



Haaga-Helia
ammattikorkeakoulu Oy

Käyttäjäkeskeinen asiakastietojärjestelmän kehittäminen

Enni Koskenvuori

Master-opinnäytetyö
ICT-palvelut ja tietojärjestelmät
2021



Tekijä Enni Koskenvuori	
Suuntautuminen ICT-palvelut ja tietojärjestelmät	
Opinnäytetyön nimi Käyttäjakeskeinen asiakastietojärjestelmän kehittäminen	Sivumäärä + liitesivumäärä 94 + 12
<p>Valtaosa tietojärjestelmän elinkaaren kustannuksista kuluu sen ylläpitoon. Ylläpitoaika on tietojärjestelmään tehtyjen investointien merkittävintä takaisinmaksuaikaa, joten kasvattamalla tietojärjestelmän elinikää voidaan kasvattaa myös sillä saavutettavia hyötyjä ja sen kustannustehokkuutta. Tietojärjestelmän ylläpidon kesto voidaan kasvattaa hyvin organisoidulla ja tehokkaalla ylläpidolla sekä kehittämällä tietojärjestelmää käyttäjakeskeisesti. Käyttäjakeskeisellä kehittämisellä voidaan kehittämisessä huomioida käyttäjien näkökulma, varmistaa tietojärjestelmän vastaavuus käyttäjien työtehtäviin ja kasvattaa tietojärjestelmän käytettävyyttä.</p> <p>Tämän opinnäytetyön tarkoituksena on tutkia käyttäjakeskeisyyden toteutumista tietojärjestelmän kehittämisen eri vaiheissa. Opinnäytetyön kehittämistyön tavoitteena on luoda Rikosseuraamuslaitoksen asiakastietojärjestelmän kehittävän ylläpidon prosessikuvaus. Prosessikuvausten lähtökohtana on käyttäjakeskeisyyden lisääminen kyseisen asiakastietojärjestelmän ylläpitovaiheessa. Rikosseuraamuslaitos on vankeusrangaistusten ja yhdyskuntaseuraamusten täytäntöönpanoviranomainen, joka toimii oikeusministeriön alaisuudessa.</p> <p>Opinnäytetyön teoreettinen viitekehys koostuu käytettävyydestä ja käyttäjakeskeisyydestä. Opinnäytetyön tutkimuksellisen kehittämisen lähestymistapa on konstrukttiivinen tutkimus. Tutkimusmenetelmänä käytettiin haastattelua. Tietojärjestelmän suunnitteluun osallistuneista henkilöistä haastateltiin kolmea henkilöä ja tietojärjestelmän tulevista käyttäjistä kuutta henkilöä. Lisäksi toteutettiin yksi ideointityöpaja.</p> <p>Tutkimuksen lopputulosten perusteella käyttäjakeskeisyyttä voidaan kasvattaa tarjoamalla käyttäjille mahdollisuus tuoda itse näkemyksiään esiin sekä aktiivisesti selvittämällä käyttäjien näkemyksiä. Kehittämiseen osallistumisen tulee olla mahdollisimman helppoa. Käyttäjät kokivat, etteivät pysty esittämään tarpeita ennen kuin itse pääsevät kokeilemaan tietojärjestelmän käyttöä. Tätä haastetta voi olla tehokasta ratkaista toteuttamalla protoilua jossain muodossa. Käyttäjät kokivat haastavana myös tietojärjestelmän monimutkaisuuden ja hajanaisuuden, johon tulee kiinnittää huomiota jo tietojärjestelmän kehittämisen alkuvaiheissa. Tietojärjestelmän tehokkaan ylläpidon edellytykset rakennetaan jo ylläpitoa edeltävissä vaiheissa varmistamalla tietojärjestelmän korkea ylläpidettävyyttä.</p>	
Asiasanat Tietojärjestelmä, käyttäjakeskeisyys, ylläpidettävyyttä, ylläpito	

Sisällys

1	Johdanto	1
2	Kehittämistyön rakenne ja tavoitteet.....	5
2.1	Tutkimuksen vaiheet	5
2.2	Tutkimuskysymykset ja tavoitteet.....	6
2.3	Lähestymistapa.....	8
2.4	Tutkimusmenetelmät.....	10
2.5	Aineiston analyysi	13
2.6	Kohdeorganisaatio	15
2.7	Käsitteet.....	15
3	Tietojärjestelmän käytettävyys	16
3.1	Käytettävyys	18
3.2	Saavutettavuus	19
3.3	EU:n tietosuoja-asetus	19
3.4	Käyttökokemus	20
3.5	Käyttäjäkeskeinen suunnittelu.....	21
4	Tietojärjestelmän käyttäjäkeskeinen kehittäminen.....	24
4.1	Käyttäjä ja asiakas	24
4.2	Käyttäjätypit	24
4.3	Käyttäjien osallistaminen tietojärjestelmän kehittämiseen	25
4.4	Käyttäjäkeskeisyys tietojärjestelmien kehittämisen eri menetelmissä.....	27
4.5	Käyttäjäkeskeisyys tietojärjestelmän kehittämisen eri vaiheissa.....	29
4.5.1	Esitutkimus.....	30
4.5.2	Vaatimusmäärittely.....	31
4.5.3	Määrittely	33
4.5.4	Suunnittelu	35
4.5.5	Toteutus ja testaus.....	36
4.5.6	Käyttöönotto.....	37
4.6	Dokumentointi.....	38
4.7	Käyttäjäkeskeisen kehittämisen haasteet.....	40
4.7.1	Kehittämistyön roolit.....	40
4.7.2	Viestintä ja vuorovaikutus.....	41
4.7.3	Käyttäjien tavoittaminen	42
5	Tietojärjestelmän ylläpito.....	44
5.1	Ylläpidon haasteet	45
5.2	Ylläpidon organisointi	46
5.3	Muutostarpeiden aiheuttajat.....	49

5.4	Muutostarpeiden hallinta ja analysointi.....	51
5.5	Dokumentaation merkitys ylläpidossa	54
5.6	Käyttäjakeskeisyys tietojärjestelmän ylläpitovaiheessa	56
5.7	Tietolähteet	58
5.8	Ylläpidettävyystekijät.....	60
5.9	Tietojärjestelmän laatu	62
6	Kehittämistyön toteutus	63
6.1	Aiheeseen perehtyminen	63
6.2	Suunnittelu.....	64
6.3	Toteutus.....	64
6.3.1	Kehittävän ylläpidon tiedonhankinnan suunnittelun työpaja.....	64
6.3.2	Haastattelut.....	69
6.3.3	Käyttäjakeskeisyyden toteutuminen ennen ylläpitovaihetta	71
6.3.4	Tietojärjestelmän monimutkaisuus	73
6.3.5	Käyttäjakeskeisyyden haasteet	75
6.3.6	Käyttäjakeskeisyys ylläpitovaiheessa	77
6.3.7	Käyttäjätiedon vaikutus ja merkitys	79
6.4	Kehittävän ylläpidon prosessikuvaus.....	81
7	Yhteenveto ja pohdinta.....	87
7.1	Lopputulokset	87
7.2	Pohdinta.....	89
	Lähteet	92
	Liitteet.....	95
	Liite 1. Suunnittelijan haastattelurunko	95
	Liite 2. Käyttäjien haastattelurunko.....	96
	Liite 3. Haastateltavat.....	97
	Liite 4. Teemat käyttäjakeskeisyyden toteutuminen ennen ylläpitovaihetta ja tietojärjestelmän monimutkaisuus haastatteluissa	98
	Liite 5. Käyttäjakeskeisyyden haasteiden teemat haastatteluissa.....	101
	Liite 6. Ylläpitovaiheen käyttäjakeskeisyyden teemat haastatteluissa.....	104
	Liite 7. Käyttäjätiedon vaikutus ja merkitys haastatteluissa.....	106

1 Johdanto

Menestyäkseen organisaation on jatkuvasti kehitettävä toimintaansa ja tehostettava prosessejaan. Nykyaikana toiminnan kehittämisen edellytyksenä on organisaation ydinprosesseista syntyvän tiedon hallinta, joka mahdollistaa toiminnan seuraamisen. Olennainen osa organisaation tiedon hallintaa ja toimintaa on tietojärjestelmä, joka on yhdistelmä inhimillistä tietojenkäsittelyä ja tietojärjestelmäarkkitehtuuria. Tieto ja tietojohdaminen ovat organisaation menestymisen edellytyksiä. (Stenberg 2006, 8.)

Tietojärjestelmillä on organisaatiokohtaisesti erilaisia käyttötarkoituksia ja ne voivat olla tyypeiltään erilaisia. Tietojärjestelmien käytön syyt voidaan karkeasti jakaa perusoperatiivisten toimintojen tukemiseen, johdon päätöksenteon tukemiseen ja kilpailuedun saavuttamiseen. Useille organisaatioille tietojärjestelmät ovat välttämättömiä, jotta organisaation toiminta on edes mahdollista. Yhteistä organisaation tietojärjestelmille on kuitenkin se, että niiden käyttäminen ei ole itsetarkoitus, vaan niiden tarkoitus on palvella organisaatiota, tukea ja tehostaa työtehtävien suoritusta ja auttaa organisaatiota saavuttamaan tavoitteensa. (Pohjonen 2002, 10-11.)

Organisaation kehitystyö aiheuttaa usein muutoksia toimintaan, organisaatioon ja tietojärjestelmään, mutta tietojärjestelmän kehittäminen toisaalta myös mahdollistaa toiminnan kehittämisen. Kehitystyön lähtökohtaa oleellisempaa onkin mahdollistaa jatkuva kehittäminen, jonka myötä toimintamallit ja työtavat uudistuvat. (Stenberg 2006, 9.)

Tietojärjestelmien elinkaari koostuu useimmiten esitutkimuksesta, vaatimusmäärittelystä, suunnittelusta, toteutuksesta ja käyttöönotosta, joka sisältää myös käyttäjien koulutuksen ja hyväksytyyn käyttöönottestauksen, joka on ylläpitovaiheeseen siirtymisen edellytys (Stenberg 2006, 105-106). Käyttöönoton jälkeen alkaa tietojärjestelmän pisin ja kallein vaihe eli ylläpito. Käytännössä eniten huomiota saava ylläpidon osa-alue on sovelluksen ylläpito, josta suurin osa on kehittävää ylläpitoa eli uusien ominaisuuksien kehittämistä. (Stenberg 2006, 114.) Ketterien kehitysmenetelmien yleistyessä eri vaiheita ei enää useimmiten eroteta toisistaan tarkkarajaisesti, vaan esimerkiksi ylläpitovaiheessa tehdään myös jatko- ja uuskehitystä. Osa tietojärjestelmästä voi olla kehitysvaiheessa, vaikka toinen osa on jo ylläpitovaiheessa.

ICT-hankinnat muodostivat suurimman hankintakategorian valtion julkisista hankinnoista vuonna 2020. ICT-palveluja hankittiin yli miljardilla eurolla. (Tutki hankintoja 2021.)

Tietojärjestelmän ylläpitoaika on siihen tehtyjen investointien merkittävintä takaisinmaksuaikaa, joten ylläpitovaiheen pitkäkestoisuus kasvattaa tietojärjestelmällä saavutettavia hyötyjä (Stenberg 2006, 91). Tietojärjestelmän hyvin organisoidulla ja

tehokkaalla ylläpitovaiheella pystytään kasvattamaan tietojärjestelmän käyttöikä ja sen myötä sen kustannustehokkuutta merkittävästi.

Käyttäjakeskeisen kehittämisen lähtökohtana on käyttäjien toiveet ja tarpeet ja sen tavoitteena on kehittää tuote, joka mahdollisimman hyvin vastaa niihin. Samaan aikaan kun tietojärjestelmistä on tullut yhä laajempia ja monimutkaisempia organisaationaalisia kokonaisuuksia, on käyttäjien osallistumista heille suunnattujen tietojärjestelmien suunnitteluun pidetty koko ajan tärkeämpänä tietojärjestelmähankkeen onnistumisen edellytyksenä. Käyttäjät tuntevat organisaation ja sen tehtävät ja he kykenevät näkemään tietojärjestelmän muutokset ja sen vaikutukset laajempina kokonaisuutena.

Tutkimusten mukaan peruskäyttäjistä vain kaikkein tyytyväisimmät ja tyytymättömimmät käyttäjät ovat aktiivisia antamaan palautetta tai osallistuvat omasta aloitteestaan kehittämiseen. Mikäli kehittämisessä halutaan tavoittaa myös muut käyttäjät, pitää heitä jollain tavoin kannustaa palautteen antoon. (Merisalo-Rantanen 2011, 52-53.) Palautteen keräämisen lisäksi tärkeää on myös kerätyn palautteen hyödyntäminen. Mikäli kerättyä palautetta ei hyödynnetä kehittämisessä, on palautteen kerääminen vain turha kulu. (Merisalo-Rantanen 2011, 16.) Käyttäjien osallistuminen tuotteen kehitykseen on sen menestymisen kannalta tärkeää, mutta organisaatioiden on mahdollista hyödyntää heille jo kertynyttä käyttäjätietoa ilman käyttäjien varsinaista osallistumista. Se kuitenkin edellyttää vakaata harkintaa sekä käyttäjätiedon tunnistamista ja hyviä käytäntöjä sen hyödyntämiseksi. Sen sijaan, että organisaatiot käyttävät resursseja käyttäjätutkimuksen tekemiseen, he voivat hyödyntää heille jo kertynyttä käyttäjätietoa. Käyttäjätieto voi olla kertynyt eri puolille organisaatiota, esimerkiksi tekniseen tukeen ja tuotekehitykseen ja on tärkeää, että tiedot kootaan yhdeksi käyttäjätiedoksi. (Savolainen 2021, 63-68.)

Tämän opinnäytetyön aiheena on kehittävän ylläpidon eli uusien ominaisuuksien kehittämisen organisointi ja tietojärjestelmän kehittämisen käyttäjakeskeisyyden kasvattaminen ylläpitovaiheessa. Opinnäytetyön kohteena on Rikosseuraamuslaitoksen asiakastietojärjestelmä. Käyttäjät ovat osallistuneet sen esitutkimukseen, vaatimusmäärittelyyn, määrittelyyn, suunnitteluun, toteutukseen ja testaukseen. Näin on haluttu varmistaa, että uusi tietojärjestelmä vastaa työtehtävien vaatimiin tarpeisiin ja että suunnittelussa on pystytty ottamaan huomioon loppukäyttäjän näkökulma. Vaikka osan suunnitteluun osallistuneiden käyttäjien osallistuminen on ollut intensiivistä ja kokoaikaista, niin suunnitteluun osallistuneita käyttäjiä on ollut lukumäärällisesti hyvin pieni määrä suhteessa kaikkiin asiakastietojärjestelmän loppukäyttäjiin.

Asiakastietojärjestelmän loppukäyttäjiä on paljon ja heidän taitonsa ja tapansa käyttää teknologiaa vaihtelevat suuresti. Suurissa organisaatioissa, kuten Rikosseuraamuslaitos,

yksittäinen työntekijä tuntee harvoin kaikki organisaation tehtävät tai toimintojen rakenteen niin hyvin, että hän osaisi ottaa huomioon kaikki eri työtehtävien ja yksiköiden erityispiirteet. Opinnäytetyön tekijä on itse työskennellyt asiakastietojärjestelmän suunnittelijana tuoden asiakastietojärjestelmän toteuttamiseen loppukäyttäjän näkökulmia ja tarpeita. Opinnäytetyö perustuu sen tekijän ajatukseen siitä, että ylläpidon käyttäjäkeskeinen organisointi on tehokkain tapa pystyä jatkokehittämään kohdeasiakastietojärjestelmä vastaamaan kaikkiin sille asetettuihin tavoitteisiin.

Uuden asiakastietojärjestelmän todellinen käytettävyys ja sen hyödyt sekä kehittämisen kohteet selviävät vasta kun kaikki loppukäyttäjät ottavat sen käyttöön. Jotta uusi asiakastietojärjestelmä pystyisi vastaamaan kaikkiin sille asetettuihin tavoitteisiin ja sen käytettävyyttä ja tuomia hyötyjä pystyttäisiin kasvattamaan, on asiakastietojärjestelmän ylläpidon kehityksenkin oltava käyttäjäkeskeistä. Opinnäytetyö toteutettiin laadullisena tutkimuksena ja on Rikosseuraamuslaitoksen toimintaa kehittävä työ, jonka kehittämistyönä suunniteltiin asiakastietojärjestelmän kehittävä ylläpidon organisointi ja laadittiin sitä kuvaava prosessikuvaus. Organisoinnin lähtökohtana oli kasvattaa käyttäjäkeskeisen kehittämisen osuutta Rikosseuraamuslaitoksen asiakastietojärjestelmän kehittämisessä osallistamalla yhä isompi osa käyttäjistä uusien ominaisuuksien kehittämiseen. Käyttäjien osallistuminen ylläpitovaiheeseen luo käyttäjäkeskeisen kehittämisen jatkumon asiakastietojärjestelmän suunnittelusta käyttöönottoon ja sen jälkeen ylläpitoon.

Prosessikuvauksen luomisen pohjaksi tarvittiin tietoa asiakastietojärjestelmän käyttäjäkeskeisyydestä sen aiemmissa kehittämisen vaiheissa ja käyttäjäkeskeisyyden haasteista ja onnistumisista. Tutkimuksen aineisto hankittiin haastattelemalla asiakastietojärjestelmän suunnittelijoita ja sen tulevia käyttäjiä. Tutkimuksen tuloksena havaittiin, että onnistunein tapa varmistaa ylläpidon organisointi käyttäjäkeskeisesti sekä ylläpidon resurssien mahdollisimman tehokas käyttö on ottaa tietojärjestelmän käyttäjäkeskeisyys sekä ylläpitovaihe huomioon jo kaikissa tietojärjestelmän kehittämisen aikaisemmissa vaiheissa. Systeemyöhön tuleekin suhtautua kokonaisuutena, jonka yksi olennainen osa on ylläpito. Tutkimustulokset osoittivat myös, että käyttäjäkeskeisyyden toteutumiseksi käyttäjien osallistuminen on oltava mahdollisimman helppoa, käyttäjien näkemyksiä on aktiivisesti selvitettävä ja jotta käyttäjien tarpeet saadaan selville tarpeeksi tarkalla tasolla, on hyödyllistä toteuttaa toteutuksen yhteydessä protoilua jollain tavalla.

Opinnäytetyöraportin luvussa kaksi on kuvattu kehittämistyön rakenne, tavoitteet, tutkimuskysymykset, tutkimusmenetelmät, kohdeorganisaatio ja käytetyt käsitteet. Luvut kolme, neljä ja viisi muodostavat raportin tietoperustan, joka käsittelee käytettävyyttä sekä

tietojärjestelmän käyttäjäkeskeistä kehittämistä sen kaikissa vaiheissa. Luvussa kuusi on kuvattu kehittämistyön toteutus ja luvussa seitsemän opinnäytetyön lopputulokset ja pohdinta.

2 Kehittämistyön rakenne ja tavoitteet

Tämän opinnäytetyön tarkoituksena on tutkia käyttäjakeskeisyyden merkitystä ja käyttäjien osallistumista tietojärjestelmän kehittämiseen tietojärjestelmän kehittämisen eri vaiheissa. Opinnäytetyön tavoitteena ja kehittämistyönä on suunnitella ja viedä käytäntöön tietojärjestelmän kehittävän ylläpidon prosessikuvaus. Kehittävän ylläpidon organisoinnin lähtökohdaksi on valittu käyttäjakeskeisyyden lisääminen, koska kehittävän ylläpidon tavoitteena on pystyä kehittämään tietojärjestelmää niin, että se yhä paremmin tukee käyttäjien työprosesseja ja mahdollistaa organisaation toiminnan kehittymisen ja kehittämisen. Jotta tietojärjestelmä voi tukea käyttäjien työprosesseja, on kehittävän ylläpidon organisoinnissa otettava huomioon tiedon hankkiminen koskien tietojärjestelmän käyttöä. Kehittämistyönä luotavalla prosessikuvauksella pyritään saavuttamaan käyttäjiä ja käyttöä koskevaa tietoa käytettävyyden ja käyttökokemuksen väliltä. Mahdollisuudet vaikuttaa aitoon käytettävyyteen tietojärjestelmän käyttöönoton jälkeen ovat rajalliset ja käyttökokemuksen muuntaminen konkreettisiksi muutosehdotuksiksi haastavaa. Opinnäytetyön kohteena on Rikosseuraamuslaitoksen asiakastietojärjestelmä. Kohdeorganisaatio ja tietojärjestelmä on kuvattu kappaleessa 2.6.

