muutoksia on koettu useampia vuosittain.

Vesiputousmalli

Yrityksen tuotannossa noudatettiin mahdollisimman puhdasta vesiputousmallia, virallisesti vuoden 2006 loppuun saakka. Ohjelmoijat, suunnittelijat ja testaajat muodostivat kaikki omat ryhmänsä ja

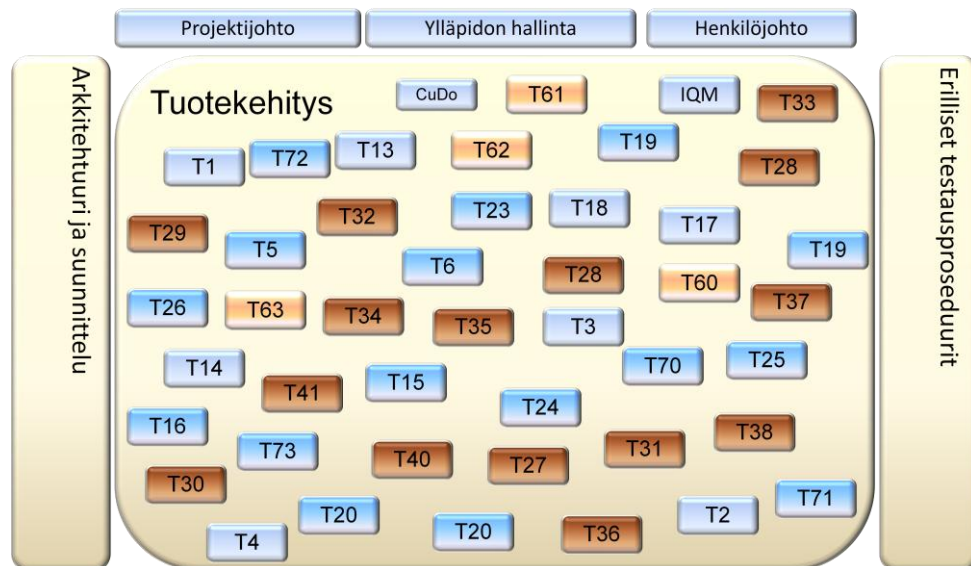
työskentelivät eri osissa toimipistettä. Suunnitteluvaiheen valmistuttua ominaisuuden kehitys siirtyi ohjelmointiryhmälle, joka taas aikanaan siirsi toiminnallisuuden testattavaksi. Henkilöiden välinen kommunikaatio tapahtui pääasiassa muilla tavoin kuin kasvokkain, vaikka senhetkisen tuotteen osalta kaikki toiminta oli keskittynyt yhden toimipisteen yhteen rakennukseen.

Murrosvaihe

Vuoden 2007 aikana yrityksessä testattiin ketterien kehitysmenetelmien soveltuvuutta muodostamalla kokeellisia pienryhmiä jotka kehittivät ominaisuuksia ketterien periaatteiden mukaisesti. Pilottivaihe toteutettiin sijoittamalla valitut ryhmät neuvottelutiloihin jotka toimivat ominaisuuden kehityksen ajan ryhmän työtilana. Ominaisuuden kehittävä ryhmä muodostettiin ottamalla tekijöitä eri henkilöstöryhmistä. Vajaan vuoden kestäneen pilottivaiheen antamien kokemusten perusteella siirryttiin nopealla aikataululla täysin ketterien kehitysmenetelmien mukaiseen toimintaan. Kaikki toiminta oli edelleen keskittynyt yhteen toimipisteeseen.

Agile

Varsinaisen Scrum -muotoisen ketterän kehitysmenetelmän käyttöön yrityksessä siirryttiin vuoden 2008 alussa. Muutos tapahtui samaan aikaan kun vanhoilla menetelmillä kehitetty aikaisempi tuote siirrettiin ylläpitoon. Vanhan tuotteen ylläpito ja jatkokehitys ulkoistettiin myöhemmin alihankkijan hoidettavaksi. Sama muutos toi myös käyttöön uusia ohjelmointikieliä ja toimintatapoja. Tätä kokonaisvaltaista muutosta tehostaakseen yritys laajensi kehittäjäpohjaa kansainvälisesti viemällä osan uudesta toiminnasta ulkopuolisten tahojen kehitettäväksi. Samalla ryhmien koostumus muutettiin sekaryhmiksi sijoittamalla ohjelmistoarkkitehdit, ohjelmoijat ja testaajat samoihin ryhmiin ja tiloihin pysyvästi. Ryhmän sisällä alettiin huolehtia alkuperäistä vaatimusta seuraavasta suunnittelutyöstä, itse ohjelmointityöstä sekä testauksen ensimmäisistä, ohjelmakoodiin kiinteästi liittyvistä vaiheista. Ryhmät olivat keskenään tasa-arvoisia ja ne pyrkivät toimimaan itsenäisesti, yrityksen sisäisesti ohjeistetun Scrum -käytännön mukaisesti.



Kuva 10. Ryhmäorganisaatio ketterässä vaiheessa (Anon 2)

Aluksi sisäiset tuotekehitysresurssit oli jaettu kahteen osaan. Toinen osa kehitti uutta tuotetta uusien, alihankkijoilta saatujen ohjelmistokehitysryhmien kanssa. Toinen osa suoritti osaamisen siirtoa vanhan tuotteen osalta alihankkijalle jonka työntekijät ottivat hiljalleen vastuun kaikesta vanhaan tuotteeseen liittyvästä toiminnasta. Molempien yrityksen omien kehitysryhmien resursseista kului tässä vaiheessa huomattava osa näiden yhteistyökumppanien kehittäjien kouluttamiseen ja tukemiseen.

Vähitellen vanhaan tuotteeseen liittyvät vastuut saatiin siirrettyä alihankkijalle täysin, ja yritykselle jäikin vain ylemmän tason kontrollointi joka ei sitonut kehitysresursseja pieniä poikkeuksia lukuun ottamatta. Kaikki sisäiset resurssit saatiin siten siirrettyä uuden tuotteen pariin.

Hybridi

Keväällä 2012 alettiin yrityksessä herätä siihen, että vuosia yritetty täysin ketteräksi muuttuminen oli osittain epäonnistunut. Tämän seurauksena alettiin työskentelytapoja ja ohjeistuksia laittaa uusiksi, keskittyen niihin kohtiin joissa oli havaittu suurempia ongelmia tai joissa syntyi selkeitä hidasteita. Tärkeimpänä muutoksena tässä nähtäen ryhmäjako ja organisaation rakenne.



Kuva 11. Uusi ryhmäjako Hybridivaiheessa (Anon 2)

Aikaisemmin käytössä olivat olleet itsenäiset ryhmät jotka toteuttivat kehitystyötä Scrum -menetelmään pohjautuvilla työskentelytavoilla. Hybridivaiheessa siirryttiin ominaisuus- tai toiminnallisuuskohtaisiin ryhmiin jotka käyttävät mahdollisimman tehokkaasti suoraan alihankkijoiden ja muiden yhteistyötahojen resursseja. Näin pyritään säilyttämään korkean tason osaaminen ja varsinainen tuotekehityksen hallinta yrityksen omistuksessa, menettämättä kuitenkaan yrityksen ulkopuolisten resurssien muodostamia hyötyjä. Uudistuksen yhteydessä lopetettiin varsinaisten Scrum -käytäntöjen toteuttaminen tuotelinjalla ryhmäkokojen muuttuessa radikaalisti pienemmiksi. Varsinaiset ketterien kehitysmenetelmien mukaiset käytännöt (suunnittelu, demot, iteraatiot, CI yms.) pyrittiin kuitenkin säilyttämään mahdollisuuksien mukaan. Samaan aikaan lisättiin myös linjamanagerien otetta tuotekehityksessä joka oli Scrum-käytäntöjen aikana ollut lähinnä henkilöstöasioiden hoitamista.

3.2.2. Yrityksen toimintaympäristö

Kappaleessa käydään tarkemmin lävitse yrityksen sisäisiä ohjeistuksia niiltä osin kuin ne merkitsevästi poikkeavat Scrum -menetelmän mukaisista ohjeistuksista. Samassa yhteydessä käsitellään myös toimintatapoja joiden toteuttaminen on muuttunut voimakkaasti organisaatiomuutosten seurauksena.

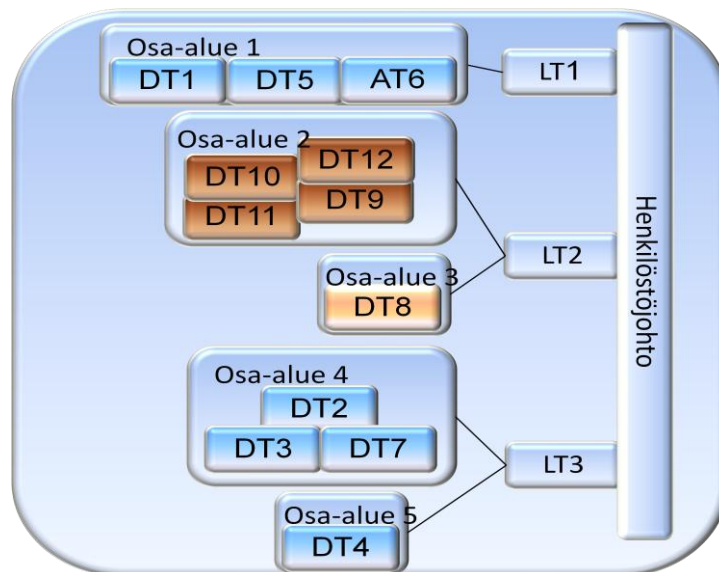
Työympäristö

Suurin yksittäinen muutos yrityksen sisällä on viime aikoina ollut kansainvälistyminen. Kun esimerkiksi vuosituhaten vaihteessa yritys oli käytännössä täysin suomalainen ja yrityksen sisäisenä käyttökielenä oli suomen kieli, on englannin kielestä muodostunut myöhemmin yrityksen virallinen kieli. Tuotekehitys kaikkien eri tuotelinjojen osalta on laajentunut maailmanlaajuisesti. Tämä kehityssuunta sisältää sekä sisäiset toiminnot että ulkoistamiset eri osa-alueilla. Kansainvälistyminen on tuonut mukanaan myös tietoisuutta eri kulttuurien erityispiirteistä, osin sisäisen koulutuksen myötä, osin oppimalla pitkään jatkuneen

vuorovaikutuksen seurauksena. Tämän seurauksena kieliongelmiin on opittu varautumaan.

Ryhmäjakoa tuotelinjan sisällä on viimeisen organisaatiomuutoksen jälkeen viritetty pois päin scrum-menetelmän ohjeistamasta ryhmäkokoonpanosta. Yrityksen sisäiset kehitysryhmät sisältävät edelleen kaikkien erikoisalojen osaajia sekoitettuna, mutta osasta scrum-käytäntöjä on ryhmätasolla luovuttu. Periaatteellinen ohjelmiston kehitystapa on edelleen ketterien menetelmien mukainen ja muodostuu pääasiassa tehtävälistan ympärille. Ryhmät on nyt jaettu osaamis- tai vastuualueiden perusteella neljään eri päävastuualueeseen.

Kunkin pääryhmän alaisuudessa toimii sitten ketteriä kehitysryhmiä useammalta eri alihankkijalta. Tämä paikallinen ryhmä pääasiassa koordinoi kehitysryhmien tekemisiä ja kantaa yleisvastuun osa-alueen toiminnasta. Alla olevassa kuvassa esitetään ryhmien rakenne organisaation sisällä, pääryhmien (LT, Leading Team) kuvatessa tässä yrityksen sisäisiä ryhmiä ja kehitysryhmien (DT, Development Team) niiden alaisuudessa toimivia ketteriä kehitysryhmiä jotka sijaitsevat jonkun toisen toimipisteen tai yrityksen tiloissa.



Kuva 12. Osa-alueiden organisaatiokaavio Hybridivaiheessa

Yrityksen sisäisten ryhmien kokoonpano rakennettiin viimeisen organisaatiomuutoksen yhteydessä mahdollisimman hajanaiseksi, eikä ryhmien kokoonpanoihin ole sen jälkeen juuri puututtu. Tämä stabiili olotila on sallinut ryhmien toiminnan tehostumisen ryhmäytymisen muodossa, vaikka osa ryhmistä onkin menettänyt avainhenkilöitä näiden jättäessä yrityksen erinäisten syiden johdosta. Uusia jäseniä paikallisiin ryhmiin ei juuri ole liittynyt.

Erilaisten yritysjärjestelyjen seurauksena yrityksen omassakin henkilöstössä on eri mantereilta ja eri yrityksistä tulleita henkilöitä. Eri yritysten aikaisemmin käyttämistä työkaluista pyritään pääsemään yhteisiin työkaluihin ja ohjelmistoihin. Käytännössä tässä pyritään samalla

myös pääsemään eroon oman yrityksen ja sen sidosryhmien aikaisemmista määräyksistä ja työkaluista. Tämä jatkuva työkalujen ja ohjelmistojen tuominen käyttöön ja käytöstä poistaminen yhdessä aiemmin ulkoistetun tietoteknisen ylläpidon kanssa on tuottanut ongelmia. Näiden osalta ongelmia ovat tuottaneet yrityksen heikosti toimiva sisäverkko ja välillä toimimattomat ohjelmistot.

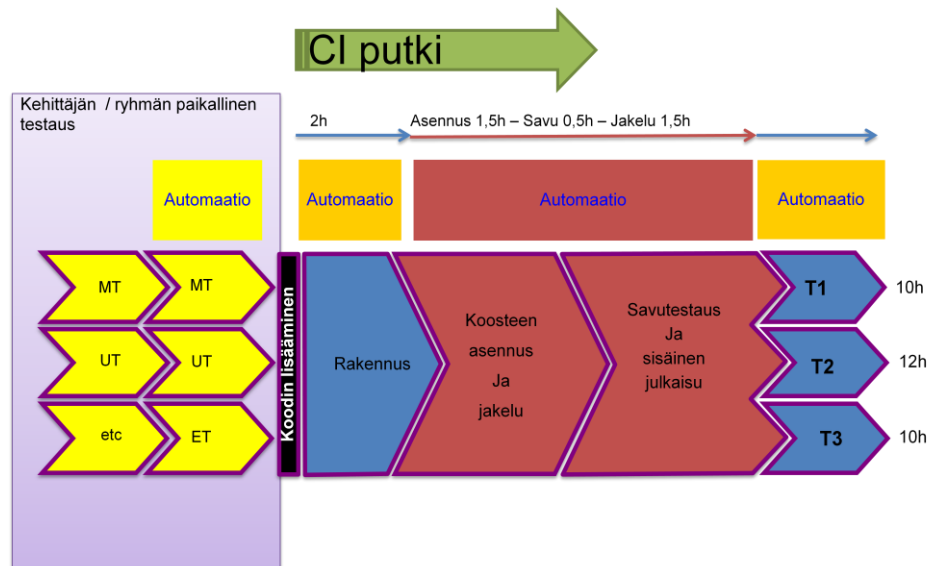
Tietotekninen laitteisto on pyritty pitämään käyttökuntoisena vaikka lähes kaikista kustannuksista on tingittävä. Käytännössä tuotannon ja laadunvarmistuksen kannalta kriittiset laitteistot pidetään kunnossa, mutta henkilöstön käytössä olevat tietotekniset laitteet eivät suurilta osin vastaa nykypäivän vaatimuksia. Laitteita uusitaan ainoastaan entisten hajotessa.

Laatu tutkimuksen aikana

Uuden tuotteen ensimmäistä versiota tuotettaessa otettiin tietoisesti teknistä velkaa aikataulukriittisistä syistä. Nyt, kaksi vuotta myöhemmin tuota velkaa maksetaan edelleen. Vanhan tuotteen osalta ulkoistuksen yhteydessä tapahtunut kouluttaminen ja sen jälkeinen ulkoistuskumppanin vaatima käytännön tuki ylittivät kaikki ennakoarviot, varaten suuren osan käytössä olevista resursseista joko osin tai kokonaan koska vanhan tuotteen kaupallinen vastuu oli kuitenkin yrityksellä itsellään.