2.1 Tutkimuksen vaiheet

Tutkimustyön tarkoituksena on tavallisesti tuottaa uutta tietoa sekä tieteen edistämisen että käytännön kannalta (Järvinen & Järvinen 2011, 3). Suunnitelmallinen ja tavoitteellinen tutkimus koostuu aiheeseen perehtymisestä, suunnitelman laadinnasta, tutkimuksen toteuttamisesta ja tutkimusselosteen laadinnasta. Näiden tunnistettujen vaiheiden sisällä tutkimusta on käytännössä monenlaista ja myös tutkimusprosessi voidaan kuvata monin eri tavoin. (Hirsjärvi, Remes & Sajavaara 2007, 63.) Tutkimusprosessin laadinta helpottaa tutkimuksen ja kehittämistyön jäsentämistä, vaikka tutkimusprosessia ei käytännössä voida tarkkarajaisesti jakaa selkeästi eri vaiheisiin. Usein työskentely tapahtuu samanaikaisesti eri vaiheissa ja prosessissa voidaan myös palata aiempiin vaiheisiin. Tämän laadullisen tutkimuksen tarkoituksena on pääasiallisesti tuottaa uutta tietoa, jolle on käyttöä käytännössä. Tutkimuksen vaiheet on laadittu pohjautuen tutkimussuunnitelmaan huomiota kiinnittävään tutkimuksen kulkuun (Hirsjärvi ym. 2007, 65) sekä tutkimuksellisen kehittämistyön prosessiin (Ojasalo, Moilanen & Ritalahti 2014, 23-26). Tutkimus on nelivaiheinen; aiheeseen perehtyminen, suunnittelu, toteutus ja raportointi.

Tutkimuksen tekeminen käynnistyy asiantilasta, johon haluamme vastauksen tai ongelmasta, jonka haluamme ratkaista. Askarruttavasta asiantilasta syntyy tutkimuksen idea. Idean jälkeen tutustutaan aihepiiriin kirjallisuuteen ja aiempaan tutkimustietoon ja

muodostetaan alustava teoriatausta. Tämän avulla pystytään kuvaamaan tutkimuksen lähtökohdat ja muodostamaan alustava tutkimuskysymys.

Tutkimuksen lähtökohtien kuvaamisen jälkeen suunnitteluvaiheessa täsmennetään tutkimuskysymyksiä peilaten aiempaan kirjallisuuteen sekä tutkimuksiin ja valitaan, millaisella lähestymistavalla kehitettävää aihetta lähestytään tehokkaimmin.

Lähestymistavan lisäksi tutkitaan ja valitaan, mitkä menetelmät soveltuvat parhaiten käytettäväksi juuri kyseiseen tilanteeseen ja tutkimukseen.

Toteutusvaiheessa kerätään tutkimuksen aineistot, analysoidaan ne sekä tehdään tutkimuksen kehittämistyö valittujen menetelmien mukaisesti.

Raportointivaihe sisältää tutkimuksen loppuraportin viimeistelyn ja julkistamisen.

Raportointivaiheessa tehdään myös kaikkiin tutkimuksiin kuuluva tutkimuksen luotettavuuden arviointi käsittelemällä tutkimuksen reliäabelius ja validius. Reliäabelius tarkoittaa tutkimuksen mittaustulosten toistettavuutta eli kykyä tuottaa ei-sattumanvaraisia tuloksia ja validius eli pätevyys tarkoittaa valitun menetelmän kykyä mitata sitä, mitä oli tarkoituskin mitata. Laadullisessa tutkimuksessa validius merkitsee henkilöiden, paikkojen ja tapahtumien kuvausten ja niihin liitettyjen tulkintojen ja selitysten yhteensopivuutta. Luotettavuutta voidaan kasvattaa selostamalla tutkimuksen toteutus kaikkine vaiheineen tarkasti. (Hirsjärvi ym. 2007, 226-227.)

2.2 Tutkimuskysymykset ja tavoitteet

Hirsjärven, Remeksen ja Sajavaaran mukaan (2007, 69) tutkimuksen aiheen valinnan jälkeen tutkimukselle tulisi löytää johtoajatus. Johtoajatus ohjaa teorian valintaa, aineiston hankintaa ja lopputuloksen arviointia. Johtoajatukseen kiteytyy tutkimuksen perusidea, tarkoitus ja näkökulma (Hirsjärvi ym. 2007, 41). Myös tutkimuksen perustana toimiva päätutkimustehtävä voidaan kiteyttää tutkimuksen johtoajatuksen mukaisesti. Etenkin kvalitatiivisessa tutkimuksessa puhutaan mieluummin tutkimustehtävästä kuin tutkimusongelmasta. (Hirsjärvi ym. 2007, 122.) Se, puhutaanko tutkimusongelmasta vai tutkimustehtävästä on enemmänkin tutkimuskohtaisesti ratkaistava asia kuin laadullista tutkimusta kattava terminologinen asia (Tuomi & Sarajärvi 2018, 105).

Perinteisesti etenevässä tutkimuksessa tutkimuksen ongelmat tai tehtävät tulee harkita ja muotoilla selkeästi ennen varsinaiseen aineiston keruuseen ryhtymistä.

Päätutkimustehtävän analysointi voi tuottaa tutkimukselle alatutkimustehtäviä, mutta eroa pää- ja alatutkimustehtävien välillä ei ole välttämätöntä tehdä. (Hirsjärvi ym. 2007, 121-122.) Konstruktiivisella tutkimuksella voidaan organisaatiossa vastata esimerkiksi sellaisiin tutkimus- ja kehittämiskysymyksiin, joiden avulla selvitetään, voidaanko jokin tehtävä

tehdä tehokkaammin tai voidaanko jokin työtä tukeva prosessi muuttaa vastaamaan paremmin organisaation tarpeisiin (Ojasalo ym. 2014, 66). Tässä opinnäytetyössä on kaksi tutkimuskysymystä, joihin tutkimuksen tekemisellä pyritään vastaamaan.

Tutkimuskysymys 1: Miten asiakastietojärjestelmän kehittävän ylläpidon osa-alue organisoidaan käyttäjäkeskeisesti?

Rikosseuraamuslaitoksen asiakastietojärjestelmän kehittävän ylläpidon yhtenä tarkoituksena on kehittää asiakastietojärjestelmää käyttäjien käytön haasteiden perusteella käytettävämmäksi ja paremmin käyttäjien tehtäviä tukevaksi. Opinnäytetyön ensimmäinen tutkimuskysymys on johdettu kehittävän ylläpidon organisoinnin tarpeesta. Asiakastietojärjestelmän kehittävän ylläpidon organisoinnin lähtökohtana tulee olla käyttäjäkeskeisyys, jotta kehittävän ylläpidon tavoitteet saavutetaan.

Tutkimuskysymys 2: Miten kehittävän ylläpidon resurssit kohdennetaan tehokkaasti?

Opinnäytetyön toinen tutkimuskysymys on johdettu asiakastietojärjestelmän ylläpidon tehokkuudesta. Ylläpidon resurssit ovat rajalliset ja ne tulee kohdistaa toimenpiteisiin, joilla voidaan saavuttaa eniten hyötyjä suhteessa käytettyihin resursseihin.

Rikosseuraamuslaitoksen kehittävän ylläpidon kehittämistoimenpiteet eivät kohdistu eksklusiivisesti vain sovelluksen toimintaan, vaan ratkaisu kehittämiskohtiin voi olla myös esimerkiksi ohjeistuksen tarkennus, koulutus tai sovelluksen ohjaustietojen muutos. Kehittävän ylläpidon organisoinnissa on huomioitava se, miten arvioidaan ylläpidon toimenpiteillä saavutettavat hyödyt. Ensimmäiseen tutkimuskysymykseen pyritään vastamaan tutkimuksen aineistolla ja toiseen kehittämistyön produktilla.

Tutkimuskysymyksiä perustana on ollut opinnäytetyön tekijän johtoajatus ”käyttäjakeskeisyys tietojärjestelmän ylläpidossa on tehokkaan ylläpidon edellytys”. Tutkimuskysymyksiä muotoiluun vaikuttivat myös opinnäytetyön tekijän ajatukset, joiden mukaan 1) kartuttamalla ja analysoimalla tietoa tietojärjestelmän käytöstä voidaan parantaa sen käytettävyyttä ja käyttöikää sekä 2) tietojärjestelmän käytön haasteita analysoimalla niihin voidaan löytää kustannustehokkain ratkaisu.

Opinnäytetyön tavoitteena on tuottaa tutkimuksellisen kehittämistyön produktina Rikosseuraamuslaitoksen asiakastietojärjestelmän kehittävän ylläpidon organisointia kuvaava prosessikuvaus. Prosessikuvauksessa kuvataan, millä tavoin kerätään tietoa siitä, mitkä asiakastietojärjestelmän osat tai toiminnallisuudet käyttäjät kokevat haastaviksi tai vaikeiksi käyttää ja miten näitä tietoja analysoimalla voidaan johtaa asiakastietojärjestelmän muutosehdotukset sen käytön helpottamiseksi ja tehostamiseksi. Prosessikuvauksen osaksi koostetaan yleisiä periaatteita käyttäjäkeskeisyydestä.

Produktina luotava prosessikuvaus koskee kohdeorganisaation ylläpitovaihetta ja on oletettavaa, että prosessikuvausta tullaan sen käyttöönoton jälkeen kehittämään paremmaksi sen käytön kokemusten perusteella. Muutoksissa tulee kuitenkin ottaa huomioon luodut periaatteet, jotka ovat yleistettävissä muihin organisaatioihin ja tuotteisiin.

Prosessikuvaus sisältää kuvauksen käytännön prosessista, jonka vaiheet ovat: 1. Tiedonhankinta siitä, mitkä asiat asiakastietojärjestelmän käytössä aiheuttavat eniten haasteita sen käyttäjille ja miksi. 2. Hankitun tiedon analysointi kattavasti eri näkökulmista sekä vaikuttavuuden arviointi. 3. Tiedon muuttaminen muutostarpeiksi. 4. Muutostarpeiden priorisointi. 5. Muutostarpeiden hallinta sekä ratkaisujen valinta. Kehittävän ylläpidon organisointimallin tavoitteena on kohdentaa ylläpidon resursseja niihin asiakastietojärjestelmän osioihin, joita muuttamalla voidaan saavuttaa eniten hyötyjä suhteessa käytettyihin resursseihin. Ylläpidon tehokkaalla organisoinnilla pystytään kasvattamaan asiakastietojärjestelmän käytettävyyttä sekä elinikää.

2.3 Lähestymistapa

Opinnäytetyö tehtiin tutkimuksellisenä kehittämistyönä. Tutkimuksellinen kehittäminen on tutkimukseen perustuvaa kehittämistä, joka on aina tarvelähtöistä. Tarve voi tulla esiin esimerkiksi organisaation muutostarpeista. Tutkimuksellisessa kehittämisessä pyritään luomaan uutta tietoa työelämän käytännöistä, uudistamaan käytäntöjä sekä ratkaisemaan ongelmia, jotka käytännöistä ovat nousseet. Kehittämisen tueksi käytännöstä sekä teoriasta kerätään systemaattisesti sekä kriittisesti arvioimalla tietoa. (Ojasalo ym. 2014, 18.) Tutkimuksellisen kehittämisen tutkimuksellisuus ilmenee niin, että kehittämisen valinnat perustellaan ja dokumentoidaan, haettu tieto on tutkittua ja käytännön tietoa, kehittämisessä luodaan analyttisesti erilaisia näkökulmia, prosessi ja lopputulos dokumentoidaan tarkasti sekä tietoa, näkökulmia, omia valintoja ja tuloksia arvioidaan kriittisesti (Ojasalo ym. 2014, 22).

Tutkimusstrategiasta, eli tutkimuksen menetelmällisten ratkaisujen kokonaisuudesta, voidaan suppeampana käsitteenä erottaa tutkimusmetodi. Sekä yksittäisen tutkimusmetodin että koko tutkimusstrategian valinta on aina riippuvainen tutkimustehtävästä. Samoin tutkimuksen lopullinen muoto on riippuvainen valitusta tutkimusstrategiasta. (Hirsjärvi ym. 2007, 121–128.)

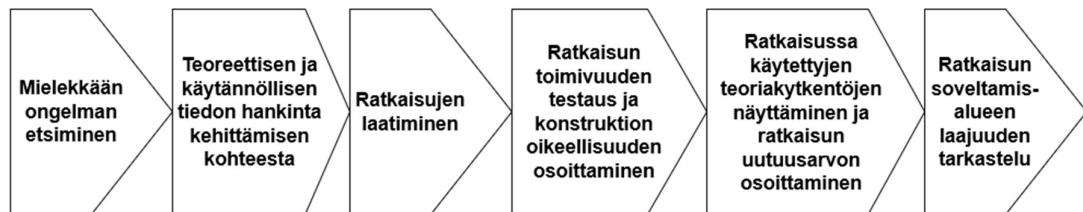
Tieteellisessä tutkimuksessa tehtävää tutkimusstrategian valintaa vastaa tutkimuksellisessa kehittämistyössä tehtävä lähestymistavan valinta. Se, minkä lähestymistavan mukaisesti kehittämistyötä suunnitellaan ei tarkoita vielä konkreettisten

menetelmien valintaa, vaan lähestymistapa valitaan ennen kehittämistyössä käytettäviä menetelmiä. Valittu lähestymistapa voi ohjata myös menetelmien valintaa, vaikka lähes kaikki menetelmät soveltuvat kaikkiin lähestymistapoihin. Kehittämistyöhön ei ole myöskään tarpeellista valita tarkkarajaisesti vain yhtä lähestymistapaa, vaan lähestymistavan valinnassa voi poimia useammasta lähestymistavasta parhaiten omaan kehittämistyöhön sopivat piirteet. Eri lähestymistapoihin liittyvät valinnat täytyy perustella ja kuvata selkeästi. (Ojasalo ym. 2014, 51-52.)

Konstruktiivinen tutkimus sopii tutkimuksellisen kehittämistyön lähestymistavaksi, kun tehtävänä on luoda jokin konkreettinen tuotos, esimerkiksi uusi tuote, järjestelmä, suunnitelma tai malli. Konstruktiivisessa tutkimuksessa pyritään ratkaisemaan ongelmia käytännönläheisesti luomalla uusi rakenne, jonka luomiseen käytetään olemassa olevaa teoreettista sekä käytännöstä kerättävää empiiristä tietoa. Tavoitteena on ratkaista käytännön ongelmia uudella ratkaisulla, jotka ovat teoreettisesti perusteltuja. Ratkaisujen tulisi tuottaa liiketoimintaan ja myös tiedeyhteisöön uutta tietoa, mutta oleellista on sitoa käytännön ongelma ja ratkaisu teoriaan. Konstruktion toimivuuden ja tieteellisen annin osoittaminen vaatii usein lisätyötä ja kehittämistöissä, kuten opinnäytetöissä, joudutaan arvioimaan, miten rakenteen toimivuus näytetään toteen. (Ojasalo ym. 2014, 65.)

Konstruktiivinen tutkimus on suunnittelua, mallintamista, toteutusta ja testaamista. Sen tavoitteena on pyrkimys muuttaa organisaation toimintaa ja käytänteitä tuottamalla merkityksellinen ja käytännössä hyödynnettävä rakenne, jota rakennettaessa tukeutuaan vahvasti aiempaan teoriaan. Konstruktiivinen tutkimus soveltuu ongelmanratkaisuun, johon tarvitaan ehdottomasti myös teoreettista tietämistä. Kehitystyön tuloksena luotuja rakenteita arvioidaan niiden käytännön hyödyn perusteella. (Ojasalo ym. 2014, 65-66.)

Konstruktiivisen tutkimuksen prosessi on kuvattu kuviossa 1.



Kuvio 1. Konstruktiivisen tutkimuksen prosessi (mukaillen Ojasalo ym. 2014, 67)

Kehittämistyön lähestymistavaksi valittiin konstruktiivinen tutkimus, koska tavoitteena oli luoda uusi malli ja tapa toimia. Kehittämistyönä oli luoda uusi prosessikuvaus, mitä kohdeorganisaatiolla ei aiemmin ollut käytössään, koska kohdetietojärjestelmä ei vielä

ollut ylläpitovaiheessa. Uuden ratkaisun paremmuutta aiempaan ratkaisuun verrattuna ei siis voitu todentaa, vaikka se on konstruktivisen tutkimuksen yhtenä tavoitteena. Konstruktivinen tutkimus soveltui lähestymistavaksi myös siksi, että opinnäytetyön tekijän arvion mukaan uuden prosessikuvauksen tulisi perustua ensisijaisesti teoreettiseen tietämykseen ja sen lisäksi empiiriseen tietoon. Tutkimusote on siis sekä deduktiivinen että induktiivinen. Teoreettinen tietämys on merkittävä perusta prosessikuvaukselle, jotta tietojärjestelmän ylläpito alkaisi mahdollisimman tehokkaasti, koska kohdeorganisaation kehittävän ylläpidon henkilöstöllä ei ole merkittävää kokemusta ylläpitotehtävistä. Empiirinen tieto tuo kehittämistyöhön arvokasta tietoa juuri kyseisen asiakastietojärjestelmän käyttäjistä ja käyttäjakeskeisyyden toteuttamisesta. Empiirisen tiedon avulla voidaan luoda kuvaus käyttäjakeskeisyyden nykytilasta ja kehittää toimintaa siinä havaittuihin haasteisiin peilaten.

Järvisen ja Järvisen innovaation toteuttamisprosessi koostuu lähtötilasta, toteuttamisesta ja tavoitetilasta. Heidän mukaansa motivaatio uuden innovaation kehittämiseen syntyy innovaation puutteesta tai vanhan innovaation huonoista tuloksista, eli konstruointi alkaa lähtötilasta, joka on erilainen kuin toivottu lopputila. Lopputila on kuvaus tilanteesta, johon toivomme pääsevämme toteuttamalla ideamme. Toteuttaminen tarkoittaa metodia, jonka avulla uskomme pääsevämme lähtötilasta tavoitetilaan. (Järvinen & Järvinen 2011, 108.)

Järvinen ja Järvinen esittävät kaksi erilaista tapaa pyrkiä lähtötilasta tavoitetilaan. Ensimmäinen on vaihejako, jossa edetään lineaarisesti vaiheittain lähtötilasta lopputilaan. Vaiheiden määrä voi vaihdella tilanteesta riippuen, mutta erillisinä vaiheina on ainakin spesifioitu tavoitetila sekä implementointi. Toisessa tavassa tavoitetilan määrittely ja sen toteutus tapahtuvat rinnakkain eli tuotetaan tavoitetilan prototyyppi ja verrataan sitä ajateltuun tavoitetilaan pyrkien koko ajan pääsemään lähemmäs tavoitetilaa. (Järvinen & Järvinen 2011, 108–109.) Molempien tapojen vahvuuksia voidaan hyödyntää eri tavoin eri tilanteissa ja sen vuoksi opinnäytetyön kehittämistyöhön sovelletaan molempia tapoja.

2.4 Tutkimusmenetelmät

Tutkimusmenetelmien valintaa ohjaa olennaisesti se, millaista tietoa tarvitaan ja miten tietoa aiotaan käyttää. Kehittämistyössä on keskeistä saada kehittämistyön tueksi erilaista tietoa ja eri näkökulmia, joten on suositeltavaa käyttää useampaa menetelmää, joilla saatava tietoa täydentää toisiaan ja lisää kehittämistyön päätöksenteon varmuutta. Asiantuntijatyön kehittämisessä on tyypillistä käyttää yhteisöllisiä menetelmiä, koska toimivat käytännön ratkaisut ovat usein tiiviin yhteistyön ja yhdessä työskentelyn lopputulos. (Ojasalo ym. 2014, 40.)