Ylläpidon osalta seurataan jatkuvasti eri julkaisujen vikamääriä. Tilanne tiedotetaan päivittäin koko osastolle sähköisenä tiedotteena, jossa myös verrataan todellista vikamäärää julkaisun rajapyykkien edellyttämiin rajoihin. Lisäksi laatuorganisaation toimesta kerätään laajempia tilastoja vioista, niiden syistä ja korjausajoista. Viime aikoina toimenpiteisiin on myös lisätty Weibull-analyysi vikamäärien kehityksestä, ja tämän analyysin tarjoamat skenaariot osoittavat suuria ongelmia koodin toiminnallisen laadun kannalta.

Ketterien kehitysmenetelmien mukaisesti koodia lisätään kokonaisuuteen pala kerrallaan, jonka jälkeen automaattiset testausjärjestelmät kääntävät koodit ja suorittavat pikatestejä joista saadaan kehittäjille palaute minuuttien sisällä. Jos koodissa on perustavanlaatuisia ongelmia, ne yleensä jäävät kiinni jo tässä vaiheessa. Jokaisen koodin lisäämisen jälkeen alkaa myös kokonaiskoodin kääntäminen ja sisäisen ohjelmistojulkaisun valmistelu SCM:n toimesta. Tälle ohjelmistopakettile ajetaan automaattisen Continuous Integration -järjestelmän kautta yölliset testiajot, joiden laajuus riittää varmentamaan toiminnallisuudet. Yöllisten testiajojen tuloksista kehittäjät analysoivat seuraavana päivänä toimimattomat osa-alueet ja korjaavat ne mahdollisuuksien mukaan välittömästi. Testiautomaatiota lisätään yrityksessä jatkuvasti eri tasoilla, ja sen kattavuutta pyritään parantamaan lisäämällä erilaisia mittareita.



Kuva 13. CI putken yleiskuva

Julkaisu ja ylläpito

Tuotteen varsinaiseen ylläpitoon ei oteta käytetyissä teorialleissa kantaa, vaan ylläpidon osalta ollaan yrityksen sisäisten toimintamallien ja ohjeistuksien varassa. Yleisesti ottaen tuotteen julkaisun saavutettua tietyn valmiustason, se siirtyy kehitystilasta ylläpitomoodiin. Tämä valmiuspiste muodostaa samalla rajan sille, miten tuotteesta löytynyttä vikaa käsitellään, ja mitä kaikkia toimenpiteitä sen korjaaminen vaatii.

Kehitysvaihe

Kehitysvaiheessa havaitut viat ja ongelmat viedään mahdollisimman nopeasti tiedoksi asianosaisille. Näitä voivat olla ryhmä, joka vastaa ominaisuudesta johon vika liittyy, tai ryhmä jonka erikoisosaamiseen kyseinen laajempi toiminnallisuus kuuluu. Viasta voidaan myös kirjoittaa ongelmaraportti, jos sen korjaamiseen tarvitaan enemmän resursseja eikä välitön korjaaminen ole mahdollista. Ryhmä, jonka tehtäväksi korjaaminen loppujen lopuksi tulee, vastaa kaikista asiaan kuuluvista toimenpiteistä. Näitä ovat mm. mahdollinen uudelleensuunnittelu, koodimuutosten toteuttaminen, eri testausvaiheet sekä toisten, rinnakkaisten ohjelmistoversioiden tarkistaminen saman vian varalta.

Ylläpitovaihe

Tuotteen saavutettua määrätyn kypsyyssasteen, siirrytään kehitysvaiheesta ylläpitovaiheeseen. Kriteerit tälle siirtymälle ovat:

- Ei kriittisiä vikoja avoinna
- Toiminnallisuustestaus suoritettu onnistuneesti
- End-to-End testaus suoritettu onnistuneesti

Tässä vaiheessa ohjelmiston testaaminen jatkuu laajemmissa testausympäristöissä ja eri testausvaiheissa. Asiakkaat saattavat kokeilla ohjelmistoa tässä vaiheessa. Uusia ominaisuuksia tuotteeseen ei enää tehdä tässä vaiheessa mutta vikakorjaukset jatkuvat. Kaikista valmiuspisteen jälkeen löydetystä vioista pidetään kirjaa virnehallinnan työkalussa. Lisäksi kaikille korjatuille ja toimiviksi testatuille löydöksille laaditaan muutosmuistio, joka liitetään myöhemmin asiakkaille toimitettavaan dokumentaatioon. Tarkempi kuvaus ylläpidon toiminnallisesta proseduurista löytyy liitteestä 3.

Kun virhe tai löydös on kirjattu työkaluun, sen jälkeen määrätty ylläpitoryhmä ottaa muutoksen koordinoinnin hallintaansa. Osan vioista tämä ryhmä voi korjata itse, osan se välittää mahdollisuuksien mukaan sille kehitysryhmälle joka on kyseisen osa-alueen paras tuntija, tai ryhmälle jolla on muuten mahdollisuudet ratkaista ongelma. Näin ollen sama henkilöstö kehittää uutta toiminnallisuutta sekä korjaa vanhojen julkaisujen vikoja. Käytännössä tämä tapahtuu kahden tai useamman eri projektin alaisuudessa, joilla on omat projektipäällikkönsä. Lisäksi on havaittu, että yhden ohjelmistovirheen korjaamisen vaatima byrokratia vian korjaavaa ohjelmistokehittäjää kohti on moninkertaistunut vuosien kuluessa.

Johtamiskäytännöt yrityksessä

Kehittämistapojen, menetelmien ja ohjeistusten eläessä vuosien aikana yksi asia yrityksessä on pysynyt vakiona. Tämä on erilaisten managerien ja johtajien lukumäärä tuotekehitysorganisaatiossa. Roolit ja johtamisnimikkeet ovat vaihdelleet käytössä olevien työskentelyn ohjeistusten mukana. Kuitenkin esimerkiksi scrum -menetelmän teoreettisessa ohjeistuksessa (Larman & Vodde, 2008) kehoitetaan ohentamaan organisaatiota mahdollisimman ohueksi ja luopumaan eri työnimikkeistä.

Käytännössä eri projektien (ylläpito, kehitys1, kehitys2, etc.) päälliköt ovat pyrkineet ohjaamaan kehitysresursseja oman projektinsa kannalta edullisimpaan suuntaan, samalla käytännössä väheksyen muiden projektien tarpeita. Tästä johtuen ryhmillä onkin välillä ollut ongelmia saamiensa tehtävien tärkeyden ja toimintajärjestysten kanssa. Pahimmillaan saman ryhmän jäsenet ovat olleet tekemässä kahta eri tuotetta, 3-4 eri managerin alaisuudessa, joista jokaisella on oma käsityksensä ryhmän tehtävien tärkeydestä. Lisäksi osa johtajistosta kommunikoi tuotekehityksen kanssa lähestulkoon ainoastaan sähköpostin ja sähköisten ohjeiden välityksellä sen sijaan että otettaisiin modernit, oikeasti tehokkaat kommunikaation välineet ja mahdollisuudet käyttöön.

3.3. Menetelmä 2: Haastattelututkimus

Yrityksen sisällä tehtävä tutkimus- ja kehitysprojekti (haastattelututkimus) toteutettiin seuraavalla tapaa vaiheistetusti. Jokaiselle osa-alueelle oli ennalta määrätty ajankohtansa jonka aikana osa haluttiin tehtäväksi.

- Tutkimuksen tavoitteen määrittely (Q4/2011)
- Tutkimuksen aiheet (Q1/2012)
- Haastateltavien valinta (Q1/2012)
- Haastattelut (Q2/2012)
- Haastattelujen tuloksien kirjaaminen (Q2/2012)
- Suppea yhteenveto (Q3/2012)
- Laajempi analysointi (Q3-Q4/2012)
- Tulosten julkistaminen (Q4/2012)

Sisällön listaamisen ja hyväksymisen jälkeen laadittiin alustava aikataulu haastatteluille ja valittiin haastateltava kohderyhmä. Ryhmän koostumukseksi saatiin laaja jakauma eri työrooleista. Edustus tutkimukseen saatiin ohjelmistokehittäjistä, testaajista, tuotteen johdosta sekä projektien johdosta. Tavoitteena oli haastatella puolet ohjelmointi- tai testaustyötä tekeviä, puolet erilaisissa johtotehtävissä olevia. Tässä vaiheessa havaittiin että lähes kaikki haastatteluihin valitut olivat erittäin halukkaita kehittämään työympäristöään, antamaan palautetta sekä vaikuttamaan työympäristönsä asioihin. Tämä oli yhteistä kaikille välittämättä siitä missä työtehtävissä kyseinen henkilö toimi.

3.3.1. Haastattelujen aihealueet

Tutkimuksen pohjana päädyttiin käyttämään sekä yrityksen sisäistä ohjeistusta erilaisten resurssien käytöstä sekä ohjeistettuja työmenetelmiä. Näitä seikkoja ja niiden käytännön toteutumista peilattiin sekä scrum-menetelmän teoreettisessa mallissa ohjeistettuihin toimintatapoihin, että yleisesti ohjelmistotuotannon hallinnassa tunnustettuihin periaatteisiin.

Aihepiirilistan laatiminen aloitettiin hyvissä ajoin keväällä 2012. Teoriapohjaan tutustumisen rinnalla kokoon kutsuttiin työryhmä varmistamaan tutkimuksen kattavuus. Tähän työryhmään kutsuttiin useamman eri ohjelmistotuotteen edustajat laatuorganisaatiosta sekä prosessipuolen edustaja. Valitut henkilöt kutsuttiin tapaamiseen, jossa laadittiin alustava listaus aiheista joita osanottajien mielestä oli hyvä selvittää. Osasta esiin tulleita ajatuksia luovuttiin keskustelun päätteeksi, ja osa tarkentui ryhmän toimesta. Tilaisuuden yhteydessä esiin tulleet ajatukset kirjattiin yhteen muistioksi joka jaettiin osanottajille. Kolme viikkoa myöhemmin ryhmä kokoontui uudelleen työryhmäksi, joka tuotti aikaisemmasta lähtömateriaalista lopullisen listauksen. Tutkimuksen tilaajan toimesta lisättiin tässä vaiheessa vielä vertaileva kysymys eri työohjeistuksien hyvistä ja huonoista puolista.

Haastattelujen alussa kerättiin hieman taustatietoa työroolien ja päätehtävien muodossa. Näitä taustoja käytettiin apuna vastausten suhteuttamisessa kokonaisuuteen, mutta varsinaisista tuloksista ne

häivytettiin haastateltavien anonyymiteetin nimissä. Varsinaiseksi aiheluetteloksi muotoutui tapaamisten, ja niiden ulkopuolella kerättyjen kommenttien perusteella oheisen kaltainen luettelo. Aiheita ei haluttu määritellä liian tarkasti, vaan jokainen haastateltava sai itse muodostaa käsityksensä mitä aihe hänen osaltaan tarkoittaa. Tällä avoimuudella ratkaistiin eri työroolien edustajien näkemys eri aiheisiin.

- Yhteinen ymmärtämys
- Ryhmien kehitys ja dynamiikka
- Muutokset organisaatiossa, työympäristössä ja työn ohjeistuksessa
- Resurssien kohdentaminen
- Kommunikaatio
- Hukkaresurssit
- Työkalut, ohjelmistot ja laitteet
- Tuote
- Kompetenssit
- Riippuvuuksien hallinta
- Demokäytännöt
- Tekninen velka
- Johtaminen
- Alihankkijat
- Asiakashyödyt
- Nykyisten ja aikaisempien työohjeistuksien hyvät ja huonot puolet
- Pahimmat esteet ja hidasteet tällä hetkellä

+Vapaat kommentit

Listattu järjestys havaittiin käytännössä niin toimivaksi, että suurimmassa osassa haastattelutilanteista kohdehenkilö käänsi omatoimisesti puheenaiheen seuraavaan kohtaan listalla. Muutamien henkilöiden kohdalla jouduttiin mainitsemaan esimerkkitapauksia tai muuten esittämään pieni johdanto haluttuun aihealueeseen ennen varsinaisten kommenttien saamista, mutta enemmistön osalta riitti lyhyt avaus aihepiiriin pariin.

3.3.2. Haastattelut tapahtumina

Seuraavana työvaiheena oli suorittaa varsinainen tutkimustyö kohdistettuina teemahaastatteluina, joilla pyrittiin selvittämään erilaisten työroolien näkemys aiheeseen. Samalla tutustuttiin erilaisten työroolien näkökulmasta prosessin toimimattomiin osiin. Eri kohdehenkilöiden tarkempi asiantuntemus eri osista kokonaisprosessia nousi tässä voimakkaasti esiin. Haastattelun kohteiksi haettiin sekä toimintaa ohjeistavia ja johtavia tahoja, että varsinaista ohjelmointi- tai testaustyötä tekeviä henkilöitä. Haastateltavien ryhmän kooksi muodostui 14 henkilöä, seuraavanlaisella jakaumalla:

- 3 ohjelmistokehityksen linjaesimiestä
- 3 tuote- tai projektitason esimiestehtävissä
- 7 ohjelmistokehityksen asiantuntijatehtävistä
- 1 laatuorganisaation edustaja

Ohjelmistokehityksen asiantuntijatehtävistä valittu kohderyhmä jakautui kolmeen aliryhmään. Nämä olivat ohjelmointi, testaaminen ja oheistukitoimintojen tehtävät. Tulosten käsittelyssä näitä kuitenkin käsiteltiin yhtenä kokonaisryhmänä tehtävien tiukan yhteenkuuluvuuden vuoksi. Ryhmän jakauma havaittiin sopivaksi tutkimuksen tarpeisiin. Ensi vaiheessa kutsut lähetettiin 12 henkilölle, joista 11 suostui haastatteluun saman tien. Yhden kieltäytyneen jälkeen ryhmää päätettiin laajentaa hieman, jonka myötä sitten päädyttiin lopulliseen kokoonpanoon.

Haastattelutilaisuuksiin oli varattu kullekin haastateltavalle tunti aikaa, jonka lisäksi tilaisuuden alussa selvitettiin haastattelun aihealue, sisältö, materiaalin käyttötarkoitus sekä tiedotettiin luottamuksellisuudesta ja hyväksytettiin keskustelun tallentaminen.

Haastattelutilaisuuden lopuksi haastateltavalle tiedotettiin jatkosta. Tämän jälkeen haastattelutallenteet muutettiin kirjalliseen muotoon. Tässä vaiheessa kieliasua siivottiin mahdollisesti hieman, mutta varsinaiset kommentit kirjattiin sanasta sanaan. Kirjallinen tuotos lähetettiin sähköisessä muodossa haastateltavalle hyväksyttäväksi. Samassa yhteydessä tarjottiin myös haastatelluille mahdollisuus muokata tai vetää takaisin mielipiteitään, mutta kukaan ei tarttunut tähän tilaisuuteen. Hyväksynnän jälkeen tekstimuotoinen tallenne arkistoitiiin odottamaan jatkokäsittelyä.

3.3.3. Tulosten käsittely

Kun kaikki haastattelut oli saatu kirjalliseen muotoon ja hyväksytettyä, alkoi tulosten yhteenveto nopeaa raportointia varten. Tässä kerättiin aihepiireittäin kommentit yhteen ja tiivistettiin asia mahdollisimman ytimekkääksi. Nämä yhteenvedot toimitettiin välittömästi niiden valmistuttua työn tilaajalle sekä laatuorganisaation edustajalle nähtäväksi.

Tulosten laajempi analysointi ja pohdinta tapahtui Q3-Q4 aikana, ja tässä yhteydessä pyrittiin myös löytämään tarkempia syitä mahdollisten haastatteluissa esiin tuotujen oireiden taustalla. Selkeistä löydöksistä tiedotettiin myös välittömästi eteenpäin yrityksen sisällä kuukausittaisen seurantalaverin yhteydessä. Näissä palaverissa myös pohdittiin mahdollisia jatkotoimenpiteitä, mutta mitään varsinaista kehitysprojektia asiasta ei aloitettu vaan keskityttiin hakemaan tapoja ja uusia näkökulmia tutkimuksen loppuun saattamiseksi.

4 TULOKSET

Tutkimuksen lähdeaineisto voidaan jakaa kahteen osaan, teoreettiseen aineistoon sekä kokemuspohjaiseen aineistoon. Teoreettinen aineisto perustuu yrityksessä käytettyihin kirjallisiin toimintaohjeisiin sekä tämän taustana olleeseen kirjalliseen materiaaliin. Kokemuspohjainen aineisto kerättiin kohdennetuilla haastatteluilla yrityksen henkilöstön keskuudesta.

4.1. Tulosten käsittely

Tulosten käsittelystä oli jo alusta alkaen sovittu tiettyjä käytäntöjä. Työn etenemistä seurattiin kuukausittaisessa palaverissa, jossa myös vastaan tulleet epäkohdat tuotiin välittömästi työnantajan edustalle tiedoksi. Suuremmista kokonaisuuksista laadittiin kirjallisia koosteita, jotka toimitettiin työnantajan edustajan lisäksi työn tilaajalle.

4.2. Haastattelusuuden tulosten yhteenveto

Haastatteluista ja niiden sisällöstä esitellään tässä kappaleessa alkuperäisten aihepiirien mukainen yhteenveto.

4.2.1. Yhteinen ymmärrys

Aihepiirin kartoittamisella pyrittiin saamaan selko siitä, onko yrityksen toiminnan eri työvaiheiden edustajilla oikeasti yhteinen ymmärrys siitä mitä tehdään ja mitä halutaan tehdä.

Haastattelujen aikana havaittiin ettei henkilöstö oikeasti aina tiedä puhutaanko samoista asioista. Näkökulmien ero ohjelmistokehittäjien ja esimiesportaan välillä havaittiin suureksi. Kehittäjät kaipasivat yhteisen näkökulman kehittymistä, kun taas esimiestasolla suurin huoli oli asioiden ymmärtämyksen varmistaminen. Yhteistä kaikille vastaajaryhmille oli teknisen näkökulman ja johtamisen yhteentörmäys - millä päätökset perustellaan.

Suurimmaksi ongelmaksi yhteisen ymmärtämyksen osalta nousi kuitenkin alihankkijoiden toiminta. Kulttuuriset erot nousivat esille, sekä yleisesti hierarkkisen ja luokka-ajatteluun perustuvan maailmankuvan osalta että työkuulttuurin ja luottamuksen kehittymisen osalta. Yrityksen omien pitkäaikaisten työntekijöiden osalta ei kyseisiä ongelmia havaittu. Ongelmien osalta havaittiin taipumusta ratkaista ne helpoimmalla ja nopeimmalla mahdollisella, usein tilapäisellä tavalla. Tämä lähestymistapa valittiin usein huolimatta siitä että asian tiedettiin tulevan uudelleen vastaan ennemmin tai myöhemmin.

4.2.2. Ryhmät

Ryhmien, ryhmädynamiikan ja sisäisten toimintojen esiin nostamisella haluttiin kartoittaa yrityksessä lähes jatkuvasti tapahtuvien

organisatoristen muutosten sekä ryhmien ohjeistettujen toimintatapojen vaikutusta ryhmien toimintaan ja tehokkuuteen.

Ryhmien kokoonpanojen seurannan osalta havaittiin puutteita. Ryhmät muodostettiin tehokkaasti, ja niiden toiminta käynnistettiin hyvin. Pääasiallisesti ryhmät saivat toimia pitkään yhdellä kokoonpanolla, jolla vältettiin ryhmädynamiikan ongelmia. Tällä muutosten vähäisyydellä on myös kääntöpuolensa. Tiimien kokoonpanon muutoksien vaikutusta ryhmän työtehoon ei huomioida ja toiminnan kontrolli havaittiin vähäiseksi. Jos ryhmä ei toimi tai kehity, sen kokoonpanoa pitäisi kyetä muuttamaan nopeasti, välttämättä kuitenkin ylimääräisiä muutoksia muutosten vuoksi. Mahdollisten muutosten tapahtuessa työtehtävissä, ryhmässä tai kohdealueessa ei yleisesti huomioida henkilökohtaisia vahvuuksia ja osaamisalueita.

Suurimpina ongelmina ryhmien toiminnassa esiin nousivatkin seurannan puutteen ohella se, ettei ryhmien kuitenkaan anneta toimia itseohjautuvasti, vaan toimintaa pyritään kontrolloimaan ulkopuolelta. Myös se, ettei ryhmien kokoonpanoja ole välttämättä tarkistettu tai muutettu henkilöstömuutosten jälkeen. Nimellisesti scrumia toteuttava ryhmä jonka scrum master on lähtenyt kuukausia aikaisemmin yrityksestä, ja jonka jokainen jäsen joutui jakamaan aikansa usean eri projektin vaatimusten välillä ei voi toimia tehokkaasti. Lisäksi havaittiin kaikilla tasoilla se, ettei yrityksessä toimita ryhmien kanssa vaan asioiden annettiin henkilöityä.

4.2.3. Muutokset

Muutosten esiin nostamisella aihepiirinä haluttiin seurata jatkuvien toimintatapojen, ohjeiden ja toimintaympäristön muutosten vaikutusta toiminnan yleiseen tehokkuuteen.

Alihankinnan ja työryhmien määrän lisäämisen seurauksena yhden henkilön toimialueeksi muodostuu pieni osa suuresta kokonaisuudesta. Alihankinnan lisääminen tärkeässä tuotteessa on muodostunut suureksi riskiksi jonka kontrollointi on vaikeaa. Tässä havaittiin suuria eroja sen osalta kuinka kokonaiskuva tuotteesta ja sen toiminnallisuudesta näkyy. Kokonaisuuden hallinta on täysin työntekijän oman aktiivisuuden varassa. Pahimmillaan työntekijällä ei ole mitään käsitystä valmiin tuotteen toiminnasta vaan ohjelmointi tapahtuu tarkasti määritellyn kirjallisen ohjeen mukaan. Teoriassa tämän pitäisi riittää, mutta käytäntö on osoittanut että hyvin toimiva ohjelmakoodi vaatii myös tuotetietämystä. Samoin toiminnallisen kokonaiskuvan hämärtyessä työtehtävien kohdentuminen voi heikentyä tai kadota kokonaan koska eri koodin osien vaikutusta koko tuotteen toiminnallisuuteen ei kyetä enää arvioimaan.

Muutosten myötä syntyy myös hukkaresursseja koska saatetaan tehdä turhaa työtä ja joudutaan opettelemaan uusia toimintatapoja. Lisäksi muutosten seurauksena aikaisemmin erillisiä työrooleja on yhdistetty kuitenkin vähentämättä niiden vaatimaa byrokratiaa. Muutosten ja uusien vaatimusten myötä on noussut esiin ilmiö jossa vastuun ottamista

saatetaan vältellä. Koetaan, että yrittämisestä rangaistaan palkitsemisen sijaan. Rangaistusluontoisena koettiin joissain tapauksissa itselle tulevan ylimääräisen työn määrä mahdollisten löydösten tai hyödyllistenkin ideointien seurauksena. Tämän seurauksena toiminnan puutteita tai suoranaisia vikoja saatetaan jättää ilmoittamatta. Näin voidaan todeta kaikkien suurempien muutosten haittaavan kilpailussa pärjäämistä - vaikutuksen keston vaihdellessa tapauskohtaisesti.

4.2.4. Resurssien kohdentaminen

Resurssien kohdentamisessa haluttiin tutkia mihin käytössä olevat resurssit suunnataan. Osittain jatkuvien organisaatiomuutosten, YT-neuvottelujen ja ydinliiketoiminnan ulkopuolisten toimintojen leikkaamisesta johtuen yleinen käsitys tuntuu olevan että aina joku taso hierarkkisessa yritysrakenteessa on hukassa sen kanssa mihin tulee keskittyä.

Toiminnan johdonmukaisuutta kaivataan myös yleisluontoisesti, tärkeimpänä havaintona esimiestason toiminnan epäloogisuudet. Ryhmä tai henkilö saa tehtävänantonsa useammalta taholta, eivätkä nämä tavoitteet välttämättä tue toisiaan. Tehtävien jaossa ei välttämättä noudateta yhdessä sovittuja ja käytännön muokkaamia pelisääntöjä jolloin asiasta tai osasta vastaava taho ei välttämättä saa tarvitsemiaan asioita tietoonsa lainkaan. Lisäksi tulee huomioda jokaisen resurssien kohdentamisen muutoksen aiheuttamat viiveet.

4.2.5. Kommunikaatio

Kommunikaatio aihepiirinä tiedettiin jo ennakolta ongelmalliseksi, mutta päätettiin kuitenkin ottaa mukaan omaksi kohdealueekseen jotta saataisiin tarkempaa tietoa kipupisteiden sijainnista. Aihepiiriin sisällytettiin sekä sähköinen että perinteisempi yhteydenpito. Kansainvälisesti hajautuneessa toimintaympäristössä luonnollisesti sähköisen viestinnän kanavat osoittautuivat kriittisen tärkeiksi. Erilaisia sähköisiä viestintäkanavia (erityyppiset verkkosivut, pikaviestimet ja sähköpostit) pyritäänkin hyödyntämään yrityksessä mahdollisimman paljon.

Alihankkijoiden ja eri kulttuuritaustaisten toimipisteiden osalta havaittiin useampi tärkeä asia. Käytännössä joissain tehtävissä vaadittiin tarkat kirjalliset ohjeet kaikesta tehtävästä. Tämän yhteydessä havaittiin että paikallisia resursseja käytettiin joskus tehtäviin jotka olisi pitänyt tehdä ulkopuolella ja joiden tekemisestä maksettiin ulkopuoliselle tekijälle. Lisäksi havaittiin usean ulkopuolisen tahon vaativan reaaliaikaista valvontaa ja kontrollointia paikalliselta tasolta, mutta aikaerojen vuoksi tämä tapahtui luonnollisesti muilla kuin reaaliaikaisilla tavoilla. Tästä aiheutuu myös viiveitä. Suureen rooliin nousivat myös kieliongelmat molempiin suuntiin, eli saavuttaako haluttu ajatusmalli myös kohderyhmän.

Sähköinen viestintä aiheuttaa myös paikallisella tasolla ongelmia. Yleisenä havaintona onkin, että sähköpostia tulee lähes kaikille aivan liikaa, ja oleellisen tärkeätkin asiat voivat kadota massan mukana aiheuttaen joskus pahojakin viiveitä viestin sanoman perille menoon. Kommunikointi on siirtynyt sähköiseen suuntaan perinteisen puheyhteyden kustannuksella. Pääasiassa tämä ei aiheuta ongelmia, mutta joissain tilanteissa olisi kannattavampaa uhrata hetki aikaa ja järjestää tapaaminen jossa haluttu asia käsiteltäisiin saman tien pois odottamasta sen sijaan että viestiä pyöritellään eri osapuolten välillä useita päiviä tai pahimmillaan viikkoja.

Sähköisessä muodossa liikkuu myös suurin osa kaikista tiedotteista. Yleinen toive kohderyhmässä olikin, että tiedotusasioiden painopistettä siirrettäisiin enemmän kasvokkaisen kommunikoinnin puolelle jolloin mahdollistettaisiin oikea, kaksisuuntainen vuoropuhelu. Nyt jäävät päätösten hahmottamista avustavat taustatiedot ja perusteet yleensä kertomatta.

Lähes yhtä pahana ongelmana nähtiin kommunikaation henkilöityminen. Sama ilmiö toistui kaikissa eri haastatelluissa kohderyhmissä, tietyillä asioilla on aina samat yhteyshenkilöt vaikka samaa asiaa käsittelee kokonainen ryhmä. Tästä seuraa epäsuorasti pahojakin viiveitä jos kaivattu yhteys ei olekaan saavutettavissa vaan täytyy lähteä etsimään korvaavaa viestintäyhteyttä. Samoin työtehtävien ja resurssien kohdentamisessa tapahtuvien muutosten myötä tämänkaltaiset suorat kontaktit vaihtuvat joka myös aiheuttaa viivettä myöhemmissä vaiheissa. Myös tehtävästä toiseen siirryttäessä kaivattiin kattavampaa perehdytystä ja opastusta.

Esimiestason kommunikointia kritisoitiin lähes kaikilla tasoilla. Yleisenä havaintona oli, ettei hierarkkisesti liikkuvaa tietoa toimiteta perille samanaikaisesti vaan eri ryhmien välillä voi olla vuorokausienkin eroja. Esimiestason kommunikointiin kaivattiin myös muutosta siinä että yleensäkin saataisiin vastauksia yhteydenottoihin. Sähköposteihin ei vastata, tai tapaamisiin ei saavuta. Yleensä tämä oli kunnossa kunkin lähimmän esimiehen kohdalla, mutta muiden kanssa ongelmia oli laajemmin. Ainoaksi luotettavaksi tavaksi nimettiinkin suora keskustelu, johon toisaalta on ongelmallista viitata myöhemmin tarpeen ilmetessä jos mahdollisia ohjeita tai tietoja ei kirjoita samaan aikaan muistiin.

Lisäksi käytännössä voimassa oleva pyrkimys syyllistää ja etsiä syytettävää sen sijaan että keskityttäisiin ensisijaisesti korjaamaan tilanne on aiheuttanut yleisesti tilanteen jossa realistinen kommunikaatio ei toteudu kaikkien osapuolten kanssa. Asioita saatetaan kaunistella tai suorastaan valehdella suuntaan tai toiseen, ja esimiehelle saatetaankin kertoa täysin eri asiat kuin läheiselle työtoverille samasta aiheesta.

4.2.6. Hukkaresurssit

Oleellisena kohtana hidasteiden kartoittamista on turhaan tehty työ tai yleisemmin, hukkaresurssit. Osuudessa pyrittiin löytämään osa-alueet,

joissa tehostamisen varaa olisi eniten. Myös kehitysprosessin sisäisiä työvaiheita kartoitettiin tavoitteena löytää mahdollisuuksia lisätä automaatiota tai muuten siirtää manuaalista työtä koneiden tai ohjelmien tehtäväksi. Kaksi kohdetta nousi haastatteluissa yli muiden, sekä esimiestason henkilöstön että tuotekehityksen tekijöiden kannalta. Nämä olivat yleisellä tasolla ihmisten käyttäytyminen ja työkalujen toiminta.

Ihmisten kohdalla havaintona olivat se, ettei yksittäisten henkilöiden varsinaista osaamista ole kartoitettu riittävän tarkasti ja siten osaamista joutuu joskus etsimään. Toinen aiheeseen liittyvä ongelma ja hidaste on välillä havaittavissa oleva vastuun välttely. Tehtäviä voidaan viivästyttää toivoen että joku muu ottaisi sen tehtäväkseen, havaituista asioista pysytään hiljaa kun yritetään välttää aiheesta aiheutuvaa työtä. Lisäksi havaittiin ettei kaikkia asioita yksinkertaisesti pysty viemään eteenpäin koska niihin ei suostuta ottamaan kantaa muiden taholta. Yleistää tätä ei tietenkään voi, vaan kyseessä on yksilökohtainen vaihtelu. Sama tilanne on oma-aloitteisuuden kanssa, joidenkin tiedetään toimivan vain käskyn tai työpyynnön saatuaan. Tämä ilmiö on yleisempi alihankkijoiden parissa mutta tuttua myös paikallisella tasolla. Myös töiden keskittyminen vaikuttaa asiaan; uusia asioita ei välttämättä opetella riittävällä tasolla ja muutama todellinen osaaja ovat sen seurauksena ylityöllistettyjä. Samoin hajautuneesta työympäristöstä johtuen osaaminen ja tekeminen ovat pirstoutuneet eri osastoille ja eri toimipisteille jolloin aikaa hukkuu paljon asioiden synkronointiin ja yleiseen kommunikaatioon osapuolten välillä. Pahimmillaan osapuolet saavat keskustelua eteenpäin yhden sähköpostin verran vuorokauden aikana johtuen aikavyöhykkeistä ja erilaisista työajoista. Myös alihankkijoiden ja ulkoistusten lisääminen ja vaihtaminen kesken projektin aiheuttaa suuria viiveitä vaadittavan byrokratian, koulutusten ja tekemällä oppimisen myötä.