Opinnäytetyön kehittämistyö koskee asiakastietojärjestelmän kehittävän ylläpidon organisointia ja sen lähtökohtana on käyttäjäkeskeisyyden lisääminen. Jotta asiakastietojärjestelmän kehittävä ylläpito voidaan organisoida mahdollisimman tehokkaasti ja käyttäjäkeskeisesti, sen suunnittelemiseksi ja sitä kuvaavan prosessimallin toteuttamiseksi tarvitaan tietoa asiakastietojärjestelmän käyttäjistä sekä keinoista, joilla voidaan hankkia tietoa käyttäjistä ja järjestelmän käytöstä asiakastietojärjestelmän ylläpitovaiheessa. Tarvittava tieto koostuu käytettävyydestä, käyttäjäkeskeisestä kehittämisestä, miten käyttäjäkeskeisyys on toteutunut kyseisen asiakastietojärjestelmän ylläpitoa edeltävissä vaiheissa, millaisia hyötyjä sillä on saavutettu, millaisia haasteita siinä on ollut sekä miten käyttäjien näkökulmasta käyttäjien tulisi osallistua asiakastietojärjestelmän kehittävään ylläpitoon.

Konstruktiivisessa tutkimuksessa pyritään luomaan ratkaisu, jonka luomiseen käytetään teoreettista sekä empiiristä tietoa. Tutkimuksen teoriaosa syventää tutkimuksen näkökulmaa ja se johdetaan tutkimusongelmasta valitsemalla ne teoriat, jotka läheisimmin liittyvät ilmiöön (Kananen 2015, 112). Tutkimuksellisen kehittämistyön suunnittelun ja toteutuksen tukena käytetään tietoperustaa, joka koostuu olemassa olevasta tutkimustiedosta (Ojasalo ym. 2014, 34). Tämän opinnäytetyön tietoperusta koostuu käytettävyydestä sekä käyttäjäkeskeisestä kehittämisestä ja käyttäjien osallistumisesta tietojärjestelmän kehittämiseen sen eri vaiheissa. Erityistä huomioita tietoperustassa on kiinnitetty käyttäjäkeskeisyyteen tietojärjestelmän ylläpitovaiheessa ja tietojärjestelmän ylläpidettävyystekijöihin. Empiirinen tieto ratkaisun luomiseen hankitaan haastatteleamalla tietojärjestelmän käyttäjiä ja kehittämiseen osallistuneita henkilöitä.

Rikosseuraamuslaitoksen asiakastietojärjestelmän kehittämiseen on osallistunut joukko nykyisten ja tulevan asiakastietojärjestelmän käyttäjiä sekä käytettävyydsiantuntijoita, joilla kaikilla voidaan olettaa olevan tietoa asiakastietojärjestelmän tulevista käyttäjistä yleisellä sekä organisaatiokohtaisella tasolla sekä ideoita siihen, millä tavoin on tehokasta hankkia tietoa asiakastietojärjestelmän käytöstä. Heidän kokemuksensa ja tietonsa hyödyntämiseksi kehittämistyössä tarvittavan tiedon hankkimiseksi päätettiin järjestää työpaja, jonka tavoitteena oli ideoida keinoja, millä voitaisiin parhaiten hankkia tietoa tietojärjestelmän käytön haasteista ylläpitovaiheessa. Ideoiminen ja ideoiden esittäminen ei kuluta merkittävästi resursseja, mutta niiden toteuttaminen kuluttaa. Ideoinnissa määrä tuottaa usein myös laatua ja siksi ideoinnin tavoitteena onkin tuoda esiin mahdollisimman paljon ideoita, joiden joukosta voidaan löytää toteuttamiskelpoisimmat ja kustannustehokkaimmat ideat. (Ojasalo ym. 2014, 160.) Työpajan ideoita arvioitiin kolmesta näkökulmasta; tiedonhankinnan toteuttamiseen kuluvat resurssit, tulosten hyödynnettävyys ja käyttäjän motivaatio. Nämä näkökulmat valittiin, jotta työpajassa

kerättyjen ideoiden tehokkuus pystyttäisiin myös arvioimaan ja käyttämään tehokkuusarvioita prosessikuvauksen laatimisen tukena. Käyttäjän motivaatioon haluttiin kiinnittää erityistä huomiota, koska käyttäjäkeskeisyyden toteuttaminen on tehokkaampaa, mikäli käyttäjät ovat motivoituneita osallistumaan kehittämiseen.

Haastattelu on yksi käytetyimmistä aineistonkeruumenetelmistä kehittämistyössä, koska se sopii moniin kehittämistehtäviin, kun halutaan saada nopeasti kerättyä syvällistäkin tietoa kehittämisen kohteesta. Haastatteluilla voidaan selventää asioita sekä kerätä syvällistä tietoa ja uusia näkökulmia kompleksisista aiheista. Haastattelun lisäksi kehittämistyössä kannattaa usein käyttää myös muita menetelmiä, sillä useimmiten ne tukevat hyvin toisiaan. (Ojasalo ym. 2014, 106.) Haastattelu sopii menetelmäksi silloin, kun halutaan korostaa ihmistä tutkimustilanteen subjektina ja merkityksiä luovana osapuolena. Haastattelun etuna voidaan nähdä myös se, että sillä voidaan syventää saatavia tietoja, esimerkiksi perusteluja. (Hirsjärvi ym. 2007, 200.) Haastattelu valikoitui opinnäytetyön aineistonkeruumenetelmäksi, koska asiakastietojärjestelmän suunnittelijoilta ja käyttäjiltä oli tarve hankkia syvällisempää tietoa asiakastietojärjestelmän käyttäjäkeskeisen kehittämisen sekä asiakastietojärjestelmän käyttämisen onnistumisista ja haasteista. Näin pystytään kuvaamaan Järvisen ja Järvisen innovaation toteuttamisprosessiin kuuluva lähtötila käyttäjäkeskeisyyden toteutumisesta. Haastatteluilla pyrittiin saamaan lisätietoa käyttäjäkeskeisen kehittämisen toteutumisesta ja haasteista ennen tietojärjestelmän käyttöönottoa, jotta aiempien kokemusten perusteella tietojärjestelmän käyttäjäkeskeisyyttä pystytään lisäämään. Lisäksi haastatteluilla haluttiin tuoda esiin käyttäjien näkökulmaa käyttäjäkeskeisyyteen ja kehittämiseen osallistumiseen ylläpitovaiheessa.

Haastattelutyyppejä on erilaisia ja tyyppi tulee valita sen perusteella, millaista tietoa kehittämistyön tueksi tarvitaan. Isoimmat haastattelutyyppeiden erot liittyvät siihen, kuinka strukturoitu haastattelu on eli kuinka paljon haastattelun kulku voi vaihdella haastattelusta riippuen. (Ojasalo ym. 2014, 106.) Teemahaastattelu on haastattelun tyyppi, joka asettuu strukturointiasteikon keskivaiheille. Teemahaastattelun aihepiirit eli teemat, joita haastattelussa halutaan käsitellä, on asetettu haastattelulle etukäteen, mutta ei välttämättä tarkkoja kysymyksiä tai teemojen käsittelyjärjestystä. (Hirsjärvi ym. 2007, 203.)

Haastattelut päätettiin toteuttaa teemahaastatteluina sen joustavuuden vuoksi.

Teemahaastattelun etuna on, että siinä on jonkin verran tilaa haastateltavan puheelle ja on mahdollista käsitellä joitain teemoja tarkemmin tai väljemmin kuin muita, mutta kaikki tarvittavat teemat kuitenkin käsitellään. Valitsemalla teemahaastattelun haastattelun tyyppiksi haluttiin varmistaa, että haastateltavien itse esiin tuomat asiat voitaisiin varmasti

käsitellä. Koska teemahaastattelun kysymyksiä voi esittää eri sanamuodoin tai eri järjestyksessä joustavasti haastateltavasta riippuen, voidaan teemahaastattelun käytöllä mahdollistaa haastattelun sijasta keskustelunomainen ilmapiiri, jossa haastateltavan voi olla helpompaa kertoa näkemyksiään. Haastattelujen teemoiksi valikoitiin teorian ja tutkimustehtävän perusteella 1. Käyttäjakeskeisyys asiakastietojärjestelmän kehittämisen aiemmissa vaiheissa 2. Käyttäjien osallistuminen tietojärjestelmän kehittämiseen ylläpitovaiheessa 3. Käyttäjakeskeisen kehittämisen haasteet 4. Asiakastietojärjestelmän käytössä koetut haasteet.

Haastateltavien suostumuksella haastattelut tallennettiin, koska haastattelujen tallentaminen mahdollistaa haastattelijan keskittymisen täysipainoisesti haastatteluun ja haastateltavaan eikä hänen keskittymisensä kohdistu esimerkiksi muistiinpanojen tekemiseen. Tallentamalla haastattelut haastatteliija pystyy myös myöhemmin palaamaan haastatteluihin ja tekemään tarkempia havaintoja sekä löytämään uusia näkökulmia haastateltavan puheesta. Haastateltava ei aina ilmaise asioita suoraan, vaan se, mitä haastateltava on todella tarkoittanut, voi paljastua vasta haastattelun tallennetta kuuntelemalla. Haastattelujen tallentaminen mahdollistaa myös niiden puhtaaksi kirjoittamisen eli litteroinnin. (Ojasalo ym. 2014, 107.)

2.5 Aineiston analyysi

Haastattelujen jälkeen niistä kerätty aineisto käsitellään analysointia varten. Aineistosta voidaan tehdä päätelmiä useimmiten vasta aineiston esitöiden jälkeen. Aineiston analysointi on helpompaa, kun suullisesti käydyt haastattelut on kirjoitettu puhtaaksi eli litteroitu. Aineiston käsittely eli esimerkiksi litterointi on hyödyllistä tehdä melko pian aineiston keräyksen jälkeen. Aineistoa voidaan kerätä ja analysoida myös osittain samanaikaisesti. Ennen litteroinnin aloittamista on tiedettävä, millaista analyysia aineistosta aiotaan tehdä. Analyysin valinta voi ohjata esimerkiksi litteroinnin tarkkuutta, johon ei ole tarkkaa ohjetta. (Hirsjärvi ym. 2007, 216-218.) Litteroinnin tarkkuuteen vaikuttaa aineiston käyttötapana analyysivaiheessa ja raportoinnissa. Jos haastateltavan sanavalinnoilla on merkitystä, tulee litterointi tehdä sanatarkasti. Jos taas haastattelijan kiinnostuksen kohteena on enemmän esille tulleet asiat, voidaan litterointi tehdä karkeammalla tasolla. (Ojasalo ym. 2014, 110.)

Laadullisen aineiston analyysin tavoitteena on aineiston tiivistäminen ja jalostaminen käsitteelliseen tai teoreettiseen muotoon ja aineiston informaatioarvon lisääminen. Aineiston analyysin tekemiseen on useita eri menetelmiä ja yhtä yleispätevää mallia ei ole. Menetelmällä tarkoitetaan konkreettista tapaa, jolla aineisto käsitellään. Olennaista on se, mitä aineisto sisältää ja mistä se kertoo ja sen selvittämiseksi aineistoa on

tarkasteltava analyttisesti ja tulkittava tehtyjä löydöksiä teorian ja oman ajattelun avulla. Analyysi on monivaiheinen prosessi, joka yleensä koostuu aineistoon tutustumisesta kokonaiskuvan rakentamiseksi, aineiston teknisestä käsittelystä ja varsinaisesta analyysin tekemisestä eli aineiston jäsentelystä, erittelystä ja tulkinnasta. Tutkijan vastuulla on löytää aineistosta tutkimustehtävän kannalta olennaiset asiat. (Günther, Hasanen & Juhila 2021.) Analyysitapaa valitessa on tehokasta valita sellainen, joka parhaiten tuo vastauksen ongelmaan (Hirsjärvi ym. 2007, 219).

Aineiston analyysin tapaa valitessa on suositeltavaa tiedostaa, tehdäänkö analyysi aineiston vai aikaisemman teorian ohjaamana. Aineistolähtöisessä analyysissä oleellista on se, että analyysiyksiköt eivät ole etukäteen sovittuja, vaan pääpaino on aineistossa ja sieltä nousevista havainnoista eli analyysiyksiköt valitaan tehtävänasettelun ja tutkimuksen tarkoituksen mukaisesti aineistosta. Aikaisemmillä tiedoilla tai teorioilla ei tulisi olla vaikutusta aineistolähtöisen analyysin toteuttamiseen tai lopputulokseen. Täysin aineistolähtöinen analyysi on kuitenkin haastava toteuttaa, koska havaintojen teoriapitoisuus on myös yleisesti hyväksytty periaate, koska käytetyt käsitteet, menetelmät ja tutkimusasetelma ovat tutkijan asettamia ja näin vaikuttavat aina tuloksiin. Analyysissä tulisi pystyä kontrolloimaan, mikä analyysissä tapahtuu aineiston ehdoilla ja mikä tutkijan ennakkoluulojen ohjaamana, mikä on käytännössä melko ongelmallista. (Tuomi & Sarajärvi 2018, 108-109.)

Teoriaohjaavalla analyysillä voidaan pyrkiä ratkaisemaan aineistolähtöisen analyysin haasteita. Teoriapohjaisessa analyysissä analyysi ei pohjaudu suoraan teoriaan, mutta teoria voi toimia analyysin apuna. Analyysiyksiköt valitaan myös teoriaohjaavassa analyysissä aineistosta, mutta aikaisempi tieto ohjaa valintojen tekemiseen. Aikaisemman tiedon vaikutus on tunnistettavissa, mutta sen merkitys on oikeastaan toimia uusien ajatusurien aukojana. Teoriaohjaavassa analyysissä tutkijan päättelyn logiikka on usein abduktiivinen, jolloin ajatteluprosessissa vaihtelevat valmiit mallit ja aineistolähtöisyys, joita tutkija pyrkii yhdistelemään. (Tuomi & Sarajärvi 2018, 109-110.)

Eräs perusanalyysimenetelmä on sisällönanalyysi, jota voidaan käyttää kaikissa laadullisen tutkimuksen perinteissä (Tuomi & Sarajärvi 2018, 103). Teemoittelua voidaan pitää yhtenä sisällönanalyysin muotona ja sen tarkoituksena on paikantaa aineistosta tutkimusongelman kannalta keskeiset teemat ja nostaa esiin tutkimustehtävän kannalta olennaiset asiakokonaisuudet (Juhila 2021). Teemoittelussa painotetaan sitä, mitä kustakin teemasta on sanottu (Tuomi & Sarajärvi 2018, 105). Teemojen käsittelyn yhteydessä tutkimusraportissa esitetään usein sitaatteja aineistosta, mikä omalta osaltaan tukee litteroinnin merkitystä. Sitaateilla pyritään havainnollistamaan teemoittelua. Teema-

haastattelua ja teemoittelua käytettäessä on muistettava, että haastatteluihin valitut teemat eivät ole sama asia kuin analyysin lopputulemana oleva teemoittelu. (Juhila 2021.)

Haastatteluilla kerätty aineisto päätettiin analysoida teemoittelemalla teoriaohjaavasti, koska se koettiin parhaaksi tavaksi yhdistää konstruktiviseen kehittämiseen molempien teoriaa ja empiriaa yhdistelevän luonteen vuoksi.

2.6 Kohdeorganisaatio

Opinnäytetyön kohteena on Rikosseuraamuslaitos ja sen uusi asiakastietojärjestelmä. Rikosseuraamuslaitos on vankeusrangaistusten ja yhdyskuntaseuraamusten täytäntöönpanoviranomainen. Rikosseuraamuslaitos perustettiin vuonna 2010, kun Vankeinhoitolaitos ja yhdyskuntaseuraamusten täytäntöönpanosta aiemmin vastannut Kriminaalihuoltolaitos yhdistyivät yhdeksi virastoksi. (Rikosseuraamuslaitos 2019.) Rikosseuraamuslaitoksessa on käytössä oma asiakastietojärjestelmä vankeusrangaistuksille ja oma asiakastietojärjestelmä yhdyskuntaseuraamuksille. Uusi asiakastietojärjestelmä yhdistää nykyiset asiakastietojärjestelmät ja vie näin eteenpäin vuonna 2010 tapahtunutta organisaatiouudistusta. Uuden asiakastietojärjestelmän tavoitteena on erilaisten asiakkuusprosessien ja toimintatapojen yhtenäistäminen sekä asiakkuuden kokonaishallinnan näkyväksi tuominen ja osin mahdollistaminen. (Rikosseuraamuslaitos 2020.) Uuden asiakastietojärjestelmän käyttöönotto on suunniteltu tapahtuvan vuonna 2022 ja sillä tulee olemaan noin 2500 loppukäyttäjää.

2.7 Käsitteet

Opinnäytetyöhön on valittu keskeiseksi käsitteeksi käyttäjäkeskeisyys. Käyttäjäkeskeisyydellä tässä opinnäytetyössä tarkoitetaan sitä, että tietojärjestelmän kehittämisen keskiössä on käyttäjä, käyttäjän näkemykset ja tarpeet. Monissa yhteyksissä käyttäjäkeskeisyyttä ja käyttäjälähtöisyyttä käytetään rinnakkain eikä niiden ero ole täysin selkeä. Tähän opinnäytetyöhön haluttiin käsitteeksi valita käyttäjäkeskeisyys sen vuoksi, että sillä korostetaan käyttäjän asettamista keskiöön ja kehittämisen keskittämistä käyttäjän tarpeisiin. Käyttäjäkeskeisyys käsitteenä mahdollistaa käyttäjälähtöisyyttä enemmän käyttäjän olevan osapuoli, jolta aktiivisesti pyydetään näkemyksiä ja korostetaan näkemyksien tärkeyttä, kun taas käyttäjälähtöisyys käsitteenä voi aiheuttaa odotusarvon siitä, että näkemyksien, tarpeiden ja kokemusten esiin tuominen on käyttäjistä lähtöisin eli käyttäjän vastuulla. Asiakastietojärjestelmää kehitetään ensisijaisesti käyttäjiä varten, joten käyttäjien tarpeiden selvittäminen tulee olla lähtöisin kehitysorganisaatiosta.

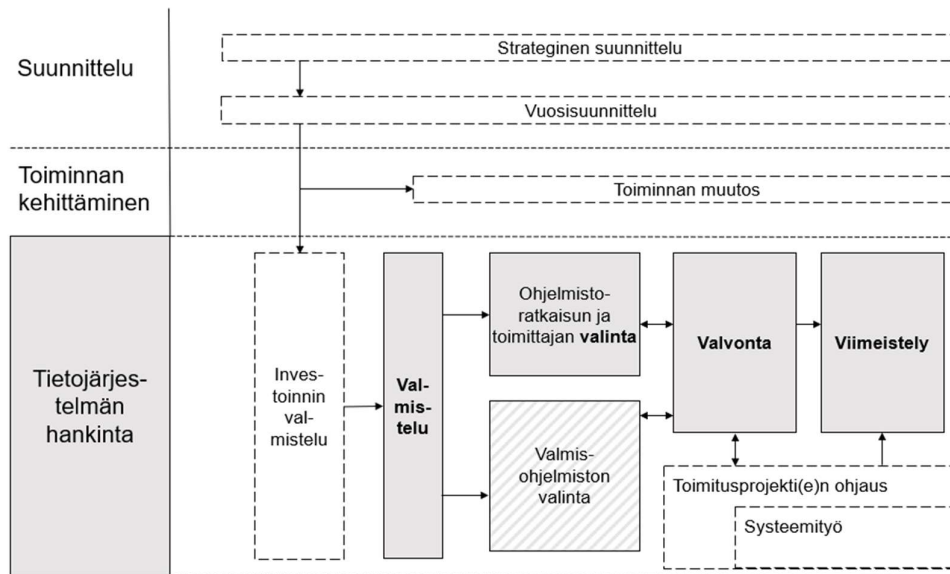
3 Tietojärjestelmän käytettävyys

Tietojärjestelmien kehittämisellä tavoitellaan aina hyötyjä organisaatiolle. Käytössä olevan tietojärjestelmän päivittäminen tai sen korvaaminen uudella tietojärjestelmällä voi olla teknologian elinkaaren vuoksi välttämätöntä, mutta myös silloin uudella tietojärjestelmällä halutaan saavuttaa myös muita hyötyjä. (Pohjonen 2002, 26.) On tehokasta kehittää samanaikaisesti sekä toimintaa että tietojärjestelmää tukemaan uutta toimintaa.