Työkalujen ja ohjelmistojen osalta löytyi myös kritisoitavaa. Vaikka käytössä oleviin työkaluihin ja ohjelmistoihin oltiinkin pääasiassa tyytyväisiä, varsinkin henkilöstöasioihin liittyvät ohjelmistot aiheuttivat pahoja ongelmia. Verkon ylitse käytettävät työkalut ovat usein alhaalla tai toimivat verraten hitaasti, jolloin yksittäinen henkilö saa asiansa hoidettua muutamien minuuttien odottamisella. Suuressa yrityksessä kun muutama minuutti kertaautuu työntekijöiden määrällä, ei kokonaissumma olekaan enää niin yhdentekevä jolloin voidaan puhua jo jättimäisestä hukkaresurssista. Myös ulkoistettu IT-tuki aiheuttaa viiveitä - työntekijät käyttävät mieluummin aikansa etsiäkseen itse ratkaisun mahdolliseen ongelmaan kuin hakevat apua siihen tarkoitettuun organisaatiolta.

Erilaisia työvaiheita on jatkuvasti pyritty automatisoimaan mahdollisuuksien mukaan. Laadunparantamisen nimissä on lisätty erilaisia analysaattoreita automatiikan ajettavaksi, mutta näitä analysaattoreita ja niiden tekemiä analyysejä ei hyödynnetä siinä määrin kuin olisi mahdollista. Vakavana hukkaresurssin synnyttäjänä mainittiin päällekkäinen testaus jossa useampi taho testaa samaa koodia ja toiminnallisuutta pahimmillaan samoilla tai lähes samoilla testitapauksilla. Myös pitkän aikavälin suunnittelun epäonnistumiset nostettiin esiin. Tuolloin useampi ryhmä saattaa ohjelmoida jotain kokonaisuutta, joskus

jopa viikkoja, ennen kuin kyseinen julkaisu perutaan tai sisältö muuttuu ja todetaan viikot käytetyksi turhaan työhön.

4.2.7. Työkalut

Käytettäviin työkaluihin ja niiden toimivuuteen luotiin myös katsaus. Aihepiirin puitteissa käsiteltiin sekä varsinaisia konkreettisia työkaluja että aiheeseen liittyviä abstrakteja ajatelmia ja suunnitelmia. Työssä käytetyt ohjelmistot vaihtuvat hitaasti, joka on luonnollista toimittaessa tiettyihin ohjelmointikieliin tai laitteistoihin sidoksissa.

Koska yrityksessä varsinainen henkilöstö ei juuri ole vaihtunut vuosien aikana, on havaittu tiettyä sokeutta työkaluille ja ohjelmistoille. Ei välttämättä edes osata kaivata mitään erilaista kun aikaisempia osataan käyttää ja niillä saadaan työt tehtyä. Havaittiinkin siis eräänlaista sokeutumista työssä tarvittaville laitteille, johon on saatu kyseenalaistuksia lähinnä juuri tuotelinjan ulkopuolelta siirtyneiden osalta. Enemmistö ohjelmistokehityksen tekijöistä kaipasikin katselmointia työkaluihin ja niiden toimivuuteen, joko yrityksen sisällä tehtynä tai jopa ulkopuolisesta näkökulmasta. Tutkimuksen tulisikin keskittyä suorituskyykyyn ulkoasun ja laajan, mutta hitaasti toimivan ominaisuusvalikoiman sijaan. Samalla kaivattiin myös aikaa ja mahdollisuutta kartoittaa vaihtoehtoisia mahdollisuuksia tai tapoja hoitaa samat asiat, mutta myös todettiin ettei nykyisellään sellaiseen tutkimukseen ole aikaa. Vastaavasti esimiestasolla ei nähty vastaavaa tarvetta vaan oltiin pääasiassa tyytyväisiä käytössä oleviin työkaluihin.

Erilaiset yrityksen sisäiseen toimintaan liittyvän byrokratian ohjelmistojen toimintaa kritisoitiin laajasti. Toiminta oli hidasta tai epävarmaa, riittäviä ohjeita ei ollut saatavilla tehtävän suorittamiseen tai kaikkea ei muista ilman listaa hoidettavasta byrokratiasta.

4.2.8. Tuotetietämys

Tuotetietämyksen osalta pyrittiin kartoittamaan sitä kuinka tuttu tehtävä tuote ja sen toiminta on henkilöstölle. Minkälaista osaamista on, ja tarvittaisiinko sitä lisää. Aihepiiri jakoi erityisen voimakkaasti mielipiteitä henkilöstön ja esimiestason välillä. Yleistason osaaminen havaittiin yleisesti riittämättömäksi. Tuotteen laajuuden huomioiden tämä ei ollut yllätys, mutta tutkimuksessa löytyi myös alueita joihin ei löytynyt osaamista koko tuotekehitysosastolta. Esimerkkinä mainittakoon tieto siitä kuinka tuote otetaan käyttöön ja määritellään asiakkaan tiloissa, eli ns. site connectivity. Osalla haastatelluista oli myös käsitys ettei kaikkea yrityksen toimintaan oleellisesti liittyvää tietoa joka ei kuulu suoraan tuotteen sisältöön, tuoda esille lainkaan.

Esimiestasolla lähes yksimielinen näkemys oli ettei osaamista ole tarpeeksi. Asiaa tarkemmin pohdittaessa alkoi näkemyksissä syntymään eroja, henkilöt painottivat eri asioita osaamisessa. Osan mielestä laajempi tuotetietämys tuotekehityksessä on turhaa. Sen sijaan pitäisi riittää että

henkilö osaa luoda tarkasti teknisten määritelmien mukaista koodia ymmärtämättä sen merkitystä suuremmassa mittakaavassa. Vastakkainen mielipide taas painotti laajempaa ymmärtämystä määritelmien ohella. Uutta toiminnallisuutta on mahdollista tehdä näiden teknisten määrittelyjen mukaisesti mutta vikatilanteiden noustessa esiin tarvitaan laajempaa systeemitason osaamista, eikä sitä ole aina tarvittaessa saatavilla.

Ohjelmiston kehittäjien suurimmaksi ongelmaksi nousi teknisten määritelmien hajanaisuus. Tieto löytyy, mutta useammasta paikasta eivätkä edes varsinaiseen suunnitteluun keskittyneet tuotteen arkkitehdit ole välttämättä osanneet huomioida kaikkia määriteltyjä erikoistapauksia. Varsinaisen ohjelmistokehityksen tekijöillä taas on käytössä yleensä vain yleismäärittelyt joihin tukeutuen saattaa huomattava osa erikoistapauksista jäädä täysin vailla huomiota. Lisäksi havaittiin ongelmia jo yksittäisen toiminnallisuuden kokonaiskuvan hallinnassa jos sen tekeminen ja testaaminen ovat jakautuneet laajasti useiden tekijöiden kesken.

4.2.9. Osaaminen

Osaaminen yleisellä tasolla nostettiin esiin, koska parin edeltävän vuoden kuluessa tapahtuneiden yritysjärjestelyjen ja syystä tai toisesta tapahtuneiden työsuhteiden purkamisten seurauksena on havaittu ongelmia eri osa-alueisiin liittyvien osaamisten siirtämisessä toisille. Monessa tapauksessa tietyn yksityiskohdan osaaminen on aikaisemmin keskittynyt muutamalle henkilölle tai pahimmillaan yhdelle, ja hänen luopuessaan tehtävästä havaitaan siinä kohdalla osaamisessa aukko. Vuosia hiotun yksityiskohdan osaamisen opiskelu nollasta ei ole nopeaa toimintaa.

Osaaminen nostettiin avainkysymykseksi myös ulkopuolisten tekijöiden kanssa toimittaessa. Osaamisen siirto on havaittu useammankin alihankkijan kanssa heikoksi ja tilannetta on katsottu kriittisesti jo hyvinkin lyhyen ajanjakson jälkeen vaikka toiminnan tehokkuuden saavuttamiseen uudessa paikassa uusilla resursseilla vie helposti kolmesta neljään vuotta. Alihankkijan osaaminen katsotaan helposti riittämättömäksi jo puolessa tätä aikajaksoa ja rooliin etsitään uusi tekijä jolloin osaamisen opiskeluaika alkaa usein nollasta. Lisäksi osaamisen siirtäminen alihankkijalle sitoo tehokkaasti yrityksen omia resursseja.

Ohjelmistokehityksessä käytetyt työkalut, ympäristö ja ohjelmointikieli vaihtuivat suurelta osin, ja osalla haastatelluista olikin ollut ongelmia sen kanssa että tuloksia olisi uusilla tavoilla pitänyt saada aikaan samalla teholla kuin vuosia harjoitetussa ympäristössä. Lisäksi osaamisen puutteeksi mainittiin tietämättömyys myöhemmästä testaamisesta eli kuinka ohjelmakoodia ja sen toiminnallisuutta testataan myöhemmissä vaiheissa. Osaamisen parantamiseksi esitettiin enemmän ideoita kuin tarkempaa kritiikkiä. Ehdotettiin pakollista tutustumista myöhempiin testausvaiheisiin sekä lyhyitä toiminnallisuuskoulutuksia koska neljännestantunnissakin ehtisi oppimaan paljon toiminnallisuuksista kun ei vaan paneuduttaisi liikaa yksityiskohtiin.

Lisäksi yleisellä tasolla kritisoitiin viestintätaitojen puutetta. Tämä asia nousi esiin kaikissa haastateltavien kohderyhmissä, koskien kaikkia henkilöstöryhmiä. Esimiehille ehdotettiin koulutusta tai tarpeen tullen lisäkoulutusta ryhmien käsittelystä, sekä ihmeteltiin jatkuvuuden ja pitkäjänteisen suunnitelmallisuuden käsitteitä siinä kuinka pitää todelliset osaajat talossa.

4.2.10. Riippuvuudet

Laajasti hajautetussa ohjelmistokehityksen ympäristössä yksittäisen toiminnallisuuden kaikkia osa-alueita ei voida teettää yhdellä ryhmällä. Ketterien menetelmien ohjeistusten mukaisesti pilkotut tehtävät jaetaan eri kehitysryhmien kesken mutta toiminnallisuuksien päällekkäisyyksien ja suorien riippuvuuksien vuoksi on havaittu ongelmia tämän prosessin synkronoinnissa. Haastateltavien näkemykset asiasta vaihtelivat voimakkaasti, ja erilaisia asiaan vaikuttavia taustaseikkoja nousikin hyvin esille. Synkronointiongelmista muodostuva riski nostettiin aihealueen suurimmaksi ongelmaksi.

Yleisenä havaintona oli se, että liian usein ja liian pieniin paloihin pilkotun tehtävän seurauksena joudutaan odottamaan toisen ryhmän tekemisiä ennen oman tehtävän kehittäminen voi jatkua. Ongelmana tässä nähtiin se, että eri ryhmien näkemykset tiettyjen tehtävien tärkeydestä saattoivat vaihdella suurestikin yhden iteraation sisällä, jolloin toisen ryhmän kannalta kriittinen osa jäi toisella ryhmällä työjonoon alhaisen prioriteetin vuoksi. Laajempien kokonaisuuksien synkronointi onkin jätetty ryhmien omaan hallintaan ilman selkeää ohjeistusta kuinka yhdenmukaistaminen tulee tehdä. Koska asiaa ei ole ohjeistettu, ei myöskään oteta vastuuta asiasta jos asia ei kuulu suoraan omaan vastuualueeseen. Suunnitteluvaiheen synkronointi on paremmin hallussa, mutta siinäkin olisi vielä paljon parantamisen varaa. Näiden tekijöiden seurauksena nähtiinkin laajempi tarve yhteiselle uudelleenfaktoroinnille ja vastuiden nimeämisille. Tämä nimettiin asiaksi joka täysin jokaisen henkilön tai ryhmän oman aktiivisuuden varassa, projektin kannalta ei aikatauluista ja suorituspainesta johtuen ole aikaa mihinkään ylimääräiseen.

Lisäksi toisenlaiseksi riippuvuussuhteeksi nimettiin yksittäiseen alihankkijaan keskittyminen ja jonkun tietyn osa-alueen toiminnallisuuden pitäminen yksinomaan alihankkijan hoidettavana. Yhteistyön katketessa osaamisen siirto yrityksen suuntaan voi olla ongelmallista, ja mitä suuremmasta ja kriittisemmästä osasta on kyse, sitä suurempi riski tästä muodostuu yrityksen toiminnalle.

4.2.11. Demot

Ketterien kehitysmenetelmien mukaisessa toiminnassa määritelty demokäytäntö on usein nimetty yrityksessä turhaksi ajanhaaskaamiseksi. Tässä osassa tutkimusta kartoitettiin haastateltavien mielipiteitä asiasta, ja asioita nousikin esille varsin kattavasti.

Nykymuodossaan demot nimettiin hyödyttömiksi ja lähinnä esimiesten pitämiseksi ajan tasalla tekemisistä. Sen sijaan kaivattiin yleistason esityksiä ja lyhyitä, muutaman minuutin yleiskuvauksia toiminnallisuuksista. Pienten yksityiskohtien sijaan haluttaisiin kuulla kokonaisuuksista selkeästi esitettynä. Tuotosten kyseenalaistamiseen ja mahdolliseen keskusteluun tulisi varata kiinnostuneille mahdollisuus varsinaisen demon ulkopuolella.

Lisäksi sekä tuotteen toiminnallisuuksien että niiden kehittämisessä käytettävien tapojen ja menetelmien seuranta, retro, jää pääsääntöisesti pitämättä eikä mahdollisista ongelmatilanteista osata ottaa laajemmin opiksi. Myöskään näin esiin nousseiden seikkojen jatkokehitykseen ei panosteta nykyisellään lainkaan ellei asiaa ole saatu jostain vietyä ryhmän tehtävälisalle.

4.2.12. Tekninen velka

Uuden tuotteen ensimmäistä versiota laadittaessa otettiin tietoinen riski teknisen velan kanssa tuotteen luomisen nopeuttamiseksi. Tätä velkaa maksetaan eri muodoissaan edelleen, joten aihe olikin hyvä nostaa osaksi aihealueen tehostamisesta ja tuotteen laadun parantamista. Aiheeseen liittyen nousi esiin lukuisia yksityiskohtia joista osaan onkin haastattelutilaisuuksien jälkeen jo haettu parannuksia.

Teknisen velan kertymään vaikuttaa moni seikka. Tärkeimmiksi nousivat puutteelliset toteutukset vanhoissa toiminnallisuuksissa sekä se, ettei anneta aikaa tehdä tarvittavia korjauksia hyvällä laadulla. Tämän seurauksena samoihin asioihin ja osa-alueisiin joudutaan palaamaan yhä uudelleen sen sijaan että alue korjattaisiin kunnolla ajan kanssa. Nykykäytännöillä osa vikojen ja toiminnallisuuksien korjauksia tehdäänkin periaatteella "purkkavirityksiä paikataan purkalla". Suurimmaksi syyksi tälle mainittiin yleisesti se ettei projektin puolelta anneta aikaa tehdä laatua. Lisäksi teknisestä velasta osin johtuva vikamäärä tuotteessa hidastaa varsinaista kehitystä. Näiltä osin nousikin useasta lähteestä kysymys käynnissä olevien ulkoistusten järkevyydestä - ilman ulkopuolista toimintaa lieveilmiöineen laadun ja aikataulujen oletettiin paranevan jo yrityksen omilla resursseilla poistuvan ylimääräisen työn ansiosta. Tuoreiden resurssien määrä saakin olla mitä tahansa jos tekijöille ei anneta mahdollisuutta oppia tehokkaiksi.

Järjestelmätason osaamisen puute mainittiin myös yhdeksi syyksi teknisen velan kertymiseen. Aina ei siis ymmärretä täysin tehdyn koodimuutoksen vaikutuksia sen pienen yksityiskohdan ulkopuolelle. Myöskään kattavaa virheenhallintaa ei tehdä, eikä sitä myöten vikakorjausten aiheuttamista toiminnallisuuden muutoksista kommunikoida muulle tuotekehitykselle juuri lainkaan. Lisäksi liukuihinomainen toiminnallisuuksien mekaaninen lisääminen ilman tarkempaa ja laajempaa analyysia niiden vaikutuksista mainittiin yhdeksi syyksi teknisen velan määrän jatkuvaan kasvamiseen.

4.2.13. Esimiehet ja johtaminen

Luonnollisena osana työelämää, tekemistä ja teettämistä ovat esimiehet erilaissa rooleissa ja johtaminen yleisesti. Aihepiirin sisällyttämällä tutkimukseen pyrittiin tuomaan esiin epäkohtia ja hidasteita aihealueeseen liittyen. Kaikilla haastateltavilla oli joku näkemys aiheeseen, osalla johtamisen kannalta, osalla tekemisen kannalta ja parhaissa tapauksissa molemmat puolet huomioiden.

Pahimmaksi hidasteeksi esimiestoiminnassa nostettiin esiin kysymyksen muodossa "keskustelevatko managerit keskenään ollenkaan?". "Tai tekevätkö he yhteistyötä keskenään?". Yleisenä havaintona johtamisen kannalta havaittiin luonnollinen kiinnostus omaan vastuualueeseen, mutta vastapainona kyvyttömyys huomioida muiden projektien tai tavoitteiden vaatimuksia. Lisäksi nostettiin esiin eräiden johtavissa rooleissa olevien henkilöiden teknisen tietämyksen vajavaisuus, jonka näkyvä muoto onkin epärealistiset odotukset ja suoriutumistavoitteet.

Eri organisaatiotasojen välillä havaittiin puutteita luottamuksesta, joka näkyi puuttumisena eri aihealueiden yksityiskohtiin sen sijaan että luotettaisiin toisten kykyihin ja osaamiseen. Tämän eräänä seurauksena usein oikeat viestintäkanavatkin unohtuvat ja uusien ohjeistusten tai raporttien toimittaminen saattaa hypätä tulevan kannalta oleellisen organisaatiotason ylitse. Erääksi lähimmän esimiehen tehtäväksi on mielletty tiedottaminen ja tarpeettoman tiedon suodattaminen pois häiritsemästä. Tässä roolissa esimiestoiminnassa havaittiin puutteita, tiedonvälityksen suodattaminen on joskus liiankin tehokasta tai vastaanottavana osapuolena ei vastata millään tapaa saatuihin viesteihin tai kysymyksiin. Lisäksi esiin nousi puute ideoista ja innovaatioista palkitsemisesta; mikä estäisi kehittämästä uusia, muista ympäristöistä tuttuja palkitsemistapoja kuten palkallinen vapaapäivä rahapalkkion sijaan.

4.2.14. Näkyvyys

Näkyvyydellä aihealueena käsitetään tässä yhteydessä sekä sitä miten ryhmät näkevät toisten ryhmien toiminnan että sitä, kuinka esimiehet ja managerit näkevät ryhmien toiminnan. Tutkitaan siis sitä kuinka hyvin yleisesti ollaan tietoisia siitä missä vaiheessa kehitystyö menee.

Yleisesti kritiikkiä sai tiedottaminen. Jos asioista halusi yleisesti tietää, oli käytäväkeskusteluilla ja epämuodollisilla tapaamisilla suurempi rooli kuin virallisella tiedottamisella. Usein tieto siirtyi projektin johdolle päin, mutta muut sidosryhmät jotka olisivat voineet hyötyä kyseisestä tiedosta, jäivät huhujen ja oletusten varaan ellei epävirallisia kanavia pitkin löytynyt tarkempaa tietoa. Samalla havaittiin ettei managereilla ole aina tietoa siitä mitä ryhmät tekevät, eikä ryhmillä ole välttämättä aina tietoa siitä mitä heidän esimiehensä tekee. Pääasiassa ryhmien välinen näkyvyys saman toimipisteen sisällä arvioitiin hyväksi ja toimivaksi.

Suurta kritiikkiä saivat ns. "task forcet", eli määrättyyn tarkoitukseen perustetut ryhmät. Niiden tiedottaminen arvioitiin yleisesti ala-arvoiseksi ja todettiin kyseisten ryhmien pääasiassa sulkeutuvan omaan tilaansa ja kommunikoivan ainoastaan kyseisen projektin managerin kanssa unohtaen yleisen, muihin vaikuttaviin asioihin liittyvän kommunikaation. Sama ilmiö laajemmassa mittakaavassa on havaittavissa eri toimialueiden ja eri toimipisteiden välillä. Usein saatetaan samaa, laajempaa kokonaisuutta rakentaa alusta asti eri paikoissa erikseen sen sijaan että kommunikoitaisiin laajemmin ja hyödynnettäisiin jo olemassa olevaa.

4.2.15. Asiakashyöty

Vaikka suurin osa haastatelluista työskentelee tehtävässä jossa ei suoraan kommunikoida asiakasrajapinnalle, nostettiin asiakashyöty aihealueena esiin. Tavoitteena oli katsoa eri työroolien näkyvyys siihen kuinka asiakas saa parhaan hyödyn tuotekehityksen toiminnasta. Aihetta käsiteltiin rajaamalla varsinaiset tuotekehitys ja vikakorjaukset alueen ulkopuolelle.

Ensimmäisenä, ja suurimpana ajatuksena nousivat aikaisemmat asiakaskokemukset. Aiheesta todettiin laajasti ettei ole kyetty oppimaan aikaisemmasta, ei hyvistä eikä huonoista kokemuksista. Sen sijaan toiminnassa pysytään kirjallisten ohjeistusten muodostamassa viitekehityksessä kykenemättä kehittämään sitä. Toisena havaintona oli tuotteen sisällön suunnittelun kannalta kriittinen asia - pitäisikö keskittyä enemmän siihen mitä asiakkaat haluavat ja tarvitsevat sen sijaan että kehitetään erilaisia aikaa vieviä toiminnallisuuksia joille ei löydy välttämättä markkinoita laajemmin.

Erilaisten hyväksyntäohjeiden ajan tasalla pitäminen nähtiin tärkeäksi osaksi asiakasrajapinnan tyydyttämistä, sekä varsinainen testaus ja sen lisääminen. Nyt nähtiin huolestuttavana piirteenä se että osa testaamisesta jätetään tietoisesti asiakkaan tehtäväksi. Lisäksi asiakkaan toimipisteellä tarvitsema tuki tulisi järjestää ensiluokkaiseksi kaikissa tilanteissa.

4.2.16. Toimintatavat

Toimintatapojen esiin nostamisella pyrittiin saamaan näkemys siihen kuinka toimivina tai toimimattomina yrityksen työntekijät näkevät erilaiset prosessin vaiheet ja ohjeistetut tai ohjeistamattomat työskentelytavat sekä niihin oleellisesti liittyvät seikat. Toimintatapojen kannalta huomioitavaa on se, että jokainen haastatelluista on työskennellyt yrityksessä vesiputousmallin, siirtymävaiheen sekä ketterien kehitysmenetelmien mukaisissa työvaiheissa.

Yrityksen toimintatavoissa ollaan käytännössä luopumassa ketterien menetelmien mukaisesta kollektiivisesta vastuusta koska se havaittiin käytännössä toimimattomaksi. Aiheeseen liittyen todettiin vastuiden määrittämisen olevan menetelmällisesti suuri askel taaksepäin, mutta kehityksen kannalta välttämättömiä. Vastuiden myös todettiin keskittyvän

henkilöille ryhmien sijaan, ja esiin nostettiin jaetun osaamisen ja osaamisen varmistamisen käsitteet.

Uusiin toimintatapoihin liittyen useampi haastateltu nosti esiin epäilyjä jotka muodostavat pidemmällä aikavälillä suuria riskejä yrityksen sisäisen osaamisen kannalta. Tärkeimpänä esiin nousi ajatus ryhmien avainjäsenien lähteminen yrityksestä. Syyksi tähän mainittiin viimeisimpien sisäisten toimintaohjeiden mukaiset muutokset, joiden toteutuessa sellaisenaan paikallisten ryhmien jäsenet kuluttaisivat suuren osan työajastaan eri osapuolten väliseen koordinointiin, kommentointiin, byrokraattisiin seikkoihin ja erilaisissa tapaamisissa olemiseen sen sijaan että pääsisivät ohjelmointityöhön johon ovat ensi sijassa aikanaan tulleet. Joidenkin kohdalla tässä on kyseessä jopa kutsumus-tyyppinen ammatti. Osaamisen pirstoutuminen eri tekijäryhmien välillä muodostaa myös riskin puutteellisten kokonaiskuvien hallinnan vuoksi. Lisäksi eri ryhmien toiminnasta ja eri vaiheissa ristiin käytetyistä arviointimenetelmistä, tarinapisteiden (*en. story points*) tunti- ja viitemalleista johtuen projektin kokonaisetenemän ennustettavuus heikkenee, tai vähintäänkin aiheuttaa sekaannuksia ennen arviointiperusteiden vakiintumista.

Lisäksi epäilyksiä nousi siinä, onko yritys valmis muuttamaan ohjeistettuja tekemisen käytäntöjä välittömästi jos havaitaan niiden olevan soveltumattomia tarkoitukseensa. Vastapainona tähän kaivattiin myös tekemisten harmonisointia eri tuotelinjojen ja osastojen välillä sen sijaan että joka tuotteelle on muodostunut omat käytäntönsä joiden muuttaminen on haasteellista. Lisäksi toimintatapojen muutoksiin liittyvä tiedottaminen nimettiin ehdottomasti vajavaiseksi.

Positiivistakin palautetta ehdotetuista ja jo käyttöön otetuista muutoksista haluttiin sanoa. Ryhmien toiminnan sisäisen koordinoinnin uskotaan paranevan muutosten myötä, sekä ohjelmakoodin eriytettyjen kehityshaarojen myötä koodin päähaaran uskotaan pysyvän paremmin kunnossa ja julkaisukelpoisena ketterien kehitysmenetelmien periaatteiden mukaisesti. Lisäksi eräänä erittäin tervetulleena muutoksena vastaan otettiin ns. suunnitteluvaihe, joka oli pidemmän aikaa jo terminäkin kielletty, koska sen koettiin olevan askel takaisin vesiputousmallin mukaiseen toimintaan. Lyhyen aikavälin aikana todettiin sen tarpeellisuus ja suunnitteluvaihe otettiin mukaan ensin toivottuna, sitten pakollisena osuutena.

4.2.17. Päähidasteita

Haastattelutilaisuuden lopuksi pyydettiin jokaista haastateltavaa miettimään hetken sitä, mikä on ollut edeltävien iteraatioiden aikana eniten haittaava tekijä oman työroolin osalta tai mikä on hidastanut ohjelmistokehityksen etenemistä voimakkaimmin. Aihepiirin osalta nousi useammasta lähteestä samoja kohteita, joista osa oli kokonaisuuden kannalta tullut esille jo aikaisempien aihealueiden käsittelyssä.

Ylivoimaisesti tärkeimmäksi hidasteeksi nousi osaajien ylityöllistyminen, ja sen laajentumana se ettei yksilöille anneta mahdollisuutta hankkia

jonkun tietyn osa-alueen erikoisosaamista. Toiseksi ryhmien tehokasta toimintaa hidastavaksi tekijäksi nousi ryhmien hajauttaminen eri tehtävien pariin, usein jopa täysin eri projektien alueille.

Ohjelmistokehittäjien parissa esiin nousivat esimiesportaasta eroten rampup/rampdown -kuviot joita yrityksen tuotekehityksessä on tapahtunut useita. Uusien yhteistyökumppanien tuominen projekteihin vaatii myös kompetenssin siirtoa johon varattu aika on nähty riittämättömäksi. Samoin ylöspäin yrityshierarkiassa näkyy vain henkilöstön pääluku joka pysyy yleisesti jokseenkin samana mutta puutteellisesta osaamisesta ja vaillinaisesta oppimisesta johtuvaa tuotoksen romahtamista ei nähdä oikeassa merkityksessään.

Lisäksi vähäisempänä hidasteena nimettiin rutiinien ja käytäntöjen puutteellisuus. Vähintään tärkeimmistä pitäisi saada tarkka ja ajantasainen muistio tai muistilista luoduksi ja kaikkien saataville. Lisääntyneen ohjelmistokehittäjiltäkin vaaditun byrokratian vuoksi voi välillä jäädä osa vaadituista tehtävistä tekemättä tai tulla väärin tehdyksi joka aiheuttaa sitten jossain myöhemmässä vaiheessa lisää työtä, tai vähintään aiheeseen palaamisen jonkun muun tehtävän kustannuksella.

Eräänä hidasteena myös mainittiin jatkuva testiautomaation sokea tuijottaminen sen sijaan että sitä järkeistettäisiin oikeasti vaadittavalle tasolle. Suuri osa viikoittaisesta työpanoksesta kuluu automaattisten testien tulosten analysointiin vaikka tätäkin voisi järkeistää ja nopeuttaa nopeasti itsensä takaisin maksavalla työllä. Viimeisenä esiin nousseena hidastavana tekijänä mainittiin historia ja sen luomat mielikuvat ja käytännöt; tehdään asioita hitaalla mutta jo omaksutulla tavalla, hitailla mutta tutuilla työkaluilla sen sijaan että etsittäisiin uusia ja tehokkaita työkaluja asioiden hoitamiseen.

4.3. Kollektiivinen näkemys muutoksista

Haastattelujen yhteydessä selvitettiin myös tulevaisuuden näkymiä kokeneiden ohjelmistokehittäjien mielestä. Mahdollisia riskejä tuotteeseen, työskentelytapoihin ja työresurssien käyttöön liittyen kartoitettiin. Näitä näkemyksiä vallitsevista työskentelytavoista ja ohjeistuksista on koostettu tähän SWOT-analyysin muodossa, jota voidaan käyttää jatkokehittelyn apuna.

Taulukko 5. Kollektiivinen analyysi uudesta organisaatiomallista

Vahvuudet	Heikkoudet
Ryhmien koko; Ryhmä voi toimia tehokkaasti	Ryhmien koko; Eivät enää oikeita ryhmiä
Vastuu osa-alueista	Kokonaiskuva puuttuu
Koodi pysyy kunnossa	Ei osaamista toiminnallisuuksista
	Puuttuvat prosessikuvaukset
	Byrokratia lisääntyy

	Riippuvuus alihankkijoista
Mahdollisuudet	Uhat
Ryhmien synkronointi tehokasta	Kyllästyminen
Alihankkijoiden määrän muokkaus	Synkronoinnin epäonnistuminen
Samanaikainen kehittäminen	Viiveet
Nopea etenemä	Henkilöityminen
	Alihankkijan resurssit
	Byrokratian kiertäminen
	Osaamisen hiipuminen

SWOT-analyysi itsessään on yksinkertainen ja ennen kaikkea hyödyllinen apuväline hankkeiden tai projektien suunnittelussa ja sitä voidaan käyttää apuna pohdittaessa kuinka vahvuudet saadaan esiin tai mahdollisuudet hyödynnettyä. Lisäksi sen avulla on mahdollista varautua uhkiin tai minimoida tunnistettujen heikkouksien vaikutusta.