Tietojärjestelmän kehittäminen ilman toiminnan kehittämistä tai uuden tietojärjestelmän kehittäminen vanhojen toimintatapojen rinnalle ei tuota hyötyjä halutussa mittakaavassa. (Stenberg 2006, 103.) Tietojärjestelmällä saavutettavien hyötyjen edellytyksenä on uusien toimintamallien hyödyntäminen sekä tietojärjestelmän ja käyttäjän yhteensopivuus. Kehittyneet toimintaprosessit, laadukas tieto sekä käytettävyydeltään laadukkaat sovellukset ovat myös tietojärjestelmän hyödyntämisen keskeisiä tekijöitä. (Stenberg 2006, 56.)

Tietojärjestelmiä voidaan hyödyntää ainakin yksilö-, työkäytäne- ja tiimitasoilla ja jokaisella näillä tasoilla saavutettavat hyödyt hyödyttävät myös organisaatiota. Yksilötasolla käytettävän sovelluksen tulee edistää yksilön työtehtävien tekemistä. Tiimitasolla toimintaprosessi on yhteinen ja sovelluksen tulee palvella kaikkia tiimin osa-alueita myös ohjauksisältönsä puolesta luoden yhteisiä työkäytänteitä. Organisaatiotasolla hyötyjä voidaan saavuttaa myös eri sovellusten integroinnilla, mikä mahdollistaa tiedonsaannin kattavasti koko organisaatiosta. Tietojärjestelmän kehittämisellä tavoiteltavat tarkat hyödyt ovat organisaatiokohtaisia ja riippuvaisia tietojärjestelmän käyttötarkoituksesta. Hyötyjen saavuttaminen edellyttää selkeiden, organisaatiota palvelevien tavoitteiden asettamista. Tavoitteiden saavuttamiseksi tarvitaan koko organisaation sitoutumista ja osallistumista sekä uusien tietojärjestelmien ja toimintamallien kehittämiseen että niiden käyttöönottoon ja muutoksen läpivientiin. (Stenberg 2006, 56-57.)

Kuviossa 2 on havainnollistettu tietojärjestelmän hankinnan kokonaiskuvaa hankinnan ohjauksen 4V-mallin avulla. Malli koostuu valmistelusta, valinnasta, valvonnasta ja viimeistelystä. Valmistelun lähtökohtana tulee aina olla liiketoiminnallinen tarve, jonka avulla hankinnalle asetetaan tavoitteet. Tavoitteiden tulisi olla liitoksissa hankittavalla ratkaisulla tavoiteltavaan toiminnan muutokseen ja hyötyihin. (Forselius 2013, 9-11.)



Kuvio 2. Hankinnan kokonaiskuva (mukaillen Forselius 2013, 20)

Käyttäjystävälliset ja käytettävyydeltään korkeatasoiset tietojärjestelmät tuottavat monia hyötyjä; tietojärjestelmän käyttöönotto on sujuvaa, sen käyttökoulutukseen ei tarvitse varata isoja resursseja, käyttäjätuen tarve on vähäinen ja käyttävät tekevät vähemmän virheitä (Jokela 2010, 11). Käytettävyydeltään matalatasoisten tietojärjestelmien käyttö heikentää ihmisen työmoraaalia, vaikuttaa työpaikan ilmapiiriin ja aiheuttaa työtyytyväisyyden alenemista (Saariluoma ym. 2010, 20). Eri organisaatioissa arvotetaan eri asioita ja on tapauskohtaista, kuinka paljon edellä mainittuja etuja hyvällä käytettävyydellä saavutetaan. Vaikka käytettävyyden merkitys tiedostetaan, ei sitä useimmiten pystytä varmistamaan niin korkeatasoiseksi kuin olisi motivaatiota tai tarpeen. Useimmiten suunnitteluprojekteilla on aikataulu- ja resurssirajoitukset, joiden puitteissa ei ole mahdollista suorittaa niin paljon tai niin laajoja käytettävyyssaktiviteettejä kuin periaatteessa pitäisi (Jokela 2010, 60). Resurssien oikea ja tehokas kohdentaminen koskettaa myös käytettävyyttä. Projekteissa on tehtävä aina useita vaikeitakin päätöksiä siitä, mihin käytettävissä olevat resurssit käytetään ja silloin on myös päätettävä, kuinka paljon käytettävyyteen panostetaan ja kuinka paljon esimerkiksi loppukäyttäjää otetaan mukaan suunnitteluun. Johdolla on suuri vastuu siitä, että käytettävyyden tunnustetaan yhtenä tietojärjestelmän laatuksena, käyttäjakeskeisen suunnittelun tarve tunnustetaan ja että käytettävyyden varmistavalle työlle varataan riittävästi resursseja (Väänänen-Vainio-Mattila 2011, 109). Suunnittelussa tulisi pyrkiä siihen, että käytettävyyden tuottamiseen vaadittavat resurssit ovat järkevissä suhteissa sillä saavutettaviin hyötyihin (Jokela 2010, 11).

3.1 Käytettävyys

Standardin ISO 9241-11 mukaan käytettävyys on "Se vaikuttavuus, tehokkuus ja tyytyväisyys, jolla tietyt määritellyt käyttäjät saavuttavat määritellyt tavoitteet tietyssä ympäristössä". Käytettävyydellä on pitkä historia tuotekehityksessä ja nykyään sen voidaan ajatella olevan tuotteen keskeinen suunnittelutavoite ja laatuksiteeri. (Väänänen-Vainio-Mattila 2011, 102.) Standardi ISO 9241-11 määrittelee käytettävyyden olevan myös riippuvainen käyttötilanteesta. Jacob Nielsen määrittelee käytettävyyden olevan osa tuotteen käyttökelpoisuutta ja hyvän käytettävyyden koostuvan käyttötilanteen opittavuudesta, virheettömyydestä, muistettavuudesta, tehokkuudesta ja miellyttävyydestä. Käytettävyydestä käytetään monia määritelmiä, mutta sen voidaan katsoa olevan yhdistelmä erilaisia menetelmiä ja teorioita, joita hyväksi käyttämällä pyritään tehostamaan käyttäjän ja laitteen yhteistä toimintaa niin, että se on käyttäjän kannalta mahdollisimman miellyttävää. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 17.)

Jotta tietojärjestelmä käytettävyyttä voidaan arvioida, on sillä oltava käyttäjä eli useimmiten ihminen. Ihminen ei ole kone, jonka toiminta on tarkkaan määriteltyä ja ennustettavaa, mutta on silti joitain asioita, jotka pätevät lähes kaikkiin ihmisiin ja heidän toimintaansa. Jokaiselle säännölle on tietysti poikkeuksensa. (Sinkkonen ym. 2006, 21.) Tietojärjestelmän käyttö on ihmisen toimintaa, joten pystyäksään kehittämään tietojärjestelmän käyttöä on ymmärrettävä ihmisen toimintaa. Ihmisen eli käyttäjän toiminnan perustana ovat synnynnäiset fysiologiset ja psykologiset rakenteet, kuten aistit ja muistirakenteet sekä melko pysyviä kulttuurisia asioita, kuten kieli ja osa normeista ja tavoista. Käyttäjällä on myös erilaisiin toimintaympäristöihin liittyviä totuttuja tapoja, joita voidaan hyödyntää oppimisessa, jos tietojärjestelmien suunnittelijat noudattavat niitä. Nämä käyttäjän toimintaan liittyvät ominaisuudet ovat suhteellisen pysyviä. Niitä tai niistä johdettuja sääntöjä ei tarvitse tutkia jokaisen tietojärjestelmän kohdalla erikseen. Käyttäjän toimintaan vaikuttavat kuitenkin myös organisaation toimintatavat, tehtävät, yksilölliset toimintarajoitukset ja -kyvyt, käyttötilanne sekä käyttöympäristö. Nämä asiat riippuvat tilanteesta, jossa tietojärjestelmää käytetään ja tehtävistä, joihin se on tarkoitettu. Persoonasta ja tilanteesta riippuvat asiat ovat aina myös tietojärjestelmäkohtaisia ja ne on selvitettävä jokaisen tietojärjestelmän kohdalla erikseen, jotta niiden avulla pystytään kasvattamaan tietojärjestelmän käytettävyyttä. (Sinkkonen ym. 2006, 23–25.) Käytettävyystieto koostuu siis suhteellisen pysyvistä asioista sekä tietojärjestelmäkohtaisista asioista, jotka ovat riippuvaisia persoonasta ja tilanteesta. Tietojärjestelmän kehittämisen yhtenä haasteena on pystyä osoittamaan näiden tietojen

hankintaan ja käyttämiseen resurssit, joiden määrä on järkevässä suhteessa niillä saavutettuihin hyötyihin.

3.2 Saavutettavuus

Käytettävyyden lisäksi tietojärjestelmää kehitettäessä on huomioitava sen saavutettavuus. Käytettävyyden ja saavutettavuuden liittyvät kiinteästi toisiinsa ja ovat osittain päällekkäisiä käsitteitä. Korkealla käytettävyydellä varmistetaan se, että käyttäjät pystyvät käyttämään järjestelmää tai palvelua tehokkaasti ja miellyttävästi ja saavutettavuuden näkökulma kiinnittää huomion käyttäjien erilaisuuteen ja heidän erilaisten tarpeidensa huomiointiin sekä sisällön ymmärrettävyyteen. (Koskela 22.11.2017.) Saavutettavuuden tarkoituksena on tuottaa järjestelmiä, palveluja ja sisältöjä, jotka ovat saavutettavia kaikille käyttäjille heidän rajoituksensa tai haasteensa huomioiden (Etelä-Suomen aluehallintovirasto 2021).

Saavutettavuus koostuu teknisestä saavutettavuudesta, helppokäyttöisyydestä ja sisältöjen ymmärrettävyydestä ja selkeydestä. Tekninen saavutettavuus toteutuu, kun palvelua voidaan käyttää erilaisilla päätelaitteilla ja avustavilla teknologioilla, kuten ruudunlukuohjelmilla ja puheohjauksella. Lisäksi palvelun lähdekoodi on virheetöntä ja loogista. Helppokäyttöisessä palvelussa navigaatio on selkeä ja siinä on tarpeeksi rajattu määrä eri tasoja. Sivujen nimet ovat kuvaavia ja selkeitä. Helppokäyttöisyyden toteutuminen tarkoittaa myös sitä, että etsitty sivu löytyy vaivattomasti ja haluttu toiminto on helppo suorittaa. Keskeinen osa saavutettavuutta on myös ymmärrettävyys, mikä tarkoittaa selkokielen ja selkeän sekä ymmärrettävän kielen käyttöä. Kuvaavia väliotsikoita tulee olla riittävästi, tekstiin helppolukuista ja jäsennehtynä lyhyiksi kappaleiksi. Tekstin selkeys ulottuu myös linkkiteksteihin. (Etelä-Suomen aluehallintovirasto 2021.) Viranomaiset ovat velvoitettuja tekemään digitaalisista palveluista saavutettavia saavutettavuusdirektiivin ja lain digitaalisten palveluiden tarjoamisesta 306/2019 nojalla (Valtiovarainministeriö).

3.3 EU:n tietosuojasetus

Tietojärjestelmien kehittämistä ja tietojen hallintaa ohjaa vuodesta 2018 lähtien kaikissa EU-maissa sovellettu henkilötietojen käsittelyä sääntelevä EU:n tietosuojasetus. Tietosuojasetukseen viitataan usein lyhenteellä GDPR, mikä tulee sanoista general data protection regulation eli yleinen tietosuojasetus. Lainsäädännön uudistuksen tavoitteena on ollut vastata digitalisaation ja globalisaation aiheuttamiin uusiin tietosuojakysymyksiin, yhtenäistää EU-maiden tietosuojasääntelyä sekä parantaa henkilötietojen tietosuojaoikeuksia ja suojaamista. Rekisteröidyllä henkilöllä on oikeus tietää, mitä henkilötietoja organisaatiolla on hänestä sekä miten ja mitä tarkoitusta varten

henkilötietoja käsitellään. Henkilöllä on myös oikeus pyytää tietojensa poistamista sekä virheellisten henkilötietojen korjaamista. (Tietosuojavaltuutetun toimisto.)

Tietosuojajärjestelmä sisältää tietosuojaperiaatteita, jotka ohjaavat henkilötietojen keräämistä, käsittelyä, säilytystä ja päivitystä. Periaatteita on noudatettava kaikissa henkilötietojen käsittelyn vaiheissa. Henkilötietoja on kerättävä ja käsiteltävä nimenomaista ja laillista tarkoitusta vain ja niitä on kerättävä henkilötietojen käsittelyn tarkoitukseen nähden vain tarpeellinen määrä. Henkilötietoja on säilytettävä sellaisessa muodossa, josta rekisteröity on tunnistettavissa tietojenkäsittelyn tarkoituksen toteuttamisen kannalta riittävän kauan. Niitä on käsiteltävä lain- ja asianmukaisesti, luottamuksellisesti, turvallisesti ja rekisteröidien kannalta läpinäkyvästi. Epätarkat ja virheelliset tiedot on oikaistava tai poistettava viivytyksettä. Tietosuojajärjestelmien noudattaminen on pystyttävä osoittamaan. (Tietosuojavaltuutetun toimisto.)

3.4 Käyttökokemus

Käytettävyys, se, miten tuloksellisesti, tehokkaasti ja tyytyväisesti käyttäjät saavuttavat tavoitteet, on osa käyttökokemusta. Käyttökokemus on teknologian käytöstä koettu henkilökohtainen, vaihteleva ja yksilöllinen kokemus henkilön käyttäessä tietojärjestelmää. Kokemus ei ole yksinkertaisesti tunne teknologian ”hyvyydestä” tai ”huonoudesta”, vaan sen keskeinen osa on se, mikä merkitys teknologialla on käyttäjälle. Organisaation työn tekemiseen tarkoitettujen tietojärjestelmien merkitysvalikoima voi olla suppeampi kuin kulutustuotteen, mutta siinäkin voi olla eri sävyjä; työtavoitteiden saavuttamisen lisäksi sillä voi olla merkitys oman osaamisen julkittomisen välineen tai tehokkuuden osoittajana. Kokemukseen vaikuttavat monet tekijät, kuten mitä teknologian käyttämisellä tavoitellaan ja mitä odotuksia siihen kohdistuu. Myös aiemmat teknologiakokemukset vaikuttavat käyttökokemukseen. Vaikka tietojärjestelmä olisi kaikkien mittareiden mukaan käytettävä ja toiminnallisuuksiltaan hyvä, ei käyttäjä ota sitä omakseen, jos kokee sen esimerkiksi aiemman negatiivisen kokemuksen näkökulmasta vaikeakäyttöisenä tai hämmentävänä. Tunne siitä, että osaa ja pystyy hallitsemaan tietojärjestelmän käyttöä, on oleellinen osa positiivisen käyttökokemuksen muodostumisessa, mutta sitä hidastavana tekijänä on negatiivinen suhtautuminen teknologiaa kohtaan. (Saariluoma ym. 2010, 41–44.) Käytettävyys on mitattavissa oleva asia, mutta siihenkin vaikuttavat tilannesidonnaiset ja yksilölliset taustatekijät (Oulasvirta 2011, 31).

Tuote yksistään ei siis muodosta käyttökokemusta, vaan käyttökokemus on aina riippuvainen käyttötilanteesta. Ihmiset käyttävät tuotetta aina uniikissa tilanteessa, joka on suhteessa toisiin esineisiin ja ihmisiin, jossa he pyrkivät saamaan jotain aikaan ja johon vaikuttaa heidän aiemmat kokemuksensa. Käyttökokemus koostuu käyttäjän

persoonallisuudesta, fyysisestä ympäristöstä, käyttäjän toiminnoista ja tavoitteista, tuotteesta ja sen suhteesta muihin tuotteisiin sekä tuotteen merkityksestä, joihin liittyy myös aiemmat kokemukset ja tuotteen muokkaaminen osaksi omaa elämää. Käyttötilanteissa nämä käytön eri puolet ovat limittyneet ja osin myös muokkaavat toisiaan. Näiden eri elementtien vuorovaikutuksen ymmärtäminen on käyttöä koskevan tiedon ydinkysymyksiä. Käyttökokemuksen ymmärtäminen toimii myös lähtökohtana sen hahmottamiselle, mistä teknologian käyttö koostuu, mikä helpottaa käyttäjätiedon keräämistä. Ihmisen toimintaa ajaa eteenpäin pyrkimys saavuttaa jotain ja teknologia ja tuote on useimmiten toiminnan väline. Nämä välineet vaikuttavat myös siihen, miten ihminen toimintansa suuntaa. (Hyysalo 2009, 30-36.)

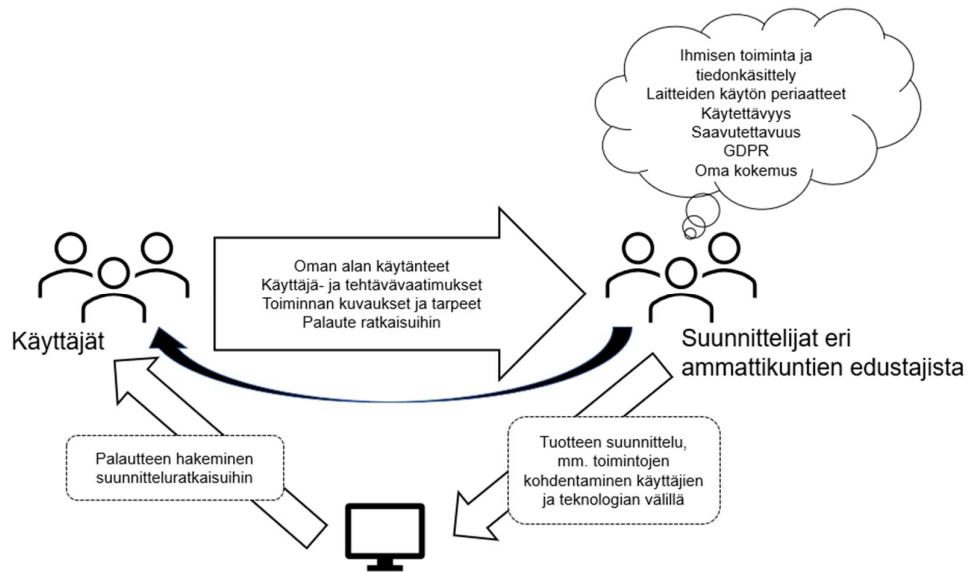
3.5 Käyttäjäkeskeinen suunnittelu

Käyttäjäkeskeinen suunnittelu on keskeinen lähestymistapa käytettävyyteen. Käyttäjäkeskeisessä suunnittelussa käyttäjä asetetaan suunnittelun keskiöön. Käyttäjäkeskeisen suunnittelun tavoitteena on tuottaa suunnitteluprosessiin käyttäjätietoa, joka mahdollistaa käyttäjien tarpeisiin vastaavan tuotteen suunnittelun ja sen myötä kehittää kyseisestä tuotteesta mahdollisimman hyvin käyttäjien tarpeita ja vaatimuksia vastaava. Sen vaiheita ovat käyttökontekstin määrittely, käyttäjien tarpeiden tunteminen ja niistä johdettavien vaatimuksien määrittely sekä aikaansaatuisten ratkaisujen toistuva arviointi käyttäjien kanssa heidän tarpeisiinsa peilaten. Tavoitteet voidaan saavuttaa ainoastaan, jos käyttäjät huomioidaan tietojärjestelmän kehittämisen kaikissa vaiheissa. Käyttäjäkeskeinen suunnittelu yhdistetään kyseisen tuotteen toteutusprosessiin. (Väänänen-Vainio-Mattila 2011, 102.)