Hahmotetut vahvuudet ja mahdollisuudet liittyvät voimakkaasti ryhmien toimintaan ja työtehtävien ja resurssien järjestelyyn. Yllättävästi ryhmien koko uudessa organisaation toimintamallissa nähtiin sekä hyvänä että pahana asiana. Toisaalta ryhmä voi pienikokoisena toimia tehokkaasti ja jokaiselle työntekijälle on selvää mitä hänen on tehtävä projektin osalta. Toisaalta taas ryhmien pienestä koosta ja jäsenten määrätystä tehtävistä johtuen kyseessä ei enää ole oikea ryhmä. Samoin nimetyt osaamis- ja vastuualueet ovat nimellisesti ketterien ohjelmistokehitysmenetelmien vastaisia. Johtavien ja kehittävien ryhmien jaottelun mukaan koodin varsinaisella julkaisuhaaralla tulee pysyä kunnossa koko ajan, joka nähtiin lähes kaikkialla tärkeimpänä vahvuutena tuotteen kannalta. Samoin ryhmämalli sallii alihankkijaresurssien muokkaamisen suuntaan tai toiseen nopeastikin tarpeen mukaan.

Kielteisistä puolista nostettiin esiin byrokratian lisääntyminen. Lähes kaikista työvaiheista on olemassa ohjeistus, ja ohjeiden määrää ja sisältöä päivitetään jatkuvasti. Kun aikaisemmissa toimintamalleissa ohjelmistokehittäjä korjasi vian ja kirjoitti vianhallinnan työkaluun lyhyen kuvauksen koodista sekä tiedon korjatusta versiosta, hän oli valmis siirtymään seuraavan tehtävän pariin. Nykymallissa, jos vikakorjaus ei valmistu välittömästi, kehittäjä kirjoittaa muun muassa ennakkoanalyysin viasta, korjausarvion, päivittäisiä raportteja edistymisestään, varsinaisen analyysin, asiakkaalle lähtevän korjauskuvauksen, sisäisen kuvauksen vikakorjauksesta sekä mahdollisen testausohjeistuksen. Lisäksi hänen vastuullaan on tämän jälkeen päivittää sisäisiä ohjelmistokehityksen dokumentteja joita muut kehittäjät hyväksikäyttävät omissa tehtävissään. Tämän seurauksena havaittiin taipumus jättää joskus osa kirjallisista suoritteista tekemättä jollei niitä erikseen pyydetä sekä yleisesti ohittamaan byrokratian vaatimat kohdat mahdollisuuksien mukaan.

Useat ohjelmistokehittäjät nostivat esille myös kyllästymis-tekijän. Suurin osa kehittäjistä on alunperin hakeutunut alalle tekemään ohjelmointityötä, ja nyt muiden tehtävien (koordinointi, dokumentointi, kuvausten

kirjoittaminen, suunnittelu, testitapausten analysointi, yms.) viedessä valtaosan työajasta, voi edessä olla kyllästyminen töiden sisältöön joka näkyy osaltaan motivaation hiipumisena ja mahdollisesti työpaikan vaihdoksena. Tämä nostaakin esiin suurimman riskin henkilöstöön liittyen - avainhenkilöiden lähtemisen. Näin varsinkin nykymallissa jossa tietyt toiminnallisuudet voivat kokonaisuudessaan jäädä yhden tai useamman kehittäjän vastuualueiksi. Samoin uhkana nähtiin yrityksen sisäisen osaamisen hiipuminen ohjelmointityön keskittyessä alihankkijoiden tehtäväksi omien resurssien keskittyessä muihin asiaan liittyviin tehtäviin.

Laaja-alaisen ohjelmistokehityksen sisäisiä ja ulkoisia riskitekijöitä ja mahdollisuuksia varautua niihin on avattu alla olevien taulukoiden mukaisesti ulkoisesta ja sisäisestä näkökulmasta. Riskien pohtimisen tavoitteena on herättää ajatuksia seuraavista seikoista:

- Mitkä riskit uhkaavat yritystä ja sen toimintaa
- Miten eri riskeihin voidaan varautua
- Miten toteutuvien riskien vaikutus minimoidaan
- Millainen vaikutus toteutuvalla riskillä voi olla
- Miten yritys selviytyy jos useampi riski toteutuu

Taulukko 6. Ulkoisen ympäristön riskit

Ulkoinen ympäristö	Riski	Todennäköisyys	Välttäminen/Kantaminen
Taloudellinen ympäristö	Ei investointeja	xx	Varautuminen ennakolta
Poliittinen ympäristö	Alihankkijan kohdemaan ilmapiiri Poliittinen ilmapiiri	x x	Varasuunnitelma Hajauttaminen
Sosiaalinen ympäristö	Alihankkijan käyttö	x	Varasuunnitelma
Tekninen ympäristö	Vaatimukset suurempia kuin mitä yritys kykenee toteuttamaan	xx	Varautuminen ennakolta
Toimialan kehitys	Yrityskaupan osana/osaksi	x	-
Markkinat ja asiakkaat	Laatuongelmat	xxx	Laadun parantaminen
Kilpailu	Kilpailutilanne ja tarjousten häviäminen	x	Laadun parantaminen

Taloudellinen ympäristö ja siihen vaikuttavat tekijät seuraavat suuressa mittakaavassa maailmanlaajuisia megatrendejä. Riskin muodostumiseen vaikuttavat operaattorien investointien pysähtyminen ja mahdollisista laatu- tai imagotekijöistä muodostuvat uhat. Nykytilanteen mukaisesti Investointeja ei välttämättä haluta tehdä, mutta dataliikenteen jatkuvan kasvun ja uusien, liikkuvien tietoliikennettä synnyttävien laitteiden yleistymisen pienentää tätä riskiä pakottamalla operaattorit investoimaan verkkokapasiteetin kasvattamiseen.

Poliittinen ympäristö on myös tärkeässä roolissa hajautetun mallin mukaisessa toiminnassa jossa toimintaa on poliittisesti epävakaisissa maissa. Mahdolliset poliittisten linjausten muutokset tai radikaalimpi toiminta maassa voivat sinänsä aiheuttaa suuriakin vaikutuksia yrityksen toiminnalle. Riskin välttäminen hajautetussa ympäristössä on lähestulkoon mahdotonta ilman jatkuvaa seuranta ja suunnitelmia nopeisiinkin muutoksiin. Vaikutuksia voidaan kuitenkin pienentää hajauttamalla riski useampien valtioiden ja yritysten välillä. Sosiaalinen ympäristö sisältää ihmisten mielikuvat yrityksestä, sen toiminnasta ja tuotteista. Riskien havainnointi ja välttäminen tapahtuvat samoilla perusteilla kuin poliittisen ympäristön kohdalla.

Tekninen ympäristö määritetään sen mukaan kuinka ajan harjalla yrityksen toiminta on. Suositun tuotteen tai toiminnallisuuden osalta voi seurata tilanne jossa ostajia olisi mutta yritys ei kykene tuottamaan haluttua tuotetta ja menettää sen myötä markkinoita kilpailijoille. Myöskin tuotteen vastaavuus ostajan vaatimuksiin sisältyy teknisen ympäristön avaintekijöihin. Parhaat mahdollisuudet riskien minimointiin aihealueen sisällä saavutetaan huolellisilla ennakosuunnitelmilla.

Toimialan kehitys maailmanlaajuisesti voi myös sisältää yrityksen kannalta riskejä joihin on vaikeampi varautua. Taloudellinen tilanne tai yrityksessä todellista valtaa käyttävät tahot voivat pakottaa esimerkiksi yrityskauppoihin nopeallakin aikataululla.

Asiakkaat muodostavat jokaisen yrityksen toiminnan kannalta oleellisen osan joka mahdollistaa yrityksen toiminnan. Niiden kanssa läheisesti yhteen kuuluvat yritysten välinen kilpailu ja markkinatilanne. Laajan aihealueen suurin yksittäinen riski muodostuu tuotteen laadusta. Jos laatu on heikkoa tai ei vastaa luvattua, voidaan tällä ohjata asiakkaat nopeastikin kilpailijan tuotteen pariin. Laatuun, hinnoitteluun ja erilaisten lupauksen pitävyyteen panostamalla voidaankin helposti torjua markkinoiden vaatimusten ja kilpailun muodostamia riskejä.

Taulukko 7. Sisäisen ympäristön riskit

Yrityksen sisäiset tekijät	Riski	Todennäköisyys	Välttäminen/Kantaminen
Organisaatio ja henkilöstö	Motivaatio-ongelmat	xx	Motivointi
Organisaatio ja henkilöstö	Osaaminen	xx	Koulutuksen lisääminen
Organisaatio ja henkilöstö	Henkilöstön vaihtuvuus	x	Motivointi
Tuotanto	Laatuongelmat Menetelmät	xxx	Laadun parantaminen Jatkuva kehittäminen
Talous	Hallitsemattomat leikkaukset	xx	Järkevä kohdentaminen

Yrityksen toiminnan sisäisistä tekijöistä organisaatio ja henkilöstö muodostavat suurimman osatekijän. Riskejä aihealueen sisällä muodostavat sitoutuminen yrityksen toimintaan ja osaamisen kehittäminen. Sitoutuminen vaikuttaa suoraan motivaatioon, ja eräs suora mittari alueella onkin henkilöstön vaihtuvuus. Sitoutumisen varmistamisessa tärkein yksittäinen tekijä onkin henkilöstön motivointi. Ammatillisen osaamisen kehittäminen on tärkeä osa itsensä motivointia joten osaamisen päivittämiseen kannattaa panostaa myös kokeneen henkilöstön kouluttamisen muodossa.

Käytännön tuotannon kannalta suurimmat yksittäiset riskit muodostuvat tuotekehityksen tavoista ja tuotteen laadusta. Tehottomat menetelmät ja jatkuva käytännön toimintatapojen muokkaaminen vievät pohjaa tehokkaalta toiminnalta. Menetelmiä ja toimintatapoja tulee tietenkin parantaa, mutta rauhallinen, vähittäinen kehittäminen auttaa välttämään riskien toteutumista aihealueella. Laadun parantaminen taas vaikuttaa pienentävästi lähes kaikkiin ajateltavissa oleviin riskeihin.

Taloudelliset tekijät itsessään muodostavat oman riskinsä. Taloudellisten syiden perusteella voidaan leikata tuotekehitystä ja menettää tärkeää osaamista joka ymmärretään vasta jälkikäteen. Tärkeä tekijä aihealueella olevien riskien minimoinnissa onkin huolellinen ennakkosuunnittelu ja mahdollisten säästöjen ja investointien tarkkaan harkittu kohdentaminen.

5 POHDINTA

Tutkimuksen pääasiallisena tavoitteena on ollut osoittaa potentiaalisia jatkokehityskohteita yrityksen toiminnassa. Käytännön kokemuksia ja teorialalleja vertaillen voidaan tehdä päätelmiä ja esittää oletuksia ketterien ohjelmistokehitysmenetelmien yrityksen toimintaa koskevilta osilta. Kaikkien ketterän scrum -mallin mukaisten kohtien soveltaminen laaja-alaisessa kehitysympäristössä on osoittautunut sitä haastavammaksi mitä laajempi ja monikulttuurisempi toimintaympäristö on.

5.1. Reliabiliteetin ja validiteetin arviointi

Tutkimuksen tulosten luotettavuutta ja paikkansa pitävyyttä voidaan arvioida kahdesta eri näkökulmasta, yleisen ohjelmistokehityksen sekä yrityksen oman toiminnan kannalta. Yhteisiä tekijöitä molempien kannalta ovat sekä maailman laajuiset megatrendit toimialalla että pienemmän mittakaavan ilmiöt. Näitä laajempia trendejä ovat muun muassa tietoverkkojen ja tiedonsiirron muuttaminen analogisista ympäristöistä digitaalisiin sekä siirrettävän tietoliikenteen määrän jatkuva kasvu. Alalla vallitsevat ilmiöt kuten markkinoiden tyydyttyminen tai yritysten yhdistymiset vaikuttavat myös toisiin saman alan yrityksiin ja niiden toimintaan vastaavalla tavalla. Käytännössä näiden tekijöiden vaikutukset työympäristöön voidaankin tiivistää kahteen päätekijään jotka ovat yrityksen sisäisen toiminnan tehostaminen ja kustannusten karsiminen.

Tutkimuksen ensimmäisen menetelmän, kokemuksiin pohjautuvan nykytilanteen kartoittamisen osalta on huomioitava muutamia seikkoja. Osa esiin tulleista aiheista on voinut muuttua tutkimuksen kuluessa, sillä tutkimuksen kokonaiskeston aikana on yrityksessä tapahtunut useita toimintaohjeistuksien muutoksia sekä kaksi organisaatiomuutosta. Tutkimusjakson ja opinnäytetyöhön kokonaisuudessaan käytetyn ajan kuluessa on yritysten välisessä yhteistyössä tapahtunut suuria muutoksia. Uusia sopimuksia on tehty, toimintaa toisissa paikoissa laajennettu ja toisaalla kokonaisen yhteistyökumppanin osalta sopimukset irtisanottu. Lisäksi empiirisen tarkastelun näkökulma ei ole suoraan ohjelmistokehitystä tekevästä ryhmästä vaan sitä lähellä olevan tukiryhmän suunnasta. Muutoksista huolimatta perusongelmat ovat pysyneet samankaltaisina. Myös epävirallisissa yhteyksissä (kahvipöytätai käytäväkeskustelut) varsinaisilta ohjelmistokehittäjiltä kuullut kommentit ja mielipiteet vahvistavat ongelmakohtien olemassaolon. Tästä johtuen voidaan todeta empiiristen havaintojen olevan paikkansa pitäviä ja luotettavia.

Ketteriä ohjelmistokehityksen menetelmiä on käytetty yrityksessä useita vuosia ja niiden käytöstä on muodostunut yrityksessä omat käytäntönsä. Tämän seurauksena voidaan tehtyjä havaintoja pitää luotettavana osoituksena työmenetelmien toimivuudesta kyseisessä työympäristössä. Yrityksen näkökulmasta tutkimuksen luotettavuus perustuu haastattelututkimuksen osiin joissa havaittiin yhteisiä ongelmakohtia, eli tapahtui ns. saturaatiota. Samoja ongelmia, ja samoihin juurisyihin osoittavia, mutta eri tavoin oireilevia kohtia nousi esiin eri

henkilöstöryhmien toimesta. Tästä syystä voidaan todeta tutkimuksen tulosten pitävän paikkansa hyvin kohdeympäristönsä osalta.

Yleisen ohjelmistokehityksen kannalta tai tieteelliseltä kannalta tutkittuna voidaan pääsääntöisesti olettaa ketterien kehitysmenetelmien teesit teoreettisella tasolla paikkansa pitäviksi. Arvioitaessa ketterien kehitysmenetelmien soveltuvuutta käytettyyn tarkoitukseensa täytyy kuitenkin muistaa että alunperin menetelmät laadittiin kokeneiden ohjelmistokehittäjien parhaiden käytäntöjen mukaisesti, kuitenkin huomioimatta globalisaation vaikutuksia tai erilaisia kehitysympäristöjä. Lähtökohtana menetelmien kehitykselle olikin selkeästi homogeeninen käyttäjäryhmä rajallisen kokoisessa ympäristössä. Varsinainen scrum -viitekehys on laajentunut (Larman & Vodde, 2010) yrityksen laatiman ohjeistuksen jälkeen ulkoistamiseen ja kansainvälisyyteen, mutta tätä muutosta ei täysin ole huomioitu yrityksen käyttämissä menetelmissä. Tulokset eivät siten ole välttämättä yleisesti paikkaansa pitäviä yleisen ohjelmistokehityksen näkökulmasta.

Kokonaisuudessaan tämän opinnäytetyön voidaan kuitenkin katsoa täyttäneen tehtävänsä erinomaisesti, joten tutkimuksen yleisen validiteetin voidaan katsoa olevan hyvällä tasolla.