Käyttäjätietoa kartutetaan useimmiten useista eri lähteistä, mutta harvoin käyttäjätietoa kerätään kaikista sen mahdollisista lähteistä (Hyysalo 2011, 152). Käyttäjien osallistuminen tuotteen suunnitteluun yleistyy koko ajan, mutta käyttäjätieto siirtyy ja sitä käytetään tuotekehityksessä myös muista lähteistä kuin suoraan kyseisen tuotteen käyttäjiltä kerättynä. Suunnittelijoiden ammattikäytännöillä ja niiden hyödyntämisellä on merkittävä vaikutus tuotteen kehitykseen. Suunnittelijoilla tulisi olla tarpeeksi tietoa ihmisen tiedonkäsittelyn lainalaisuuksista, yleisistä laitteiden käyttöä ohjaavista periaatteista sekä hyvän käytettävyyden periaatteista, jotta niitä voidaan hyödyntää tuotetta suunnitellessa. Merkittävää on myös kiinnittää huomioita sen jatkuvaan kehittämiseen, miten suunnittelijat ohjautuvat ottamaan käyttäjät huomioon. Suunnittelijat käyttävät työssään hyödyksi usein myös omaa kokemustaan kansalaisena ja käyttäjänä. Tämä perustuu ajatukseen, että suunnittelijan oma toiminta on yleisen käytön tai ainakin tuotteen käyttäjien käytön kanssa samankaltaista. Moni tuote on kuitenkin epäonnistunut

sen vuoksi, että suunnittelijoiden oletukset käyttäjien toiminnasta eivät olleet oikeita. Usein projekteissa käyttäjät perehdyttävätkin tuotekehittäjiä kyseisen oman alansa käytänteisiin ja periaatteisiin ja näin täydennetään jo muilla menetelmillä kartutettua käyttäjätietoa. Todellisen ja mahdollisimman tarkan käyttäjätiedon muodostamisella on siis olennainen merkitys projektin onnistumiselle. (Hyysalo 2011, 136-139.) Tuotteen kehittämisen ja käytön sovittaminen yhteen tuottaa monia hyötyjä ja yksi tapa kytkeä ne yhteen on käyttäjäkeskeinen suunnittelu (Hyysalo 2011, 148-149).

Käytettävyyssuunnittelun prosessia koskeva ISO13407-standardi määrittää käyttäjäkeskeiselle suunnittelulle neljä periaatetta: Käyttäjien aktiivinen osallistuminen sekä käyttäjä- ja tehtävävaatimusten selkeä ymmärtäminen, toimintojen kohdentaminen käyttäjien ja teknologian välillä, suunnitteluratkaisujen iterointi ja monialainen suunnittelu. Ensimmäisen periaatteen mukaisesti tuotteen kehittäjien tulee olla läheisessä vuorovaikutuksessa todellisten käyttäjien tai laajaa käyttäjäryhmää edustavan soveltuvan otoksen käyttäjistä kanssa. Sekä vaatimusten määrittelyssä että tarkemmassa suunnittelussa käyttäjien osallistuminen on tuottoisaa. Periaate toimintojen kohdentamisesta käyttäjien ja teknologian välillä pyrkii resurssien tehokkaaseen käyttöön niin, että käyttäjien ja teknologian työnjako suunnitellaan teknologian ja ihmisten kykyjen sekä tietojärjestelmän merkityksellisen käytön perusteella. Vaikka teknologialla pystyttäisiin toteuttamaan useita käyttäjän tekemiä tehtäviä, on joidenkin tehtävien jättäminen käyttäjän tehtäväksi järkevää esimerkiksi käyttäjien vahuuksien suhteessa teknologiaan vuoksi. Suunnitteluratkaisujen iterointi pyrkii varmistamaan sen, että tehtyihin suunnitteluratkaisuihin haetaan ja saadaan käyttäjiltä sekä organisaatiolta toistuvaa palautetta, jonka perusteella voidaan tehdä tarvittavia suunnittelumuutoksia. Iterointi pyritään ajoittamaan mahdollisimman aikaiseen vaiheeseen, jotta suunnittelussa pystytään varmistamaan suunnittelun oikeasta suunnasta. Monialaisen suunnittelun periaatteen mukaisesti käyttäjäkeskeiseen suunnitteluun tarvitaan eri osaamisalueiden ja ammattikuntien edustus, jotta suunnittelussa pystytään huomioimaan sekä käyttäjän vaatimukset että teknologian mahdollisuudet. Eri osaamisalueita ovat esimerkiksi käyttäjätutkimus, käyttöliittymäsuunnittelu ja tekniikka. (Väänänen-Vainio-Mattila 2011, 106–107.) Käyttäjäkeskeinen suunnittelu ei siis toteudu vain sillä, että osallistetaan loppukäyttäjät mukaan kehittämiseen vaan siihen tarvitaan riittävää osaamista myös eri ammattilaisilta. Käyttäjäkeskeinen suunnittelu ja sen vaikuttimet on kuvattu kuviossa 3.



Kuvio 3. Käyttäjäkeskeinen suunnittelu

4 Tietojärjestelmän käyttäjäkeskeinen kehittäminen

Käyttäjien tarpeiden huomiotta jättäminen on yksi suurimmista yksittäisistä syistä teknologiaprojektien epäonnistumiseen. Tuotteen käytön suunnittelun ongelmat johtuvat useimmiten siitä, ettei käyttöä suunnitellessa käytetä riittävän syvällistä ja laajaa käyttäjätietoa. Käyttäjätiedolla tarkoitetaan sitä, ketkä tuotetta käyttävät, mitä tarkoitusta varten ja missä käyttöyhteydessä ja ympäristössä. (Hyysalo 2011, 127-128.)

Käyttäjätietoa ei ole valmiina, vaan sen muodostuminen ja jalostaminen käyttökelpoiseksi tiedoksi vie kauan aikaa ja useita tuoteversioita. Käyttäjätieto on myös projektikohtaista ja uniikkia eikä siis sellaisenaan siirrettävissä tuotekehitykseen tai yhdestä projektista toiseen. Jokaisen tuotteen käyttäjätieto on erilaista. (Hyysalo 2011, 132-134.)

Käyttäjätietoa voidaan käyttää tuotekehityksen apuna, mutta sen käyttämisessä on ymmärrettävä, että käyttäjäisyys syntyy vasta itse käyttämistilanteessa. Käyttäjätiedon hyödyntämisessä on otettava huomioon, että teknologian konkreettinen käyttö voi tuoda esiin paljon uutta tietoa teknologian ominaisuuksista ja näihin ominaisuuksiin on reagoitava iteraatiokierroksilla. Tuotteen käyttämisellä tavoitellaan tuotteen käytölle erikseen määriteltyjä päämääriä eikä sen käytön tavoitteena ensisijaisesti ole käytön kohteen suunnittelu. (Hyysalo 2011, 133.)

4.1 Käyttäjä ja asiakas

Käsitteet käyttäjä ja asiakas sekoitetaan helposti. Käyttäjä on henkilö, joka käyttää tuotetta ja on konkreettisesti vuorovaikutuksessa sen kanssa. Asiakas on henkilö, joka tekee ostopäätöksen tuotteesta. Käyttäjä ja asiakas voivat olla myös sama henkilö esimerkiksi kuluttajatuotteiden kohdalla; Henkilö ostaa matkapuhelimen omaan käyttöönsä. Yritystuotteissa tyypillisesti käyttäjä ja asiakas ovat eri henkilöitä. Asiakas päättää yritysjärjestelmän ostamisesta organisaation käyttöön ja organisaation työntekijät käyttävät tuotetta. (Jokela 2010, 14–15.) Organisaation käyttöön kehitettävän tietojärjestelmän käyttäjät ovat useimmiten organisaation työntekijöitä, jotka tuntevat organisaation ja sen tehtävät ja ovat sen vuoksi yksi keskeinen informaation lähde järjestelmää suunniteltaessa (Pohjonen 2002, 47).

4.2 Käyttäjätyypit

Minkään tietojärjestelmän käyttäjät eivät ole samanlainen joukko ihmisiä, vaikka heitä kutsutaan yhteisellä nimittäjällä. Tietojärjestelmällä on erilainen rooli käyttäjälle ja heidän tapansa käyttää ja suhtautua tietojärjestelmään vaihtelee. Käyttäjien osallistumista suunniteltaessa kannattaakin ottaa huomioon heidän yleinen tietotekninen osaamisensa,

mikä voi vaikuttaa myös heidän osallistumiseensa. Osa käyttäjistä ei välttämättä ole erityisen kiinnostuneita tietotekniikasta tai voivat jopa pelätä sitä. Näille käyttäjille kynnyks osallistua tietojärjestelmän kehityshankkeisiin voi olla korkea, vaikka osallistuminen ei vaadi varsinaista tietoteknistä osaamista. Käyttäjät, jotka tuntevat tietotekniikkaa jonkin verran, voivat olla kehittämislle hyvin hyödyllisiä, koska heidän joukossaan on usein myös kiinnostusta osallistua kehittämiseen. Toisaalta tämän tason käyttäjistä voi olla myös haittaa kehittämislle, sillä vaikka he ymmärtävät tietotekniikkaa jonkin verran, eivät he välttämättä ymmärrä koko tietojärjestelmän kehittämisen prosessia ja siihen vaikuttavia seikkoja. Tietämyksensä vuoksi heillä voi olla motivaatio siirtyä käyttäjän roolista myös suunnittelun puolelle. (Pohjonen 2002, 48.)

Uuden innovaation omaksujat voidaan luokitella viiteen kategoriaan; innovoijat (innovators), varhaiset omaksujat (early adopters), varhainen enemmistö (early majority), myöhäinen enemmistö (late majority) ja hitaat omaksujat (laggards). Prosessiin tiedon leviämislle yksilölle vaikuttaa oleellisesti se, millaisena innovaation omaksujat kokevat uuden innovaation hyödyt suhteessa aiemmin käytössä olleeseen välineeseen, sen yhteensopivuuden aiempien kokemusten ja tarpeiden kanssa sekä sen monimutkaisuuden, kokeiltavuuden ja näkyvyyden. Omaksujien luokittelu perustuu henkilöiden asenteeseen uusia innovaatioita kohtaan ja heidän motivaatioonsa niiden omaksumiseen. Vain pieni osa (2-3%) väestöstä kuuluu innovoijiin, jotka ovat kiinnostuneita uusista innovaatioista ja ovat ensimmäisten joukossa ottamassa ne käyttöön. Noin 10-15% väestöstä kuuluu varhaisiin omaksujiin, jotka suhtautuvat uusiin innovaatioihin kiinnostuneesti, mutta ovat innovoijia valikoivampia ja harkitsevampia ottaessaan uusia innovaatioita käyttöön. Suurin osa ihmisistä (60-70%) kuuluu varhaiseen tai myöhäiseen enemmistöön ja he suhtautuvat uusiin innovaatioihin selvästi varauksellisemmin kuin edelliset omaksujat. Noin 10-20% ihmisistä kuuluu hitaisiin omaksujoihin, jotka vastustavat uusia innovaatioita. (Interaction design foundation.)

4.3 Käyttäjien osallistaminen tietojärjestelmän kehittämiseen

Tietojärjestelmien kehittämisen alkuaikoina keskityttiin lähinnä toteuttamaan sovelluksia, joiden tarkoitus ja tavoite oli helposti määritettävissä, esimerkiksi aiemmin määritellyn manuaalisen järjestelmän automatisointi. Silloin ajateltiin, että vain IT-ammattilaiset osallistuvat tietojärjestelmien suunnitteluun ja kehittämiseen ja käyttäjien rooli on käyttää heille tehtyä tietojärjestelmää. Nykyään tietojärjestelmät ovat paljon monimutkaisempia ja moniulotteisimpia kokonaisuuksia, joiden ajatellaan ratkaisevan käyttäjien ongelmia, joten käyttäjien osallistuminen tietojärjestelmän suunnitteluun nähdään tietojärjestelmäprojektin onnistumisen edellytyksenä. (Pohjonen 2002, 46.)

Käyttäjien osallistumista tietojärjestelmien suunnitteluun pidetään tärkeänä myös siksi, että se on oleellinen osa tietojärjestelmän vuorovaikutussuunnittelua ja hyvän käytettävyyden saavuttamista. Käytettävyydeltään korkeatasoinen tietojärjestelmä voi vähentää käyttäjätuen tarvetta, vähentää tehtyjen virheiden määrää, tehostaa työprosesseja, parantaa työn laatua ja nostaa työntekijöiden työhyvinvointia. (Jokela 2010, 11.) Tekniikka ilman sen käyttötarkoitusta ei ole käytettävää tai epätarkoituksenmukaista, vaan teknisiä ratkaisuja voidaan arvioida suhteessa siihen, miten tehokkaasti sitä käyttämällä käyttäjä saavuttaa tavoitteen, jonka saavuttamista varten hän teknologiaa käyttää. Tekniikan suunnittelijalla on oltava suunnittelunsa pohjana tiedossa päämäärät, jotka teknologialla pyritään saavuttamaan ja ongelmat, joita teknologialla ratkaistaan. Ihmisen toiminnan ymmärtäminen on olennaisessa osassa teknologian kehittämistä. (Saariluoma ym. 2010, 237–238.)

Hyvän käytettävyyden saavuttamiseen ja arviointiin on monia menetelmiä. Käytännössä hyvän käytettävyyden saavuttamisen kulmakivi on tuotteen loppukäyttäjien ottaminen mukaan sen suunnitteluprosessiin. Loppukäyttäjät tuovat suunnitteluun mukaan tietoa tuotteen todellisesta käyttöympäristöstä ja minkä tavoitteiden saavuttamiseen tai työtehtävien tekemiseen tuotetta käytetään. Loppukäyttäjät asettavat tuotteen laajempaan kontekstiin. (Jokela 2010, 62.) Mitä aiemmin myös loppukäyttäjät otetaan mukaan suunnitteluprosessiin, sitä paremmin heidän näkemyksensä voidaan ottaa huomioon käytettävyyttä suunnitellessa.

Käyttäjien huomiointi ja suunnitteluun mukaan ottaminen edesauttaa myös organisaation inhimillisen tietojenkäsittelyn huomioimista uudessa tietojärjestelmässä. Inhimillinen tietojenkäsittely tarkoittaa pääasiassa ihmisen toimesta tehtäviä toimintaprosesseja ja tehtäviä sekä niiden edellyttämiä tietotarpeita. Käyttäjillä on usein paljon kokemuksen myötä kertynyttä hiljaista tietoa, joka pitäisi pyrkiä selvittämään ja kuvaamaan tietojärjestelmän kehittämisen ensimmäisissä vaiheissa. Hiljaisen tiedon siirtäminen tietojärjestelmän osaksi on haastavaa, mutta ensiarvoisen tärkeää. Uutta tietojärjestelmää kehitettäessä on tärkeää selvittää myös organisaation toimintatapoihin liittyvät kehittämistarpeet, jotta uusi tietojärjestelmä palvelisi organisaatio parhaalla mahdollisella tavalla. Mikäli tietojärjestelmän toimintamalli ja organisaation toimintamalli eroavat toisistaan, saatetaan toimintaa tehdä jopa aiemmalla toimintamallilla sekä uuden järjestelmän vaatimalla tavalla, mikä lisää organisaation työmäärää merkittävästi ja turhaan. (Stenberg 2006, 90.) Tietojärjestelmien kehittämisessä tulisikin huomioida inhimillinen tietojenkäsittely ja sen erilaiset muodot eri ryhmien ja tiimien sisällä. Yhden ryhmän sisällä samoissa tehtävissä työskentelevien organisaation jäsenten välille syntyy samanlaisia työkäytänteitä. Toisen ryhmän sisällä, esimerkiksi toisessa toimipaikassa,

samoissa työtehtävissä työskentelevien jäsenten välille voi syntyä kuitenkin erilaisia työkäytänteitä. Ryhmän työkäytänteisiin voi vaikuttaa myös se, millaisten sidosryhmien kanssa he työskentelevät. Tietojärjestelmän käytön suhteen näiden eri ryhmien käyttötarpeet ja tottumukset voivat olla erilaiset, vaikka heidän työtehtävänsä periaatteessa ovat samanlaiset. Näiden eri työkäytänteiden huomiotta jättäminen kehittämistyössä voi olla haitallista uuden tietojärjestelmän käyttöönotolle. (Stenberg 2006, 45–46.) Käyttäjien osallistamisessa tietojärjestelmän kehittämiseen onkin tärkeää selvittää käyttäjien tarpeita mahdollisimman laajasti ja erilaisin tavoin, jotta saadaan selville hiljaista tietoa ja eri ryhmäkohtaisia työkäytänteitä. Hiljaisen tiedon selvittämisen haasteena on se, että sitä ei useimmiten tunnisteta ennen kuin jokin hiljaiseen tietoon liittyvä tehtävä tai työprosessi konkreettisesti muuttuu.

4.4 Käyttäjakeskeisyys tietojärjestelmien kehittämisen eri menetelmissä

Tietojärjestelmän kehittämisen menetelmäksi on olemassa useita vaihtoehtoja. Yksikään menetelmä ei ole paras ratkaisu jokaiseen tilanteeseen vaan oikea valinta kehitysmenetelmäksi riippuu tietojärjestelmän yksilöllisistä piirteistä. Kehittämismenetelmää päätettäessä yksi huomioonotettava piirre on käyttäjien osallistaminen tietojärjestelmän kehittämiseen ja eri kehittämismenetelmät huomioivat käyttäjät eri tavoin. Useimmiten eri kehittämismenetelmiä käsitellessä mainitaan vesiputousmalli ja ketterät menetelmät. Kohdeprojektin kehittämistä on tehty scrum-mallia noudattamalla.

Vesiputousmalli kehitettiin 1960-luvulla sen ollen yksi ensimmäisistä tietojärjestelmien elinkaarimalleista (Pohjonen 2002, 40). Huolimatta sen saamasta kritiikistä se on edelleen laajasti sovellettu elinkaarimalli (Juvonen 2018, 16-17). Vesiputousmallista käytetään myös nimitystä vaihejakomalli, koska sen periaatteisiin kuuluu vahva vaiheistaminen. Vaihejakomallissa kehittäminen tapahtuu vaiheittain ja perusvaiheet ovat esitutkimus, määrittely, suunnittelu, toteutus ja käyttöönotto. (Stenberg 2006, 104.) Vaiheiden nimet, sisällöt ja määrä voi olla erilainen eri projekteissa niiden tilanteista ja tarpeista riippuen. Pääsääntöisesti ensin määritellään mitä, sen jälkeen suunnitellaan miten, sitten toteutetaan ja lopuksi testataan. Vaiheet seuraavat toisiaan niin, että seuraava vaihe voi alkaa, kun edellinen vaihe on päättynyt ja täysin valmis. Kehittämisen edetessä tulisi pyrkiä siihen, ettei valmiiksi arvioituun vaiheeseen tarvitsisi enää palata, koska muutoksenhallinta vaatii resursseja. Vaiheittaista etenemistä voi perustella sillä, että jokaisen vaiheen toteutus on suunnitelmallisempaa ja sen edetessä ei todennäköisemmin tule vastaan isoja yllätyksiä, kun aikaisemmissa vaiheissa on dokumentoitu tarpeeksi tietoa ja eri osapuolet ovat verifioineet tiedon oikeaksi ja riittäväksi. Esimerkiksi tekninen

suunnittelu on turvallisempaa tehdä, kun kokonaisuus on tiedossa. (Lehtimäki 2006, 150-151.)

Hyvästä suunnittelusta huolimatta usein toteutuksen edetessä havaitaan puutteita aiempien vaiheiden toteutuksissa ja niiden aiheuttamien muutosten vaikutukset ovat laaja-alaiset ja vaikuttavat herkästi koko projektin aikatauluun. Koska vesiputousmallin periaatteisiin kuuluu, että aiempien vaiheiden on oltava valmiita ennen viimeisimpien vaiheiden joukossa olevaa käyttöönottovaihetta, joudutaan tietojärjestelmän arvon tuottoa odottamaan pitkään. (Juvonen 2018, 103.)

2000-luvun alkupuolella saivat alkunsa ohjelmistokehitykseen sovellettavat ketterät menetelmät osittain vastauksena vesiputousmallin haasteisiin. Ketteristä menetelmistä kenties eniten käytetty menetelmä on scrum. Scrumin ja ketterien menetelmien tärkein elementti on iteratiivisuus eli ohjelmiston toteuttaminen pieni osio kerrallaan alusta loppuun. Ketterien menetelmien suosio perustuu nopean toimittamisen sykliin eli parhaimmillaan jokin ohjelmiston osa suunnitellaan, toteutetaan, testataan ja otetaan käyttöön muutaman viikon ajanjakson aikana. Näin tuote valmistuu ominaisuus kerrallaan. (Juvonen 2018, 17-20.)