5.2. Menetelmien soveltuvuus

Yrityksen käyttöön valittua scrum -menetelmää tutkittaessa havaitaan useita seikkoja joiden tärkeyttä mallin kuvauksessa korostetaan, mutta joita ei yrityksessä olla onnistuttu toteuttamaan.

Liitteessä 2 on esitetty yhteenvetomaisesti taulukko erilaisten ketterien ohjelmistokehitysmenetelmien ominaispiirteistä. Yrityksen toiminnassa tapahtuneisiin muutoksiin pohjautuen voidaankin kyseenalaistaa oliko scrum oikea valinta menetelmäksi. Muutoksia alkuperäisestä, toimimattomaksi havaitusta toimintatavasta onkin tehty pääasiassa suunnitteluvaiheeseen jota on tehostettu voimakkaasti alussa käytetystä kevyestä suunnittelusta. Yhteenvedon mukaisesti toinen kyseenalainen kohta on skaalautuvuus. Scrum menetelmänä on laadittu skaalautuvaksi, mutta yrityksen ohjeinaan käyttämään teoriamalliin on vasta jälkikäteen (Larman & Vodde, 2008, 2010) lisätty ohjeistus kuinka laajentaa toimintaa kansainvälisessä ympäristössä. Yrityksen kannalta voidaankin todeta toiminnan laajentumisen kansainväliseksi tapahtuneen villisti ohjeistuksen kehittyessä samalla ainoastaan osittain.

Teoriamallin mukaisessa toiminnassa moni asia liittyy projektin johtamiseen. Kun yrityksessä ensivaiheessa kokeilun luonteisesti kehitettiin toiminnallisuuksia käyttäen scrum -menetelmää, onnistuttiin asiassa hyvin koska ryhmille joissa kokeilut tehtiin annettiin täysi vapaus ominaisuuden toteuttamisessa. Kokeiluprojekteissa ryhmille annettiin tehtävä ja viitekehys minkä sisällä toimia, ja iteraatioiden lopputuotokset ja kokemukset antoivat pohjaa päätökselle siirtyä kokonaisuudessaan ketterien ohjelmistokehitysmenetelmien pariin. Johtamisen rooli on kuitenkin erilainen kuin perinteisemmissä toimintatavoissa. Yrityksen

toiminnassa ei kyetty pääsemään eroon vanhoista johtamistavoista, eivätkä ketterät ryhmät ole sen seurauksena päässeet itseohjautuvuuteen siten kuin malli olisi vaatinut. Johtamisen kannalta eri työrooleissa toimivat henkilöt olisivat kaivanneet sekä enemmän koulutusta uusiin menetelmiin että tarkemman ohjeistuksen siihen mitä heidän oma roolinsa sisältää. Nyt menetelmäkoulutus kohdistui pääasiassa ohjelmistokehittäjiin.

Lisäksi resurssien jakaminen eri projektien välillä on aiheuttanut ongelmia haitaten ohjelmistokehitysprosessia monin tavoin. Tuotekehitysprojektin aliprojekteille olisikin mahdollisesti pitänyt määrittää omat tekijänsä sen sijaan että resursseja kohdennettiin villisti jonka tuloksena iteraation tuotosta on haastavaa arvioida ennakolta. Johtamisen ongelmaksi voidaan myös laskea yrityksen kyvyttömyys motivoida työntekijöitä, joka on suoraan näkynyt suoritustason laskemisena. Eräänä suurena tekijänä ongelmiin voidaan myös mainita tekninen velka jota otettiin tietoisesti mutta jota ei olla myöhemmin kyetty poistamaan. Muutoksen suorittaminen on aina haastavaa, ja yrityksessä epäonnistuttiin vaatimaan muutos henkilötasolla. Toisaalta, usea tutkimuksessa hidasteena ilmi tullut asia on ollut esillä koko ajan, mutta aikaisemmin asian tuominen laajempaan tietoisuuteen on tuomittu saman tien muutosvastarintana vaikka kyseessä on selvästi toiminnan tehokkuuteen vaikuttava seikka.

6 KONTRIBUUTIO

Työn tavoitteena oli selvittää mahdollisimman kattavasti hidasteita ja mahdollisia virheitä toimintatavoissa. Löytyneet seikat nostettiin esiin sekä tuotelinjan johtotasolle että laajemmin toimintatapoja, laatua ja menetelmiä kehittäväälle osastolle. Kontribuutio yrityksen kannalta kohdistui pääasiallisesti tulevien kehityskohteiden osoittamiseen esiin nousseiden tulosten mukaisesti.

Eri kehitysmenetelmiä ja niiden toimivuutta yrityksen sisällä verrattaessa muodostui kuitenkin ristiriitoja kokemusten osoittamien toimivien tapojen ja teorian esittämien toimintatapojen välillä. Tämä osoitti yrityksessä muodostuneet käytännöt ainakin osittain epätarkoiksi tai soveltumattomiksi tarkoitukseensa. Osa eroista on muodostunut tietoisesti, osa tiedostamattomista syistä.

Mahdollisina parannuskohteina esiin on noussut muutama oleellinen seikka. Eräänä puutteena havaittiin iteraation keskellä välittömästi tehtäväksi tulevat, mutta suunnitellun sisällön ulkopuoliset tekijät ja niiden ylös kirjaaminen. Varsinaista toimintaa ja aikatauluja häiritsevät työtehtävät tulisi kirjata välittömästi ylös riittävällä tarkkuudella. Tilastoihin tulisi sisällyttää vähintään tehtävän esittäjä, ajankohta, syy, kesto ja muita mahdollisia, oleellisia tietoja joita hyväksikäyttämällä voidaan vastaisuudessa pyrkiä välttämään vastaavat tilanteet. Samoin yrityksessä laadittuja ja yleisesti kommunikoituja tiekarttoja tuotteen osalta ei päivitetä niin usein kuin ketterien kehitysmenetelmien mukainen tarkka seuranta vaatisi.

6.1. Henkilöstön motivointi

Henkilöstöltä puuttuvat sekä teknisiin että menetelmällisiin asioihin liittyen selkeät kommunikaatiokanavat. Hierarkkisen organisaatiokaavion mukaiset viestikanavat riippuvat täysin henkilöiden omista mielipiteistä ja sanoman eteenpäin kulkeminen riippuukin täysin siitä mitä mieltä kohdehenkilöt asioista ovat. Käytännössä vaihtoehtoiksi jäävätkin joko esimiesten täydellinen ohittaminen tai sähköinen levittäminen laajalla jakelulla. Palaute kommentteista jää yleisesti liian usein saamatta, johtaen lopulta tilanteeseen jossa henkilöstö tuntee ettei sitä kuunnella lainkaan.

Lisäksi henkilöstön motivointiin ja palkitsemiseen käytetyt menetelmät tulisi muuttaa siihen suuntaan että henkilöstön jäsenet voivat itse vaikuttaa omiin palkitsemisjärjestelmiin. Viime aikojen muutokset ovat käytännössä poistaneet henkilökohtaiset vaikutusmahdollisuudet esim. *short term bonus plan* -järjestelmän piirissä olevilta työntekijöiltä. Työmotivaation voi uskoa paranevan vähentämällä yrityksen laajuisia muuttujia ja lisäämällä ohjelmaan pienemmän mittakaavan kohteita joihin on mahdollista vaikuttaa omalla suoralla työpanoksella. Lisäksi voisi harkita muissa, erityyppisissä organisaatioissa käytettyjen täysin erilaisten palkitsemis- ja motivointitapojen esittelyä myös ohjelmistokehityksen alueella, esimerkkinä vaikkapa palkallinen vapaapäivä (vrt. kuntoisuusloma puolustusvoimissa).

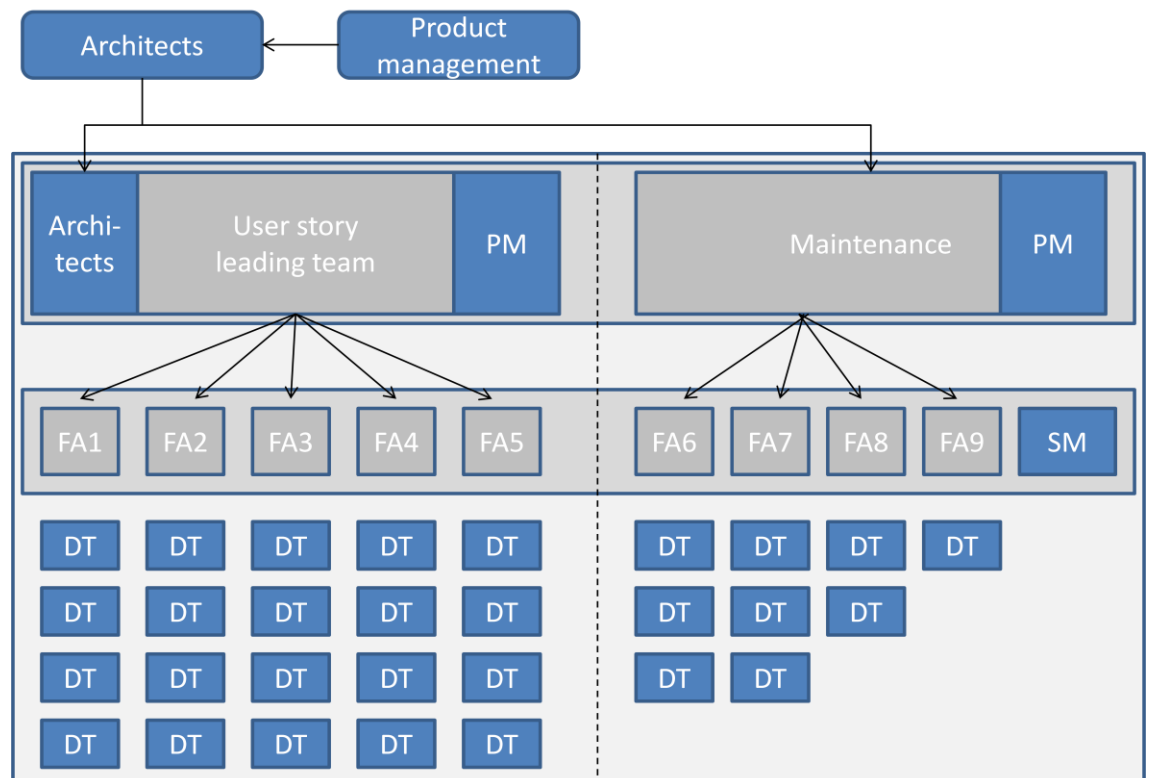
6.2. Ennustettavuuden parantaminen

Erilaiset iteraation suunnitellun sisällön ulkopuoliset työtehtävät todistetusti hidastavat varsinaista tuotekehitystä ja heikentävät ennustettavuutta. Tämä voisi olla vältettävissä jakamalla organisaatio ja ryhmät eri tavalla kuin nykyisten käytäntöjen mukaisesti tehdään.

Käytännössä organisaation toimintaan tulisi kaksi suurempaa muutosta. Näitä olisivat käyttäjätarinoiden laatiminen iteraation sisällöksi ja työresurssien jakaminen uudiskehityksen ja ylläpidon välillä.

Resurssien jakamista uudiskehityksen ja ylläpidon välillä tulisi muuttaa joustavammaksi samalla kun ne erotetaan toisistaan. Ehdotuksessa tämä on tehty seuraavalla tavalla. Iteraatioiden välillä projektijohto määrittelee painopistealueet ja siirtää tarvittaessa iteraation vaihtuessa osan tuotekehityksryhmistä ylläpidon puolelle tai toisinpäin. Oleellinen ero nykyiseen olisi se, että tuotekehityksryhmät tekevät ainoastaan tehtävälialta löytyviä tehtäviä. Ylläpidon puolella olevat ryhmät keskittyvät vikakorjauksiin, testitulosten analysointiin, vanhojen testitapausten päivittämiseen ja kaikkeen muuhun toimintaan joka ei suoraan liity uudiskehitykseen tai joka tulee kiireellisenä tehtävälialta ulkopuolelta.

Taulukko 8. Resurssien jakaminen organisaatiossa



Ylläpitoryhmillä on myös käytössään tehtävälialta, jonka kohteiden toteutumista ei kuitenkaan oleteta samalla tapaa kuin uudiskehityksen puolella. Tältä tehtävälialta voi löytyä esimerkiksi koodin uudelleenfaktorointia ja muita suunnitelmallisia parannuksia, joilla ei

kuitenkaan ole aikataulullisesti kiirettä. Aikataulut laaditaan ja niitä päivitetään sen mukaan miten tehtävää osoitetaan uudiskehitysryhmille. Näin saavutetaan parempi ennustettavuus uudiskehityksen toteutumisessa, sekä tarjotaan kaikille ryhmille parempi keskittymismahdollisuus tärkeimpiin tehtäviinsä.

Lisäksi otettaisiin käyttöön uudelleen scrum master -työrooli. Tämän roolin täyttäisivät linjamanagerit henkilöstöjohdon tehtäviensä ohella.

6.3. Iteraation sisällön määrittäminen

Iteraation sisällön määrittäminen tapahtuisi uudenaikaisessa työryhmässä. Suunnittelupäivän aluksi kokoontuisi ryhmä, jossa olisi edustajat arkkitehtuurin puolelta, projektijohto sekä edustus jokaisesta ryhmästä joka osallistuu uudiskehitykseen. Ryhmä käsittelee seuraavan iteraation sisällä tehtävät muutokset, jakaa ne ryhmille ja määrittelee tehtäville bisnestärkeyden lisäksi myös sisäisen tärkeyden joka käytännössä mahdollistaa työtehtävien priorisoinnin iteraation sisällä. Ryhmä kokoontuisi myös tarpeen tullen iteraation kuluessa scrum-palaverin tapaan joka siten mahdollistaa välitöntä huomiota kaipaavien seikkojen käsittelyn. Näitä olisivat esimerkiksi käyttäjätarinoiden tärkeysjärjestyksen muuttamiset tai eri osa-alueiden synkronointiin liittyvät seikat.

Uudiskehitystä tekevät ryhmät siirtyvät tämän jälkeen purkamaan saamansa tehtävät ja laativat ryhmälle ja sen aliryhmille listan tehtävistä ja määrittelevät myös tarvittaessa järjestyksen tehtäville aikaisemman vapaan listalta poiminnan sijaan. Uudiskehitystä tekevät ryhmät eivät tee iteraation aikana mitään tehtävälistan ulkopuolisia tehtäviä, lukuun ottamatta mahdollisia tapaamisia ja konsultointitehtäviä joihin on tehtävälistalle varattava aikaa oletusluontoisesti.

6.4. Työkalujen ajanmukaistaminen

Tuottavuuden, seurattavuuden ja ennustettavuuden parantamiseksi sekä henkilökohtaisten päivitysten nopeuttamiseksi tulisi Excel-taulukoista seurannan työvälineinä luopua täysin. Tilalle tulisi ottaa joku yrityksessä jo käytössä olevista ketterien menetelmien työkaluista (Jira, Accept 360) tai ottaa käyttöön joku nykyisen käytön ulkopuolinen sovellus (Version One, Kunagi, tms.).

Lisäksi tuotekehityksessä ja projektin hallinnassa käytetyt työkalut ja ohjelmistot tulisi katselmoida, mahdollisesti ulkopuolisen tahon toimesta. Näin mahdollistuisi puolueeton näkökulma työkaluohjelmistojen toimintaan ja päivitystarpeisiin.

6.5. Tulevia kehityskohteita

Tulosten esittely yrityksessä on sisältänyt useita eri kanavia. Tärkein tuotos on kuitenkin valmis tutkielma, johon työmenetelmien jatkokehityksessä voidaan viitata. Tämän lisäksi esiin tulleita asioita on

esitetty yrityksen sisällä työprosesseja työstävän ryhmän edustajalle, sekä materiaaliin tutustuminen on mahdollistettu asioihin välittömästi liittyvien sidosryhmien edustajille. Osa löydöksistä tiedotettiin välittömästi niiden esiin tullessa asianomaisille jotka saivat sen myötä mahdollisuuden nopeaan reagointiin. Käytännössä mahdollisten muutosten toteuttaminen kuitenkin jää yrityksen vastuulle.