Eri elinkaarimallit huomioivat käyttäjäkeskeisyyden eri tavoin. Vesiputousmallia on kritisoitu siitä, että elinkaaren keskivaiheilla tietojärjestelmän käyttäjiltä eli tulevalta käyttäjäorganisaatiolta katoaa näkyvyys tietojärjestelmään. Mikäli käyttäjät eivät ole mukana tietojärjestelmän teknisessä suunnittelussa toteutuksessa, voidaan tehdä helposti väärään suuntaan meneviä ratkaisuja. Pahimmassa tapauksessa käyttäjät otetaan mukaan jälleen hyväksymistestausvaiheessa ja silloin voi olla jo myöhäistä korjata esiin nousseita ongelmia kohtuullisilla resursseilla. (Lehtimäki 2006, 151.)

Ketterät menetelmät ja niissä tapahtuva iteroiva kehittäminen tarjoavat käyttäjille enemmän mahdollisuuksia osallistua kehittämiseen koko kehittämisprojektin aikana. Vaikka käyttäjät eivät osallistuisi toteuttamisvaiheeseen, se kestää kerrallaan niin lyhyen aikaa, ettei sen aikana ehditä toteuttaa tietojärjestelmää liiallisesti väärään suuntaan. Tietojärjestelmää toteutetaan lyhyissä ajanjaksoissa lisää ja jokaisen ajanjakson jälkeen käyttäjät pääsevät näkemään, kokeilemaan ja kommentoimaan toimivaa sovellusta. Valmiista ohjelmistosta on myös helpompaa antaa palautetta kuin suunnitelmista ja dokumenteista, joissa voi olla liian vähän konkretiaa palautteen antamiseksi. (Lehtimäki 2006, 152.)

Myös iteroivaan kehittämiseen liittyy riskejä. Kun tietojärjestelmää kehitetään pala kerrallaan, voi kokonaisuuden hahmottaminen olla vaikeaa. Tällöin suunnitelmien, jotka

perustuvat kokonaisuuksiin, esimerkiksi arkkitehtuurisuunnitelmien, tekeminen voi olla vaikeaa. (Lehtimäki 2006, 153.) Erityisesti jos kyseessä on laaja ja monimutkainen tietojärjestelmä, jossa on paljon liittyviä ja riippuvuuksia eri toiminnallisuuksien välillä, olisi hyödyllistä hahmottaa ne kokonaisuudessaan jo suunnittelun alkuvaiheessa. Monimutkaisen toiminnallisuuden toteuttaminen pala palalta kokeilemalla voi kuluttaa liikaa resursseja (Lehtimäki 2006, 165). Kun tietojärjestelmä on jo tuotantokäytössä, voi laajojen muutosten tekeminen siihen olla vaikeaa, ellei mahdollista. Usein yhden toiminnallisuuden määrittelyssä tulee esiin jotain sellaista, jonka takia aikaisempiin toiminnallisuuksiin täytyy tehdä muutoksia (Lehtimäki 2006, 153). Myös käyttäjäkeskeisyyden näkökulmasta on valittava paras mahdollinen menetelmä juuri tiettyyn projektiin ja uskallettava myös soveltaa valitun menetelmän periaatteita. Esimerkiksi joissain tilanteissa on mahdollista pitää vesiputousmalli lyhyenä tai toteuttaa sen aikana pilotointi, jonka avulla voidaan osallistaa käyttäjät ja saada heiltä käyttökokemuksia toteutuksen tueksi (Lehtimäki 2006, 153). Käyttäjän on helppoa antaa palautetta ja kertoa käyttökokemuksestaan, jos hän pääsee konkreettisesti käyttämään ohjelmistoa. Useiden eri versioiden tekeminen toimivana ohjelmistona voi kuitenkin kasvattaa kustannuksia tai viivästyttää projektia liikaa, joten tiimin on tehokasta ja joissain tilanteissa välttämätöntä kehitettävä muitakin tapoja kommunikoida ja ymmärtää tehtyjä ratkaisuja kuin itse tietojärjestelmä. Ilman toimivaa kommunikaatiota asiakasorganisaation osaamista ja toimittajan teknistä osaamista ei saada tehokkaasti yhdistettyä. (Lehtimäki 2006, 165.)

4.5 Käyttäjäkeskeisyys tietojärjestelmän kehittämisen eri vaiheissa

Ei ole olemassa yhtä tietojärjestelmän kehittämisen tapaa, joka sopisi kaikkiin tietojärjestelmiin tai tietojärjestelmäprojekteihin. Useimmiten tietojärjestelmän tai sen jonkin ominaisuuden kehittämisessä on kuitenkin tunnistettavissa esitutkimus, vaatimusmäärittely, määrittely, suunnittelu, toteutus, käyttöönotto ja ylläpito (Pohjonen 2002, 26). Valitusta menetelmästä riippuen vaiheet tai osa vaiheista voidaan toteuttaa kertaluonteisesti toisiaan seuraten tai iteratiivisesti useamman kerran. Hyvin tyypillistä on, että ainakin osa vaiheista toteutetaan limittäin tai rinnakkain ja että tietyn vaiheen toteuttamiseen tarvitaan muiden vaiheiden tuotoksia. Vaiheiden toteuttaminen on aina projektikohtaista. Tietojärjestelmän kehittämiseen kuuluu myös paljon toimintoja, jotka vaikuttavat ja on otettava huomioon kaikissa elinkaaren vaiheissa. (Pohjonen 2002, 39.)

Nykyajan tietojärjestelmissä on tarpeen huomioida muitakin kuin tietoteknisiä näkökulmia, koska ne ovat laajoja organisationaalisia kokonaisuuksia. Mikäli tietojärjestelmä ja sen muutos vaikuttaa radikaalisti työntekijöiden työn kuvaan ja työtehtäviin, on suositeltavaa

ottaa heidät kehitystyöhön mukaan. (Pohjonen 2002, 46.) Kun käyttäjät otetaan mukaan palvelun suunnitteluun mahdollisimman varhaisessa vaiheessa, palvelu vastaa käyttäjien tarpeisiin, on miellyttävä käyttää ja sen käytettävyysongelmat pystytään minimoimaan (Saariluoma ym. 2010, 40). Tietojärjestelmien kehittäminen on laaja prosessi, jonka hallittavuuden kannalta se on järkevä vaiheistaa, mutta eri vaiheiden tulee silti muodostaa yhtenäinen kokonaisuus. Heti tietojärjestelmän suunnittelun alkuvaiheissa tulisi suunnitella myös alustava suunnitelma ylläpidosta, jotta myöhemmin ei ilmaantuisi ongelmia vaatimushallinnan ja ylläpidettävyyden osalta (Stenberg 2006, 108). Mitä myöhemmin jokin asia, vaatimus, ominaisuus tai tietojärjestelmän erityispiirre havaitaan, sitä enemmän sen huomioimiseen kuluu resursseja. Kaikki tietojärjestelmän suunnittelun alkuvaiheissa tehty ja dokumentoitu työ edesauttaa tietojärjestelmän myöhemmän suunnittelun vaiheita. Maailma myös muuttuu koko ajan ja myös tietojärjestelmälle asetetut odotukset ja vaatimukset voivat muuttua sen kehittämisen aikana. Tietojärjestelmää suunnitellessa kannattaakin sen kaikissa vaiheissa muistaa myös muutoksen mahdollisuus ja tehdä kaikki suunnitelmat niin, että muutokset myöhemmin olisivat toteutettavissa mahdollisimman tehokkailla työmäärillä. (Juvonen 2018, 59.)

4.5.1 Esitutkimus

Kun tarve tietojärjestelmän päivittämiselle tai uuden kehittämiselle on esitetty, voidaan aloittaa esitutkimusvaihe. Esitutkimuksen tarkoituksena on selvittää, miksi tietojärjestelmä tulisi kehittää, onko järjestelmän toteuttaminen mahdollista ja mielekästä sekä edellytykset sille. Esitutkimuksen perusteella tehdään päätös, ryhdytäänkö tietojärjestelmää kehittämään eli sen perusteella voidaan päätyä myös tietojärjestelmän kehittämättä jättämiseen. Esitutkimuksessa ei vielä toteuteta mitään eikä tehdä teknisiä ratkaisuja, vaan selvitetään uuden tietojärjestelmän päätavoitteet, viiteryhvät sekä mahdolliset ratkaisuvaihtoehdot toteuttamiselle. (Pohjonen 2002, 27.) Esitutkimuksessa kerätään tietoa myös organisaation nykytilanteesta sekä organisaatio- ja käyttäjäkohtaisista muutostarpeista. Tieto voidaan kerätä esimerkiksi käyttäjiä ja sidosryhmiä haastatellen. (Stenberg 2006, 105.) Käytettävyyssuunnittelun painopiste on tietojärjestelmän kehittämisen alkuvaiheissa (Väänänen-Vainio-Mattila 2011, 119) ja mitä aiemmin käytettävyys huomioidaan tietojärjestelmän kehittämisessä, sitä suurempia hyötyjä sillä saavutetaan. Käytettävyyden aikainen huomiointi välttyään kalliilta viime hetken muutoksilta. (Jokela 2010, 12.) Käyttäjakeskeistä suunnittelua voidaan toteuttaa esitutkimusvaiheessa esimerkiksi seuraavilla käytettävyyssuunnittelu menetelmillä; Käyttäjien määrittely, käyttäjätarvetutkimus tai skenaariot (Väänänen-Vainio-Mattila 2011, 119).

Organisaatioissa käyttöön otettava uusi tietojärjestelmä ja sen myötä tapahtuvien toimintamallien muutosten onnistunut läpivienti edellyttää muutosjohtamista. Muutos voidaan toteuttaa onnistuneesti vain, jos siihen saadaan mukaan suurin osa tai koko henkilöstö. Jotta henkilöstö saadaan mukaan muutokseen, on heille perusteltava muutoksen tarve. Organisaation työntekijöiden on ymmärrettävä, mikä muuttuu ja ennen kaikkea, miksi muutos on tehtävä. (Alahuhta 2015, 46.) Esitutkimusvaiheessa tuotetaan usein paljon erilaista aineistoa, jota voidaan hyödyntää myöhemmin samassa projektissa tai muissa projekteissa (Pohjonen 2002, 27–28). Tuleva muutosten läpivienti on tehokkaampaa ja kokee vähemmän muutosvastarintaa, kun käyttäjät otetaan mukaan tietojärjestelmän kehittämiseen jo esitutkimusvaiheessa ja heidän näkemyksiään muutostarpeista on kuunneltu. Muutosten viestinnässä ja toteuttamisessa voidaan käyttää materiaalina esitutkimusvaiheessa tuotettua aineistoa.

4.5.2 Vaatimusmäärittely

Kun esitutkimuksen perusteella päätetään ryhtyä kehittämään (uutta) tietojärjestelmää, alkaa vaatimusmäärittelyvaihe. Vaiheessa tuotetaan vaatimusmäärittelydokumentti, jossa tulisi olla ainakin kuvaus kehittämishankkeen toimeksiannosta, yleiskuvaus organisaation nykytilasta kyseisen tietojärjestelmän osalta, tietojärjestelmälle asetetut tavoitteet sekä eri sidosryhmien asettamat toiminnalliset ja ei-toiminnalliset vaatimukset. Vaatimukset ovat eri sidosryhmien tarpeita, jotka ovat riippumattomia valitusta teknisestä toteutuksesta. Toiminnalliset vaatimukset koskevat sitä, mitä tietojärjestelmän odotetaan tekevän, miten eri sidosryhmät työskentelevät tietojärjestelmän kanssa ja miten tietojärjestelmä toimii ulkoapäin tarkasteluna. Ei-toiminnalliset vaatimukset määrittelevät sen, minkälaisilla tietojärjestelmän ominaisuuksilla se täyttää toiminnalliset vaatimukset. (Pohjonen 2002, 28.)

Vaatimusten keräämiseen käytetään usein monia eri menetelmiä, koska kerääminen on vaativaa työtä ja ei ole olemassa yhtä yksittäistä menetelmää, jonka käyttäminen automaattisesti takaisi vaatimusten riittävän määrän ja tason. Menetelmien valinta on myös riippuvainen organisaation ja uuden tietojärjestelmän piirteistä. Usein käytetty tapa vaatimusten keräämiseen on haastatella organisaation ja tietojärjestelmän sidosryhmien jäseniä, mm. käyttäjiä. Koska vaatimusmäärittelydokumentissa kuvataan myös organisaation nykytilaa, kannattaa haastatteluja hyödyntää myös hankittaessa tarpeellista tietoa tietojärjestelmän tulevista käyttäjistä ja heidän työtehtävistään ja vastuista, tietojärjestelmän ympäristöstä sekä organisaation rakenteista. (Pohjonen 2002, 28.)

Hyvän käytettävyyden perusteet luodaan viimeistään käyttäjien tarpeita ja vaatimuksia selvitettäessä. Käyttökontekstin analysointi, käyttäjien tarpeiden tunteminen sekä

vaatimusten määrittely ovat käyttäjakeskeisen suunnittelun keskeisiä osa-alueita. (Väänänen-Vainio-Mattila 2011, 102–103.) Käyttökonteksti on laaja kokonaisuus, joka sisältää fyysisen ympäristön, tehtäväkontekstin ja sosiaalisen, psykologisen ja teknisen kontekstin. Käyttökonteksti vaikuttaa tietojärjestelmän käytettävyyteen, joten käyttökontekstin ymmärtäminen ennen tietojärjestelmän suunnittelua on käytettävyyden kannalta oleellista. Erityisesti organisaation käyttöön tulevan tietojärjestelmän käytettävyyden kannalta on myös oleellista, että se suunnitellaan tietylle käyttäjäryhmälle. Jotta voi suunnitella organisaation työntekijöiden tarpeita vastaavan tietojärjestelmän, on tunnettava käyttäjien ja heidän organisaationsa piirteet ja tavoitteet. (Väänänen-Vainio-Mattila 2011, 108.)

Vaatimusten keräämiseen ja käsittelyyn kannattavaa varata riittävästi aikaa, jotta niihin saadaan koottua kaikki tarvittavat vaatimukset kaikille ymmärrettävässä, yksiselitteisessä muodossa. Kuten useimpien muutosten, myös puuttuvien vaatimusten lisääminen myöhemmin voi kuluttaa paljon resursseja. Myös vaatimusten väärä tulkinta aiheuttaa hidasteita projektille, joten niiden tulisi olla niin selkeitä, ettei tulkinnalle jää tilaa. Vaatimusten tulee olla sellaisia, että niiden toteutumisen voi yksiselitteisesti todentaa. Esimerkiksi vaatimus ”työnteon tehostumisesta” on liian yleisellä tasolla. Koska tietojärjestelmän kehittämisen rinnalla kehitetään useimmiten myös organisaation toimintaa, saatetaan vaatimukseen kirjata virheellisesti myös ratkaisuja organisaatiossa koettuihin ongelmiin. Hyvin harvoin tietojärjestelmällä kyetään kuitenkaan ratkaisemaan sellaisia haasteita, joihin organisaatiolla itsellään ei ole ratkaisuja. (Pohjonen 2002, 30–31.) Sen lisäksi, että vaatimukset määritellään tässä vaiheessa tarpeeksi tarkalle tasolle, on tärkeää, että niitä toteutettaessa niitä tarkennetaan yhdessä käyttäjien tai käyttäjien edustajan kanssa, jos se on tarpeellista. Mikäli keskusteluyhteys toteuttajien ja käyttäjien välillä katkeaa ja tietojärjestelmä toteutetaan vaatimusmäärittelyiden perusteella sellaisenaan, on vaarana, että mahdolliset muutospyynnöt ja tarkennukset esitetään liian myöhään. (Juvonen 2018, 88.)

Viimeistään vaatimusmäärittelyvaiheessa on kannattavaa asettaa myös tietojärjestelmän käytettävyystavoitteet. Käytettävyys ei ole asia, joka tietojärjestelmällä joko on tai ei ole, vaan se on jatkuvasti muuttuva suure. Sen vuoksi on arvioinnin kannalta tarkoituksenmukaisempaa asettaa määrätty taso, joka käytettävyydelle halutaan saavuttaa. Oikea taso on taas riippuvainen organisaatiosta sekä sen toiminnasta. Vaatimusmäärittelydokumentissa kuvattu organisaation nykytila, ympäristö ja rakenne ohjaavat tason määrittystä. Strategiset käytettävyystavoitteet kuvastavat organisaation strategisia valintoja sen suhteen, kuinka paljon resursseja voidaan osoittaa käytettävyyden kasvattamiseen. (Pohjonen 2002, 30–31.) Jakob Nielsen on määritellyt

käytettävyydelle seuraavat komponentit; Opittavuus, tehokkuus, muistettavuus, virheiden välttäminen ja niistä palautuminen sekä tyytyväisyys. Näistä komponenteista voidaan johtaa käytettävyystavoitteet ja mittarit. Mittarit voivat olla määrällisiä tai laadullisia ja niiden avulla käytettävyyttä voidaan systemaattisesti kehittää. Mittareita itsessään voidaan myös kehittää suunnittelun edetessä. Käytettävyystavoitteena voi olla esimerkiksi tiettyyn tehtävään käytetty aika, jonka attribuuttina on tehokkuus. (Väänänen-Vainio-Mattila 2011, 103–104.)

4.5.3 Määrittely

Vaatimusten määrittelyn jälkeen vaatimukset analysoidaan ja niistä johtamalla tehdään tietojärjestelmän toiminnallinen määrittely. Toiminnallisen määrittelyn tavoitteena on selvittää ja kuvata toteutusriippumattomasti, mitä uuden tietojärjestelmän tulee tehdä. Määrittelyssä tuotetaan kuvaus tietojärjestelmän toiminnoista, käsiteltävistä tiedoista, liittymistä tietojärjestelmän ulkopuolelle, tietojärjestelmän rajoitteista sekä eri viiteryhmistä. Määrittelyssä tuotettu dokumentaatio kuvaa tietojärjestelmän toiminnan yleisellä tasolla sekä jokaisen toiminnon yksityiskohtaisesti. (Pohjonen 2002, 31–32.) Määrittelyvaiheessa siis kuvataan koko tietojärjestelmä loogisesti ja määritellään myös sen koulutus ja käyttöönotto sekä hahmotellaan tuleva ylläpitomalli. Tässä vaiheessa on järkevää myös esiteltä koko tietojärjestelmän kokonaistoiminta, jotta mahdollisten epäloogisuuksien tai puutteiden eteneminen pidemmälle voidaan estää. (Stenberg 2006, 105.)

Tietojärjestelmän myöhempien vaiheiden toteutus pohjautuu pääosin määrittelyvaiheessa tuotettuihin kuvauksiin, joten ne kannattaa tehdä huolellisesti ja selkeästi. Jokainen valinta on järkevää perustella perusteellisesti ja huolehtia, että määrittelyissä selkeytynyt tahtotila tulee dokumenteista selkeästi esille. (Pohjonen 2002, 32.) Mikäli jostain syystä määrittelyissä ei ole kuvattu tarpeeksi tarkasti, miten tietojärjestelmän halutaan toimivan eikä toimittaja tarkenna tätä asiakkaan kanssa, vaan olettaa asiakkaan tarpeita, aiheuttaa se turhia ja kalliita ylläpitotöitä (Koistinen 2002, 202). Määrittelyprosessin tulee olla käyttäjäkeskeinen ja tietojärjestelmien suunnittelun ammattilaisten tarkoituksena on ohjata prosessia ja selkeyttää mahdollisuuksia (Stenberg 2006, 105).

Määrittelyvaiheessa organisaatiossa kauan työskennelleiden henkilöiden osaaminen ja ymmärrys tulisi muuntaa tietojärjestelmän toiminnaksi eli tulee kuvata, miten tietojärjestelmän pitäisi toimia. Usein se tuottaa vaikeuksia. Isojen linjojen ja perustyöprosessien kuvaaminen on usein suoraviivaista, mutta mitä tarkemmalle tasolle ja yksityiskohtiin määrittelyssä edetään, sen haastavampaa se on. Lähes kaikissa organisaatiossa ja sitä myöden tietojärjestelmissä on monia sääntöjä ja niihin poikkeuksia, joita ei aina osata edes perustella. Tämä kaikki tulisi osata aukottomasti määritellä

tietojärjestelmän tehtäväksi. (Lehtimäki 2006, 160-161.) Haasteita voivat aiheuttaa myös tilanteet, joissa tietojärjestelmän on määriteltävä toimimaan tilanteissa, joihin organisaatiossa ei ole vielä toimintaohjetta. Tällöin tietojärjestelmän toiminnallisuutta ennen on ratkaistava, miten tilanteessa tulisi toimia ja kuka päätöksen voi tehdä.

Määrittelytyö on haastavaa siksikin, että se tehdään projektin alkuvaiheessa, jolloin toteutustiimillä on vähiten ymmärrystä tietojärjestelmän taustoista ja toiminta-alueesta. Määrittelyvaiheessa tehty tutustuminen kontekstiin ja toimialueeseen vähentää virheiden riskiä myös toteutusvaiheessa. Kun kehittäjä ymmärtää jollain tasolla tekemänsä tietojärjestelmän käyttäjiä ja käyttäjien toiminta-alueita, pystyy hän tekemään parempia ratkaisuja itsenäisesti ja tulkita määrittelyjä oikein. Myös tästä syystä käyttäjien ja tulevan käyttäjäorganisaation asiantuntijoiden osallistuminen määrittelytyöhön on tärkeää ja hyödyllistä. Määrittelytyöhön osallistumisen lisäksi he osaavat kertoa taustoja ja perusteluja tehdyille määrittelyille ja näin lisätä toteutustiimin tietämystä tietojärjestelmän taustalla olevasta toiminnasta. Ymmärrys siitä, mitä on tekemässä antaa hyvät edellytykset onnistua määrittely- ja toteutustyössä. (Lehtimäki 2006, 161-163.)

Lähes jokaisen toiminta-alueen toiminta on yksityiskohdiltaan monimutkaista ja siksi myös määrittely on haastavaa. Toiminnan monimutkaisuus voi kuitenkin olla aitoa tai rakennettua. Aito monimutkaisuus perustuu ympäröivään maailmaan, toiminta-alueeseen, lakeihin, ohjeisiin ja toimintamalleihin, jotka yhdessä tekevät kokonaisuudesta monimutkaisen. Aito monimutkaisuus on käsiteltävä ja ratkaistava jotenkin, joko toteuttamalla se tietojärjestelmään tai tietyissä tilanteissa päättää jättää toiminnallisuus ihmisvoimin toteutettavaksi. Esimerkiksi kaikkein harvinaisempien tilanteiden kohdalla voi olla tehokkaampaa, ettei toimintaa automatisoida tietojärjestelmän tehtäväksi. Rakennettu monimutkaisuus on aiheutunut aiempien vuosien aikana ja mahdollisten aikaisempien tietojärjestelmien keinotekoisista rakenteista. Kun uutta tietojärjestelmää ryhdytään määrittelemään, tulee selvittää, mitkä tarpeet ovat toimintojen ja määrittelyjen taustalla ja pyrkiä yksinkertaistamaan niitä niin paljon kuin mahdollista. Aiempien tietojärjestelmien monimutkaisuutta tai vuosien aikana aiheutuneita toimintatapoja ei kannata monistaa uuteen tietojärjestelmään selvittämättä aitoa tarvetta niiden taustalla.

Asiakasorganisaation asiantuntijan kanssa on käytävä keskusteluja siitä, miksi jokin toiminnallisuus halutaan toteutettavan juuri määritellyllä tavalla. Keskusteluissa tulee kuitenkin välttää kyseenalaistamista tarvetta automaattisesti vaan pyrkiä ymmärtämään toimintaa ja peruseriaatteita sen taustalla. (Lehtimäki 2006, 163-164.)

4.5.4 Suunnittelu

Suunnitteluvaiheessa käyttäjien tarpeiden mukaisesti tehty toiminnallinen määrittely muunnetaan tietojärjestelmän tekniseksi määrittelyksi eli suunnitellaan, miten tietojärjestelmä toteutetaan (Pohjonen 2002, 33). Suunnittelussa tarkennetaan käyttäjien toimintakuvauksia vielä tarkemmalle tasolle ja suunnitellaan tietokantojen rakenne, sisältö ja sovellusten ohjelmistot. Tässä vaiheessa voidaan testata tietojärjestelmää kokonaisuutena sekä protoilla osatoimintoja ja saada käyttäjiltä palautetta niistä. Käyttäjien tulisi olla aktiivisesti mukana suunnitteluvaiheessa, vaikka tietojärjestelmien suunnittelun ammattilaisten rooli kasvaa. (Stenberg 2006, 105.)

Suunnitteluvaiheessa suunnitellaan myös tietojärjestelmän ohjelmistojen käyttöliittymät. Vaikka hyvä käytettävyys on paljon muutakin kuin toimiva käyttöliittymä, niin loppukäyttäjälle tietojärjestelmän käytettävyys näyttäytyy useimmiten käyttöliittymän kautta (Jokela 2010, 63). Käyttöliittymä koostuu visuaalisesta tasosta ja vuorovaikutustasosta ja näistä vuorovaikutustaso jakautuu arkkitehtuuriin ja vuorovaikutuselementtitasoon. Vuorovaikutustaso on periaatteessa järjestelmän logiikan ydin. Arkkitehtuuri määrittää käyttöliittymän perusrakenteen ja navigoinnin järjestelmässä ja vaikuttaa näin laajemmassa mittakaavassa siihen, miten käyttäjä ryhtyy ja pystyy tekemään työtehtäväkokonaisuuksia eli laajaan käytettävyyteen. (Jokela 2010, 17.) Arkkitehtuurissa tehdyillä valinnoilla on iso vaikutus laajaan käytettävyyteen ja sen toteutumiseen (Jokela 2010, 23). Vuorovaikutuselementtitasolla tarkoitetaan käyttöliittymän yksittäisten elementtien toteutusta eli millaisia termejä, valikoita, tietokenttiä, painikkeita ja niin edelleen tietojärjestelmässä käytetään. Tämä taso vaikuttaa suppeaan käytettävyyteen eli siihen, millä tavalla käyttäjä pystyy tekemään yksittäisiä isompiin kokonaisuuksiin kuuluvia osatehtäviä, esimerkiksi valitsemaan oikean arvon tietokenttään. (Jokela 2010, 17.)

Vuorovaikutuselementtitasoon on useimmiten helppo tehdä muutoksia, koska niiden vaikutukset muualle ovat pienet. Yksittäisen kentän muodon muutos ei aiheuta muutoksia muualle tietojärjestelmään. Arkkitehtuuri on niin olennainen osa tietojärjestelmää ja siihen tehdyt muutokset vaikuttavat laajasti, että muutokset arkkitehtuuriin vaativat usein paljon resursseja. (Jokela 2010, 18.) Jotta arkkitehtuuriratkaisuihin ei olisi tarvetta tehdä myöhemmin aikaa vieviä ja kalliita muutoksia, tulisi sen suunnitteluun käyttää tarpeeksi aikaa ja osallistaa myös käyttäjät antamaan palautetta suunnitteluratkaisuista. Mikäli käytettävyys huomioidaan vasta siinä vaiheessa, kun arkkitehtuuri ja käyttöliittymä on jo suunniteltu, ei välttämättä ole resurssien puitteissa mahdollista tehdä enää sellaisia muutoksia, jotka vaikuttavat läpi tietojärjestelmän ja joita kuitenkin vaadittaisiin hyvän

käytettävyyden saavuttamiseen. (Jokela 2010, 63.) Hyvän käytettävyyden saavuttaminen on sitä helpompaa ja tehokkaampaa, mitä aiemmin se otetaan tietojärjestelmän kehittämisessä huomioon.

Eryityisesti tietojärjestelmää suunnitellessa tulisi muistaa, että vaikka tietojärjestelmän kehittämisen kaikissa vaiheissa olisi kaikki toimenpiteet tehty oppikirjamaisesti niin hyvin kuin mahdollista, on muuttumassa maailmassa väistämätöntä, että tietojärjestelmän vaatimukset ja määrittelyt voivat muuttua ennen käyttöönottoa tai viimeistään ylläpitovaiheessa. Suunnittelun lähtökohtana kannattaakin pitää se, että tulevat muutokset pystytään toteuttamaan mahdollisimman helposti. Tähän voi olla ratkaisuna esimerkiksi suunnitella tietojärjestelmä koostumaan eri komponenteista, jolloin muutoksia pystytään kohdentamaan helpommin vain yhteen tietojärjestelmän osaan (Stenberg 2006, 103). Toisena ylläpidettävyystekijänä voidaan pitää ohjaustietojen parametroitua käyttäjän ylläpitämäksi (Stenberg 2006, 116). Myös tietojärjestelmässä olevan virheen korjaaminen on riskittömämpää, jos jo suunnitteluvaiheessa on pyritty siihen, että kukin toiminto on oma komponenttinsa. Näin yhden virheen korjaaminen aiheuttaa vähemmän todennäköisesti sivuvaikutuksena uusia virheitä. (Pohjonen 2002, 38.)

4.5.5 Toteutus ja testaus

Mitä huolellisemmin tietojärjestelmän aikaisemmat vaiheet toteutetaan, sitä helpompaa sen varsinainen toteutus on. Kun kaikki rakennetta ja toiminnallisuutta koskevat haasteet on jo ratkaistu, voidaan tässä vaiheessa vain suoraviivaisesti toteuttaa ohjelmisto. Tosiasiassa näin ei kuitenkaan aina ole ja varsinkin isojen ja monimutkaisten tietojärjestelmien toteutusvaiheessakin esiin nousee uusia ongelmia ja vaatimuksia, joita ei ole osattu aiemmin ottaa huomioon huolellisesta valmistelusta huolimatta. Toteutuksen onnistuminen onkin usein kiinni siitä, kuinka hyvin toteutuksen aikana esiin tulevat vaatimukset käsitellään ja toteutetaan. (Pohjonen 2002, 34.) Projektin aikataulut voivat toteutusvaiheessa olla käymässä jo todella tiukoiksi. Siitäkin huolimatta myös uusien vaatimusten käsittelyn osalta on tärkeää, että niiden suunnittelussa otetaan huomioon käyttäjäkeskeisyys ja varmistetaan, että uudet vaatimukset ovat sekä selkeitä että täysin harkittuja kokonaistilanteeseen nähden. Myös toteutusvaiheessa on tärkeää tehdä jatkuvasti ratkaisuja, jotka parantavat tietojärjestelmän ylläpidettävyyttä. Eryityisesti toteutuksen osalta on pyrittävä tekemään pitkäkestoisia ja kestäviä ratkaisuja, vaikka aikataulu olisi käymässä tiukaksi.

Ennen tietojärjestelmän käyttöönottoa se tulee testata. Testauksen tarkoituksena on löytää ohjelmiston virheellinen toiminta, mutta täydellisen kattavan, kaikki virheet löytävän testauksen toteutus on usein mahdotonta. (Pohjonen 2002, 35-36.) Testaukseen

kannattaakin suhtautua niin, että jokainen löydetty virhe on parempi kuin virhe, jota ei löydetä. Testaus tulee suorittaa monipuolisesti huomioiden laaja-alaisesti koko tietojärjestelmä ja käyttää eri testausstrategioita (Pohjonen 20020, 36). Testauksen määrällä voidaan suoraan vaikuttaa tietojärjestelmän laatuun. Testauksen suunnittelussa kannattaa kiinnittää erityistä huomioita testauksen tekijöiden laajaan vaihteluun. Asiantuntijoiden suorittamassa testauksessa testataan, toimiiko sovellus vaatimusten ja määrittelyjen mukaisesti. Sovelluksen vastaaminen määrittelyjä on tärkeää, mutta määrittelyjä vasten testaaminen voi jättää huomiotta erilaisten käyttäjien toiminnan. Siksi on tärkeää suorittaa myös käyttäjätestausta, jossa käyttäjät pääsevät testaamaan tietojärjestelmää heidän todellisia työtehtäviään ja niiden tarpeita vasten. Näin voidaan selvittää, tukeeko tietojärjestelmä tulevien käyttäjien työtä. Ennen tietojärjestelmän käyttöönottoa suoritetaan hyväksymistestaus, jossa käyttäjät testaavat tietojärjestelmän toiminnallisuudet kokonaisuudessaan. (Stenberg 2006, 92.)

Vaikka käyttäjät eivät toteuta itse tietojärjestelmän toteutusta, on heidän osallistumisensa varmistaminen myös toteutusvaiheessa tärkeää. Tietojärjestelmää toteutettaessa nousee esiin uusia kysymyksiä ja vaatimuksia, jotka on tärkeä ratkaista käyttäjäkeskeisesti tiukoista aikatauluista huolimatta. Käyttäjäkeskeisen suunnittelun kolmas periaate koskee suunnitteluratkaisujen iterointia, jossa oleellista on jatkuva palautteen hakeminen käyttäjiltä. On siis tietojärjestelmän laadun kannalta välttämätöntä, että käyttäjät testaavat uutta tietojärjestelmää jatkuvasti, jotta heidän näkemyksensä voidaan ottaa huomioon mahdollisimman aikaisessa vaiheessa. Suunnitteluratkaisut voidaan esittää myös esimerkiksi erilaisten prototyyppien avulla. (Väänänen-Vainio-Mattila 2011, 106.) Protoilua on suositeltavaa tehdä toteutuksen yhteydessä, koska se konkretisoi ratkaisuja käyttäjille ja aktivoi käyttäjien palautteenantoa. Protoilun etuina on myös epäselvien vaatimusten tarkentuminen. (Stenberg 2006, 103-105.)

4.5.6 Käyttöönotto

Tietojärjestelmän kehittäminen huipentuu uuden tietojärjestelmän käyttöönottoon. Tarve tietojärjestelmän kehittämiselle on useimmiten peräisin toiminnan kehittämisestä, jossa on luotu uusia toimintatapoja ja uusia toimintatapoja tukemaan tarvitaan uusi tai päivitetty tietojärjestelmä (Stenberg 2006, 103). Tietojärjestelmän käyttöönotto ei siis useimmiten tarkoita ainoastaan tietojärjestelmän käyttöönottoa vaan samoihin aikoihin uuden tietojärjestelmän käyttöönoton kanssa muuttuu myös organisaation toiminta. Organisaatiolle on hyvin haitallista ajatella, että tietojärjestelmän käyttöönotto riittää, käyttäjät ottavat sen ilmielin käyttöön ja ryhtyvät noudattamaan uusia toimintatapoja. Todellisuudessa tietojärjestelmän käyttöönotto ja organisaation toimintatapojen muutos

vaatii vakuuttavaa muutosjohtamista, ohjeistusta, koulutusta sekä muutosvastarinnan käsittelyä. Kaikkiin näihin tulee varata riittävästi resursseja sekä aikaa. (Juvonen 2018, 54.) Hyvänä käytäntönä voidaan pitää erillisen käyttöönottosuunnitelman laatimista. Koska ei ole lainkaan itsestään selvää, että organisaatio omaksuu uudet toimintatavat, on hyödyllistä ottaa se erikseen huomioon käyttöönottosuunnitelmassa ja suunnitella tarvittavat toimenpiteet sitä varten. (Lehtimäki 2006, 175-176.) Uusien toimintatapojen koulutuksen lisäksi on tärkeää järjestää käyttäjille tarpeeksi laaja koulutus sovelluksen käyttöön (Stenberg 2006, 105-106).

Tietojärjestelmän käyttöönotto koskettaa luonnollisesti kaikkia tietojärjestelmän käyttäjiä, mutta heidän roolinsa tässä vaiheessa on enemmän käyttäjäkeskeinen kuin käyttäjälähtöinen. Onnistuneen käyttöönoton edellytykset rakennetaan jo tietojärjestelmän kehittämisen aikaisemmissa vaiheissa. Muutosten jalkautus ja läpivienti voidaan aloittaa jo hyvissä ajoin ennen tietojärjestelmän konkreettista käyttöönottoa, kun organisaation toiminnan muutostarpeet on tiedostettu, kuvattu ja kommunikoitu organisaatiolle. Jotta uusilla toimintatavoilla ja niitä tukevalla uudella tietojärjestelmällä pystyttäisiin saavuttamaan niillä tavoitellut hyödyt, tulee tietojärjestelmän käyttöönoton merkitys pystyä tiedostamaan ja aiemmissa vaiheissa. Näin siihen pystytään myös osoittamaan tarpeelliset resurssit eikä käyttöönoton tuki jää vajaaksi. (Juvonen 2018, 54–55.)

4.6 Dokumentointi

Dokumentaatiota voidaan helposti ajatella kehitysprojektin tylsänä ja pakollisena osiona, joka on tehtävä, mutta johon ei palata. Dokumentoinnin haasteena onkin se, että hyvän dokumentaation merkityksen ymmärtää vasta, kun se puuttuu tai on huonolaatuista ja hyvän dokumentaation tuottaminen jälkikäteen voi olla vaikeaa tai jopa mahdotonta. (Stenberg 2006, 120.) Dokumentaation tuottaminen passiivisesti vasta kun sille on tarvetta voi olla myös perusteltua, mutta dokumentaatio on parempilaatuista ja helpommin hyödynnettävissä, kun sitä tuotetaan ja päivitetään jatkuvasti. Koska dokumentaation tekemisen merkitystä tai niiden kokonaistarvetta ei välttämättä ymmärretä niiden tuottamisen aikana, voi kehitystyön isoimmat puutteet olla juuri dokumentaatiossa. Puutteet dokumentaatiossa voivat johtua myös aikataulupaineista; kiireen kasvaessa toimivan tuotteen toteuttamista pidetään tärkeämpänä kuin laadukasta dokumentaatiota, vaikka sen merkitys ymmärrettäisiinkin. Ihmiset voivat luottaa myös liikaa omaan muistiinsa tai siihen, että laadukas tuote korvaa puutteelliset dokumentaatiot. Dokumentaation ajan tasalla pitäminen on yhtä tärkeää kuin sen tuottaminen; vanhentunut tieto voi olla haitallisempaa kuin tiedon puuttuminen. Dokumentaatioiden päivittäminen muutostilanteissa tulisikin olla projektin sisäänrakennettuna

oletustoimintona. Pääsääntöisesti laadukasta dokumentaatiota ei voi olla liikaa, mutta dokumentaation oikea määrä riippuu projektin laajuudesta ja monimutkaisuudesta. Dokumentaatiota tehdään eri tarpeisiin ja eri kohderyhmille, esimerkiksi teknistä dokumentaatiota kehittäjille, ohjeita käyttäjille sekä projektinhallintaan liittyvää dokumentaatiota. (Pohjonen 2002, 79.)

Laadukas ja ajan tasalla oleva dokumentaatio tuottaa monia hyötyjä käyttäjän, käytettävyyden sekä tietojärjestelmän ylläpidon kannalta. Toteutettaessa uutta tietojärjestelmää tekevät sen suunnittelijat ja kehittäjät päivittäin useita ratkaisuja erikokoisiin ongelmiin. Osa ongelmista voivat koskea tietojärjestelmän ainutkertaisia toiminnallisuuksia, mutta osa ongelmista koskee toiminnallisuuksia, jotka toistuvat useassa paikassa läpi koko tietojärjestelmän, esimerkiksi prosessin käsittelyn siirtäminen toiselle käyttäjälle tai painikkeiden kielimuoto (verbi, sijamuoto jne.). Usein toistuvien toiminnallisuuksien ratkaisut ja toteutustavat on hyödyllistä dokumentoida kaikille organisaation jäsenille ymmärrettävään ja käytettävään muotoon. (Vesselov & Davis 2019, 17.) Kun kerran tehty ratkaisu ja päätetty toteutus on kaikkien käytettävissä, ei samaa ongelmaa tarvitse ratkoa useamman kerran ja aikaa vapautuu muiden ongelmien ratkaisemiseen tietojärjestelmän suunnittelussa, toteutuksessa sekä testaamisessa (Vesselov & Davis 2019, 28-29). Toteutettujen ratkaisujen tarkka ja avoin dokumentointi vähentää myös määrittelyjen väärinymmärrysten sekä toteutuksen epäjohtonmukaisuuksien riskiä ja helpottaa uusien työntekijöiden perehdyttämistä sovelluksen toimintaan. Kun aiemmin toteutettujen toiminnallisuuksien kuvaukset ovat kaikkien saatavilla, voidaan usein toistuvat toiminnallisuudet toteuttaa tietojärjestelmään samanlaisina sen toteuttaneesta kehittäjästä huolimatta. (Vesselov & Davis 2019, 38.) Tämä kasvattaa tietojärjestelmän yhtenäisyyttä ja johdonmukaisuutta, mikä kasvattaa omalta osaltaan tietojärjestelmän käytettävyyttä (Vesselov & Davis 2019, 39). Kun samanlaiset toiminnallisuudet toimivat samanlaisina läpi tietojärjestelmän, käyttäjät myös oppivat käyttämään tietojärjestelmää nopeammin. Osa käyttäjistä opettelee tietojärjestelmän käytön niin, että he toistavat tiettyjä toimenpidesarjoja niin usein, että he osaavat tehdä ne virheettömästi. Tällöin pienikin prosessikohtainen eroavaisuus näennäisesti muualta tutusta toiminnallisuudessa hidastaa käyttäjän oppimista, koska yksikin ero toimenpidesarjan liikkeissä tarkoittaa useamman toimenpidesarjan opettelua. (Sinkkonen ym. 2006, 233.) Yhtenäiset ja toimivaksi testatut toiminnallisuudet ja niiden dokumentointi myös lyhentää aikasykliä, minkä kuluessa uudet toiminnallisuudet saadaan toimitettua loppukäyttäjälle (Vesselov & Davis 2019, 42-43).