Jatkossa toimintaa oletettavasti parantavana kohteena suositellaan kaikille ohjelmistoprojekteja johtaville tahoille tutustumista Steve McConnellin teokseen Ohjelmistotuotannon hallinta (en. Rapid Development), vähintäänkin kappaleeseen klassiset virheet.

Lisäksi suositellaan esimerkiksi mahdollisena tulevana opinnäytetyön aiheena tai vastaavana tutkimuksena tehtäväksi selvittää millainen ryhmän rakenne olisi toimivin laaja-alaisessa ohjelmistokehitysprojektissa jossa sovelletaan ketteriä menetelmiä. Nykyiset ryhmät ovat muodostuneet lähinnä kaveruuteen perustuen.

LÄHTEET

3T, Nortio J., 2012, Halpakoodi käy kalliiksi, Salainen tutkimus: Intialainen koodi on suomalaista kalliimpaa, http://www.3t.fi/artikkeli/uutiset/talous/salainen_selvitys_intialainen_koodaus_on_suomalaista_kalliimpaa (24.9.2012)

Agile Alliance (2001a), Manifesto for Agile Software Development, <http://www.agilealliance.org/the-alliance/the-agile-manifesto/> (28.8.2012)

Agile Alliance (2001b), Principles: The Agile Alliance, <http://www.agilealliance.org/the-alliance/the-agile-manifesto/the-twelve-principles-of-agile-software/> (28.8.2012)

Ambler S., 2002, Agile modeling: Effective Practices for eXtreme Programming and the Unified Process. John Wiley & Sons, Inc., new york

Balk A., Blogi työntekijöiden potentiaalin tunnistamisesta - Henkilöarviointi ja viisaat rekrytointipäätökset, 2010, <http://www.henkiloarviointi.fi/2010/nain-pienyritys-voi-kehittaa-rekrytointiprosessiaan> (1.5.2012)

Boehm B., 1981, Software Engineering Economics, Englewood Cliffs

Fowler M., 2006, Continuous Integration, <http://www.martinfowler.com/articles/continuousIntegration.html> (13.9.2012)

Hackman R. & Oldham G., 1980, Work Redesign, Reading, Addison-Wesley

Helakorpi S., Osaamisen johtaminen, <http://openetti.aokk.fi/sisu/TEEMAT%20AIHEALUEITTAIN/Kasvatus%20ja%20yhtykunta/Koulun%20johtaminen/osaamjohtam.pdf> (22.4.2012)

Hetzel B., 1993, Making Software Measurement Work: Building an Effective Measurement Program, New York, John Wiley & Sons

Hirota T. & Ikujiro N., 1986, The new new product development game, Harvard Business Review

Jones C., 1994, Assessment and Control of Software Risks, Englewood Cliffs

Kainulainen A., 2008, Agile-menetelmät, Opinnäytetyö, Jyväskylän Ammattikorkeakoulu, <https://publications.theseus.fi/handle/10024/17857> (29.11.2012)

Kamensky, M. 2008. Strategien johtaminen. Menestyksen timantti. Kariston Kirjapaino Oy Hämeenlinna.

Kitson D. & Masters S., 1993, An Analysis of SEI Software Process Assessment Results 1987-1991, Washington, D.C. IEEE Computer Society Press

Klinge T., Asiantuntijaorganisaation johtaminen., Intensio Newsletter, <http://www.asiakashyoty.fi/asiantuntijaorganisaation-johtaminen/> (1.5.2012)

Kokko, O., 2010, Pomon huono käytös vie voimasi, Taloussanomat, <http://www.taloussanomat.fi/tyo-ja-koulutus/2010/02/07/pomon-huono-kaytos-vie-voimasi/20101828/139> (1.5.2012)

Kopakkala A., 2005, Porukka, jengi, tiimi. Ryhmädynamiikka ja siihen vaikuttaminen. Helsinki, Edita

Larman C. & Vodde B., 2008, Scaling Lean & Agile Development: Thinking and Organizational Tools for Large-Scale Scrum, Addison-Wesley

Larman C. & Vodde B., 2010, Practices for Scaling Lean & Agile Development: Large, Multisite, and Offshore Product Development with Large-Scale Scrum, Addison-Wesley

Larson C. & LaFasto F., 1989, Teamwork: What must go right; What can go wrong. Newbury Park, Sage

Lecht C., 1967, The Management of Computer Programming Projects, American Management Association

Lehto-Kannisto E., Keva, Henkilöstöjohtamisen haasteita eilen, tänään ja huomenna, Henkilöstöjohtamisen seminaari 27.3.2012, http://www.keva.fi/SiteCollectionDocuments/Tyossa_jatkaminen/Uutiskirje/uutiskirje_2012_02/Lehto_Kannisto_270312_diat.pdf (23.4.2012)

Marciniak J. & Reifer D., 1990, Software Acquisition Management, New York, John Wiley & Sons

McConnell S., 2002, Ohjelmistotuotannon hallinta, Helsinki, Edita

Mielonen S. & Hintikka K., 1998, Web-palvelujen käytettävyys ja tuotanto, Mediastudioverkon tutkimukset ja selvitykset, <http://www.uiah.fi/mediastudio/survey4/> (17.2.2013)

Palkitsemisen tutkimusohjelma, Aalto-yliopiston perustieteiden korkeakoulu, <http://www.palkitseminen.tkk.fi/aiheet/lukijalle/index.htm> (1.5.2012)

Pihkala T., Oikarinen T., Pulkka-Stone L, Henkilöstöjohtamisen haasteet verkostoituneessa organisaatiossa, Lahden ammattikorkeakoulun julkaisu Sarja A tutkimuksia, osa 10., Juvenes Print, Tampere 2007, http://www.lamk.fi/material/or-bits-julkaisu_nettiin.pdf (23.4.2012)

Saviaro M., Kansainvälinen Henkilöstöjohtaminen, Return Ticket, <http://returnticket.fi/fi/hr/kansainvaelinehenkiloestoejohtaminen> (23.4.2012)

Schmidt T. ja Vanhala S., Henkilöstöjohtaminen Suomessa 1992- 2009, Aalto-yliopiston kauppakorkeakoulu, Aalto-Print 2010, <http://hsepubl.lib.hse.fi/pdf/hseother/b115.pdf> (23.4.2012)

Schwaber K., 2007, The Enterprise and Scrum, Microsoft Press
Sosiaali- ja Terveysministeriö, Työsuojelu Suomessa - tavoitteena terveellinen ja turvallinen työympäristö, <http://www.stm.fi/tyosuojelu> (1.5.2012)

Standish Group, 1994, Charting the Seas of Information technology, The Standish Group

Talouselämä, 2012, Pomo on ajattelun este, http://www.talouselama.fi/tyoelama/tosiasioita_tyoelamasta/pomo+on+ajattelun+este/a2154933 (12.11.2012)

Tampereen yliopiston henkilöstönkehittämisyksikkö heko, Perehdytyspolku, Tampereen yliopisto, <http://www.uta.fi/hallintokeskus/heko/perehdytyspolku/sivu.php?y=koulutus&s=henkilostokeh> (1.5.2012)

Työsuojeluhallinto, 2012, Työsuhdeasiat, <http://www.tyosuojelu.fi/fi/tyosuhdeasiat> (1.5.2012)

Työterveyslaitos, 2012, Ikäjohtaminen, http://www.ttl.fi/fi/tyoyhteiso_ja_esimiestyo/johtaminen_ja_esimiestyo/ika_johtaminen/sivut/default.aspx (1.5.2012)

Työterveyslaitos, 2012, Käyttäjälähtöiset toimistotilat, tilaratkaisut, sisäympäristö ja tuottavuus; TOTI-hankkeen loppuraportti. http://www.ttl.fi/fi/verkkokirjat/Documents/TOTI_loppuraportti.pdf (11.9.2012)

Työterveyslaitos, 2010, Muutosjohtaminen, http://www.ttl.fi/fi/tyoyhteiso_ja_esimiestyo/muutoksen_hallinta_ja_kehittaminen/muutosjohtaminen/sivut/default.aspx (1.5.2012)

Työterveyslaitos, 2010, Rekrytointiprosessi, http://www.ttl.fi/fi/tyoyhteiso_ja_esimiestyo/johtaminen_ja_esimiestyo/rekrytointi/rekrytointiprosessi/sivut/default.aspx (1.5.2012)

Wikipedia, <http://fi.wikipedia.org/wiki/Vesiputousmalli> (27.8.2012)

Wikipedia, http://en.wikipedia.org/wiki/Waterfall_model (27.8.2012)

Wikipedia,
http://en.wikipedia.org/wiki/Semi_Automatic_Ground_Environment
(27.8.2012)

Wikipedia, Työsuojelu, <http://fi.wikipedia.org/wiki/Ty%C3%B6suojelu>
(1.5.2012)

LÄHTEET (ANON) (LUOTTAMUKSELLINEN)

Yrityksen sisäiset lähteet on poistettu luottamuksellisina tästä versiosta.

Ketterän ohjelmistokehityksen julistus

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

Yksilöitä ja kanssakäymistä
enemmän kuin menetelmiä ja työkaluja

Toimivaa ohjelmistoa
enemmän kuin kattavaa dokumentaatiota

Asiakasyhteistyötä
enemmän kuin sopimusneuvotteluja

Vastaamista muutokseen
enemmän kuin pitäytymistä suunnitelmassa

Jälkimmäisilläkin asioilla on arvoa, mutta
arvostamme ensiksi mainittuja enemmän.

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
Dave Thomas

<http://agilemanifesto.org/iso/fi/>

Ketterät kehitysmenetelmät

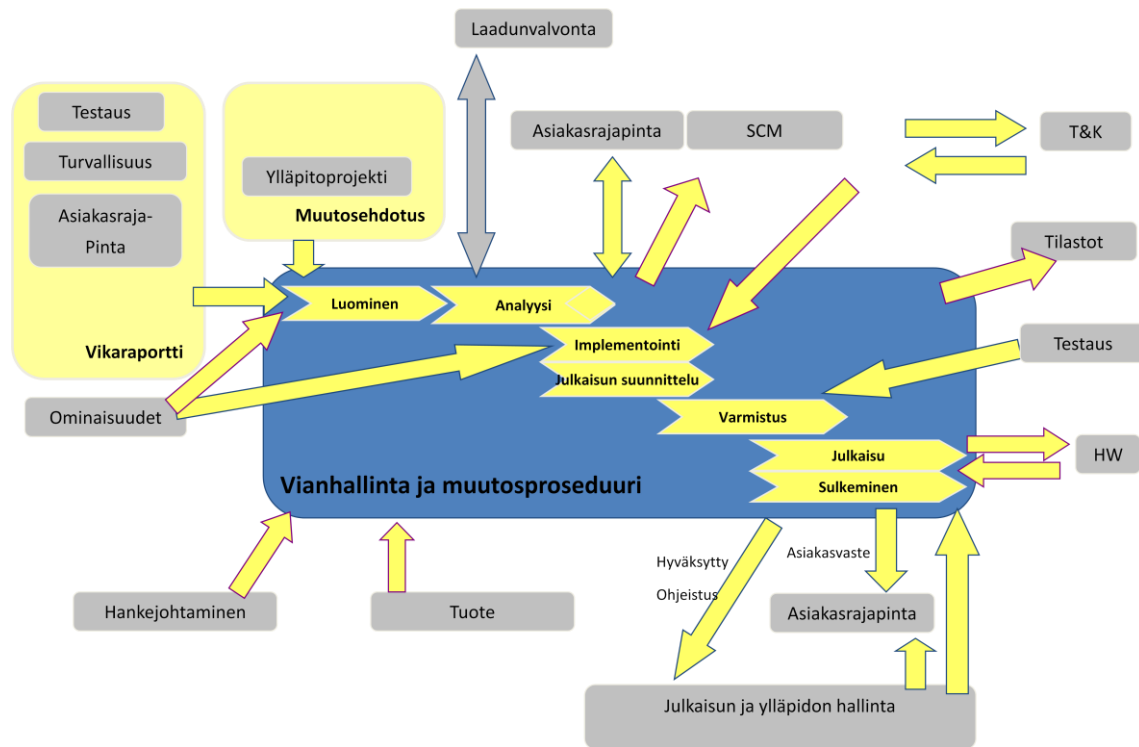
Antti Kainulainen esitti seuraavanlaisen tiivistelmän erilaisten ketterien kehitysmenetelmien eroista ohjelmistokehityksessä opinnäytetyössään Jyväskylän Ammattikorkeakoululle vuonna 2008.

Taulukko 7. Vertailu ketteristä menetelmistä (Kainulainen 2008)

	Scrum	XP	ASD	Crystal	FDD	DSDM
Suunnitteluvaiheen laajuus	Osin määritelty, lyhytkestoinen, mutta riittävä	Osin määritelty, mahdollisimman lyhytkestoinen, mutta riittävä	Tarkoin määritelty, kattava ja pitkäkestoinen	Ei määritelty	Tarkoin määritelty, melko kattava, melko pitkäkestoinen	Tarkoin määritelty, kattava ja melko pitkäkestoinen
Iteratiivisuus	Kevyt suunnittelu, toteutusvaihe iteratiivinen	Kevyt suunnittelu, toteutusvaihe iteratiivinen	Jaksottainen suunnittelu- vaihe, toteutusvaihe iteratiivinen	Heikosti määritelty, ainakin toteutus- vaihe iteratiivinen	Jaksottainen suunnittelu- vaihe, toteutusvaihe iteratiivinen	Jaksottainen suunnittelu- vaihe, toteutusvaihe iteratiivinen
Itäraation pituus	30 päivää	2-4 viikkoa	4-10 viikkoa	2-4 kuukautta	2 viikkoa	ei määritelty
Asiakkaan osallistuminen	Suunnittelu, itäraatioiden suunnittelu, tulokset	Päivittäinen, tiimin aktiivinen jäsen	Suunnittelu, itäraatioiden tulosten katselmointi	Ei tarkoin määritelty, mutta aktiivinen	Suunnittelu, itäraatioiden osalta ei määritelty	Suunnittelu, itäraatioiden tulosten katselmointi
Asiakkaalta vaadittu tekninen osaaminen	Ei erityistä vaatimusta	Korkea	Ei erityistä vaatimusta	Ei määritelty	Ei määritelty	Ei erityistä vaatimusta
Prosessin määrittely	Vaiheet tarkasti määritelty, työskentelytavat vapaat	Vaiheet yllimalkaisesti määritelty, vaihetuotteet ja työskentelytavat määritelty yksityiskohtaisesti	Vaiheet määritelty, vapaat työskentely- tavat	Ei määritelty	Vaiheet ja vaihetuotteet yksityiskohtai- sesti määritelty, työskentelytavat vapaat	Vaiheet ja vaihetuotteet yksityiskohta- isesti määritelty, työskentelytav at vapaat
Vaihetuotteiden määrä	4	6	5	Ei määritelty	5	7
Soveltuva henkilömäärä	Ei rajoituksia, skaalautuvuus hoidettu	Pienet, alle 10 hengen projektit	Ei rajoituksia, skaalautuu hyvin	Eri prosessit erikokoisille projekteille	Ei rajoituksia, suunniteltu suuremmille projekteille	Ei rajoituksia, käytetään yleensä suuremmissa projekteissa

VIANHALLINTA

Yrityksen tuotekehityksessä käytetty viankorjaamisen ja jälkikäteen tapahtuvien muutosten tekemisen prosessikaavio. Tuotekehitysryhmien panos viankorjauksessa sisältää oheisen kaavion ydinkohdat analyysi, implementointi ja varmistus.



Kaavio 1. Yrityksen vikakorjausproseduuri (Anon 1)