4.7 Käyttäjäkeskeisen kehittämisen haasteet

Kirjallisuuden perusteella panostamalla käyttäjien osallistumiseen tietojärjestelmän kehittämiseen organisaatio voi saavuttaa monia hyötyjä, jotka liittyvät sekä itse tietojärjestelmän laatuun että organisaation toiminnan kehittämiseen.

Käyttäjäkeskisyyteen liittyy kuitenkin myös haasteita, joiden vuoksi käyttäjien osallistumiseen ei välttämättä kohdisteta tarpeeksi resursseja tai käyttäjien osallistumiseen liittyy ongelmia, joiden vuoksi sillä ei saavuteta sille asetettuja tavoitteita. Useimpien käytettävyyssmenetelmien toteuttaminen vaatii paljon resursseja ja niihin käytettyjen resurssien hyötyvaikutuksia ei välttämättä hahmoteta. Sen lisäksi haasteita voi aiheuttaa kehittämistyön eri roolit, kommunikaatio ja käyttäjien tavoittaminen, jos niitä ei huomioida käyttäjien osallistumisen suunnittelussa. Tietojärjestelmän onnistumisen kannalta on oleellista, että käyttäjät osallistuvat sen kehittämiseen, mutta myös käyttäjien onnistunut osallistuminen vaatii panostusta ja suunnittelua (Pohjonen 2002, 49).

4.7.1 Kehittämistyön roolit

Tietojärjestelmän kehittämiseen osallistuu ihmisiä monissa eri rooleissa ja usein vastuiden ja tehtävien tarkkarajainen jakaminen eri rooleille on haastavaa. Periaatteessa on selkeää, että tietojärjestelmän kehittäjä vastaa tietojärjestelmän rakentamisesta vastaamaan käyttäjien tarpeisiin ja käyttäjä käyttää tietojärjestelmää. Tietojärjestelmän kehittäjällä tarkoitetaan määrittelijää, suunnittelijaa sekä ohjelmoijaa. Tietojärjestelmän tehtävänä on ratkaista käyttäjän tietojenkäsittelyongelmia, mutta ratkaisemiseen tarvittava tietämys organisaatiosta on saatavissa vain tietojärjestelmän tulevilta käyttäjiltä. (Pohjonen 2002, 46.) Tietojärjestelmälle asetetut vaatimukset dokumentoidaan kehittämisen vaatimusmäärittely-vaiheessa ja tietojärjestelmä tulisi suunnitella ja toteuttaa vastaamaan näihin vaatimuksiin. Todellisuudessa vaatimuksiin joudutaan usein palaamaan ja tekemään tarkennuksia suunnittelu- ja toteutusvaiheissa. Näissä vaiheissa onkin hyödyllistä toisinaan muistaa kehittäjän ja käyttäjän roolien erot. Kehittäjän ammattitaito on tietojärjestelmän rakentamisessa ja käyttäjän työtehtävissä, joita tietojärjestelmä tukee. Kehittäjän ei tarvitse osata tehdä käyttäjän työtä, mutta hänen täytyy ymmärtää käyttäjän vaatimus ja sen taustat, jotta hän voi rakentaa tietojärjestelmän. Käyttäjän ei tarvitse osata suunnitella tietojärjestelmää, vaan esittää tarpeet, jotka tietojärjestelmän tulisi ratkaista. Kehittäjän ja käyttäjän on hyödyllistä työskennellä yhdessä, mutta roolien liiallinen sekoittuminen voi johtaa huonoihin ratkaisuihin tietojärjestelmän toiminnassa. Karkeasti jaoteltuna käyttäjän tulisi vastata kysymyksiin "mitä" ja "miksi" ja kehittäjän kysymykseen "miten" (Stenberg 2006, 124).

Käyttäjien osallistuminen suunnitteluun voi aiheuttaa myös turhaa työtä. Käyttäjät saattavat toivoa tuotetta, jossa on kaikki mahdolliset toiminnallisuudet, mutta joka ei kuitenkaan tue heidän työtään. Siksi tietojärjestelmän suunnittelijoilla tulee olla selkeä näkemys tietojärjestelmän käytöstä kokonaisuutena, jotta he osaavat kriittisesti arvioida sen eri toiminnallisuuksien tarpeellisuuden. (Väänänen-Vainio-Mattila 2011, 109.)

Tietojärjestelmän käytön pidentämiseksi tulee jo sen suunnittelu- ja toteutusvaiheissa kiinnittää huomiota tietojärjestelmän ja sovelluksen ylläpidettävyystekijöihin. Suunnittelu ja toteutus ovatkin monessa suhteessa hyvin lähekkäisiä ja jopa päällekkäisiä vaiheita. Tietojärjestelmän suunnittelijalla olisikin hyödyllistä olla tietämystä myös järjestelmien toteuttamisesta ja siihen liittyvistä seikoista, jotta suunnitelmien muuttamisessa toimivaksi järjestelmäksi pystyttäisiin huomioimaan ylläpidettävyystekijät. (Pohjonen 2002, 35.)

4.7.2 Viestintä ja vuorovaikutus

Yksittäisen organisaation ja tietojärjestelmän kehittäminen on laaja kokonaisuus, johon osallistuu lukuisia eri osapuolia kehittämisen kohteena olevasta organisaatiosta sekä muista organisaatioista. Osapuolien taustoissa, työskentelytavoissa, osaamisessa ja oman työn tavoitteissa voi olla suurta vaihtelua ja niiden yhteensovittaminen vaatii suunnittelua, yhteisesti sovittuja toimintamalleja ja niiden jatkuvaa kehittämistä. Työ on vaativaa, mutta onnistuneen lopputuloksen edellytys. (Stenberg 2006, 124.) Kun projektissa työskentelee lukuisia henkilöitä eri organisaatioista eri taustoilla, vaatii se toimiakseen aktiivista kommunikointia kaikilta osapuolilta. Projektissa on varmistettava kaikkien osapuolien tarpeellinen tiedonsaanti suunnittelemalla projektille sopivat viestintä-, seuranta- ja kommunikointikäytänteet. (Juvonen 2018, 85-87.)

Pelkkä kommunikoinnin aktiivisuus ja oikeat käytänteet eivät aina riitä, vaan haasteita voi aiheuttaa pelkästään sekin, että tietojärjestelmän käyttäjä ja kehittäjä tulevat usein hyvin eri maailmoista. Tietojärjestelmää rakennettaessa kehittäjä haluaa useimmiten tietää järjestelmän ominaisuuden tietoja, esimerkiksi käyttöliittymään ja tietojärjestelmän tietojen käsittelylogiikkaan liittyviä yksityiskohtia. Käyttäjän asiantuntijuus liittyy taas oman työnsä tehtäviin ja sisältöihin ja organisaation tavoitteisiin ja toimintaan. Vaikka kehittäjä ja käyttäjä keskustelevat samasta aiheesta, voi heidän näkökulmansa aiheeseen aiheuttaa kommunikaatio-ongelmia, väärinymmärryksiä ja pahimmillaan myös tietojärjestelmän virheellistä toimintaa. (Pohjonen 2002, 49-50.) Kehittäjien ja käyttäjien erilaiset taustat voivat aiheuttaa myös tiedon käsittelyä eri tavalla ja eri terminologialla, mikä voi aiheuttaa väärinymmärryksiä.

Vaikka tietojärjestelmän kehittäjän ja käyttäjän kommunikointi voi tuntua haastavalta, on heidän onnistuneen lopputuloksen aikaansaamiseksi pyrittävä kommunikoimaan

kummankin ymmärtämällä tavalla. Kommunikaation parantamiseksi on tiedostettava henkilöiden erilaiset taustat ja osaamisalueet ja suhtauduttava eroihin ymmärtävällä asenteella. Henkilöiden erilaiset tietämyserot voi ottaa huomioon esimerkiksi tietoisesti välttämällä käyttämästä oman alan slangia, mikäli keskustelun osapuolet tulevat eri aloilta. (Koistinen 2002, 53.) Kehittämällä tietoisesti kommunikaatiota voidaan varmistaa, että kaikki ymmärtävät sovitut asiat samalla tavalla. Näin voidaan välttää turhaa työtä aiheuttavat väärinymmärrykset ja saada kaikkien osapuolien arvokas tietotaito hyötykäyttöön.

Tietojärjestelmän kehittäjä ja käyttäjä pyrkivät pääsemään yhteiseen ymmärrykseen siitä, miten tietojärjestelmän tulisi toimia ja erityisesti, miten käyttäjä tietojärjestelmää käyttää. Henkilöiden erilaiset taustat, ymmärrys tietojärjestelmän toiminnasta ja termit voivat aiheuttaa tähän haasteita. Ketterät menetelmät, iteroiva työote ja erilaiset pilotit ovat houkuttelevia, koska niitä käyttämällä kehittäjä ja käyttäjä eivät ole ainoastaan sanallisen kommunikaation varassa, vaan kehittäjä voi konkreettisesti näyttää, miten tietojärjestelmä voisi toimia ja käyttäjä voi arvioida, vastaako tietojärjestelmä hänen tarpeitaan. Käyttäjän mahdollisuus päästä kokeilemaan tietojärjestelmän eri versioiden käyttöä kuluttaa kuitenkin resursseja, joten tehokkaampaa on mahdollisuuksien mukaan pyrkiä parantamaan kommunikaatiota muilla keinoin. Tavoitteena on käyttäjän substanssiosaamisen ja kehittäjän teknisen osaamisen tehokas yhdistäminen.

Uutta tuotetta, esimerkiksi tietojärjestelmää, kehitettäessä syntyy usein uutta, juuri kyseiseen tietojärjestelmään tai sen kehittämiseen liittyvää terminologiaa. Terminologia sovitaan tai se muotoutuu tuotekehitystiimin sisällä. Nämä termit jäävät helposti mukaan lopulliseen tuotteeseen, mikä heikentää sen käytettävyyttä. (Sinkkonen ym. 2006, 31.) Yhteiset termit voivat helpottaa kommunikaatiota ja vähentää väärinymmärryksiä kehitystiimin sisällä, mutta eivät palvele tuotteen käyttäjiä, jotka eivät ole olleet kehittämisessä mukana.

4.7.3 Käyttäjien tavoittaminen

Käyttäjakeskeisen kehittämisen tärkein toiminta on käyttäjien kuunteleminen ja heidän näkemyksiensä selvittäminen. Jos käyttäjien lukumäärä on kuitenkin korkea, voi haasteeksi muodostua heidän kaikkien kuunteleminen. Käyttäjät eivät myöskään ole joukko samanlaisia käyttäjiä, vaan heidän osaamisensa, näkemyksensä ja taitonsa vaihtelevat. Käyttäjakeskeisen kehittämisen haasteena onkin se, miten saadaan selville suurimman osan käyttäjien näkemyksistä. On tyypillistä, että tietojärjestelmän kehittämiseen, esimerkiksi vaatimusmäärittelyihin, osallistuu organisaation esimiestasolta henkilöitä, jotka tuntevat organisaation toiminnan ja pystyvät sinällään määrittelemään

vaatimuksia, mutta he eivät välttämättä ole kuitenkaan tietojärjestelmän loppukäyttäjiä tai tunne loppukäyttäjien työnkuvaa tarpeeksi tarkalla tasolla. Koko kehittämisen ajan on tärkeää, että kehittämiseen osallistetaan juuri niitä henkilöitä, jotka eniten tulevat käyttämään uutta tietojärjestelmää päivittäisessä työssään. (Juvonen 2018, 58.) Organisaation työkäyttöön tarkoitetun tietojärjestelmän käyttäjiä voi olla myös vaikea tavoittaa tai heillä ei välttämättä ole aikaa osallistua kehittämiseen (Väänänen-Vainio-Mattila 2011, 115). Organisaatiossa ei välttämättä osata motivoida käyttäjiä osallistumaan kehityshankkeisiin tai osallistumista ei oteta huomioon muita työtehtäviä vähentämällä (Pohjonen 2002, 50).

Hilkka Merisalo-Rantasen tutkimuksen tulokset osoittavat, että peruskäyttäjät eivät ole kovin aktiivisia antamaan palautetta tai halukkaita osallistumaan omasta aloitteestaan. Mikäli palautteen antaminen ei automaattisesti kuulu heidän työtehtäviinsä, tarvitsevat he osallistumiseen kannustimen tai muun motivaattorin. Tämä ei päde kuitenkaan kaikkein tyytymättömämpiin tai tyytyväisimpiin käyttäjiin. Henkilökohtainen yhteydenotto peruskäyttäjään voi kuitenkin toimia tarvittavana kannustimena sekä kasvattaa käyttäjän myönteistä asennetta sekä tietojärjestelmää että palautteen antamista kohtaan. (Merisalo-Rantanen 2011, 52-53.) Palautteen saamisen ja hyödyntämisen kannalta on yhtä tärkeää se, että palautteen antaja saa palautteestaan jonkinlaisen vastauksen kuin se, että palaute käsitellään ja se saavuttaa sitä hyödyntävät ihmiset (Merisalo-Rantanen 2011, 53-54).

5 Tietojärjestelmän ylläpito

Tietojärjestelmän elinkaaren pisin vaihe on sen käyttö ja käytönaikainen ylläpito. Ylläpidolla tarkoitetaan kaikkien tietojärjestelmän osien muutos- ja kehitystyötä, joka tehdään tietojärjestelmän käyttöönoton jälkeen. (Stenberg 2006, 114.) Ylläpidon käsitettä käytetään joskus tarkoittamaan vain tietojärjestelmässä olevien virheiden korjaamista, jolloin muita tietojärjestelmään tehtäviä muutostöitä kutsutaan pienkehitykseksi (Koistinen 2002, 20). Eri ihmisillä on eri käsityksiä siitä, mitä kaikkea ylläpito kattaa tai mitä käsite tarkoittaa eikä sitä olekaan yksiselitteisesti määritelty kaikkiin tilanteisiin sopivaksi. Tehokkaasti organisoitu ylläpito voi kuitenkin mahdollistaa organisaatiolle tietojärjestelmän virheiden korjauksen lisäksi myös tietojärjestelmän tehokkaan ja hyvin suunnitellun jatkokehittämisen ja uusien asioiden toteuttamisen. Kun ylläpito ymmärretään muunakin kuin virheiden korjaamisena, sen merkitys ja arvostus myös kasvaa. (Koistinen 2002, 35.) Uuden tietojärjestelmän virheet saadaan pääsääntöisesti korjattua noin vuoden kuluessa käyttöönotosta, jonka jälkeen valtaosa ylläpitotöistä on jatkokehittämistä (Koistinen 2002, 39). Tässä tietoperustassa ylläpidolla tarkoitetaan kaikkea työtä, jota tietojärjestelmään tehdään sen käyttöönoton jälkeen.

Ylläpito kohdistuu kaikkiin tietojärjestelmän osiin, joista ylläpitovaiheessa useimmiten eniten huomiota saa sovelluksen ylläpito. Se voidaan jakaa korjaavaan, mukauttavaan, kehittäväan ja ennakoivaan ylläpitoon. Näistä isoin osuus on useimmiten kehittävä ylläpito eli uusien ominaisuuksien kehittäminen sovellukseen. Tämä voi olla merkki siitä, että aiemmissa kehittämissä vaiheissa on unohdettu jotain oleellista tai olla vain luonnollista ajan myötä ilmenevien muutostarpeiden toteuttamista. (Stenberg 2006, 114.) Muutosten luokittelu tuottaa tietoa siitä, minkä tyyppisiä muutoksia sovellukseen tehdään ja näin tietoa järjestelmän tilasta. Yksittäinen ylläpitotehtävä voi sisältää eri luokkiin kuuluvia töitä, jolloin on kannattavaa purkaa tehtävä luokkien mukaisiin osiin. (Koistinen 2002, 147.)

Tietojärjestelmän ylläpito on myös olennainen osa organisaation toiminnan kehittämistä. Tästä syystä ylläpidon tehtäviä on seurattava ja kehitettävä sekä tietotekniikan että liiketoiminnan näkökulmasta. Tietojärjestelmään tehtävillä muutoksilla voidaan mahdollistaa liiketoiminnan kehittäminen ja toisaalta on myös seurattava, ovatko kaikki ylläpidon tehtävät edelleen oleellisia liiketoiminnan kannalta. (Koistinen 2002, 30.) Tehokkaalla ylläpidolla voidaan mahdollistaa ja tukea liiketoiminnan kehittämistä ja sopeuttamista muutoksiin, mutta vastuu prosessien muutostoimenpiteistä on liiketoiminnan edustajilla (Koistinen 2002, 93). Ylläpitoon ei tulisi suhtautua tietojärjestelmän pakollisena käytön mahdollistajana, vaan seurata jatkuvasti, onko siihen panostetut resurssit oikeassa suhteessa sillä saavutettuihin hyötyihin. Yksi liiketoiminnan

ylläpidolle asettama tavoite onkin, että ylläpidolle osoitetut resurssit palauttavat liiketoiminnalle sen odottaman lopputuloksen ja siksi ylläpidon toimintamallien on oltava tehokkaita (Koistinen 2002, 93).

5.1 Ylläpidon haasteet

Ylläpitovaiheessa haasteita voi aiheuttaa esimerkiksi aika- ja resurssipula, muutoshallinnan organisointi, vastuu ja priorisointi sekä erilaiset tekniset ympäristöt (Stenberg 2006, 114). Yleinen ylläpidon haaste on myös vajavainen tai epäselvä dokumentaatio. Etenkin jos tietojärjestelmän kehitystyö on jatkunut kauan, on ylläpitovaiheessa erityisen tärkeä pystyä selvittämään tietojärjestelmän taustalla vaikuttavat suunnittelu- ja toteutusratkaisujen perusteet. (Pohjonen 2002, 37–38.) Onnistuneen ylläpitovaiheen edellytykset rakennetaan jo aiemmissa kehitystyön vaiheissa, mikäli kehitysorganisaatiossa ymmärretään ylläpitovaiheen merkitys ja osataan ottaa se huomioon.

Ylläpitovaihetta edeltäneillä vaiheilla onkin suuri merkitys ylläpitovaiheen sujuvuuteen. Kaikissa tietojärjestelmän kehittämisen vaiheissa tehdään virheitä, mutta virheen korjaamiseen käytettyjen resurssien kannalta sillä on suuri merkitys, missä vaiheessa virhe tapahtuu. 82 prosenttia kaikesta ylläpitoon kuluvesta työajasta käytetään sellaisten virheiden korjaamiseen, joka on tapahtunut tietojärjestelmää määriteltäessä ja 13 prosenttia työajasta kuluu teknisten suunnitteluvirheiden korjaamiseen. Virheen korjaaminen on sitä kalliimpaa, mitä myöhäisemmässä vaiheessa se havaitaan eli panostamalla tietojärjestelmän suunnitteluun voidaan ylläpidon kustannuksia pienentää huomattavasti. (Koistinen 2002, 41-42.)

Yleinen ylläpidon haaste on sen organisointi ja toimintamalli sen suhteen, miten ylläpidon tehtäviä hallitaan ja muutosten toimeksiantoja käsitellään. Tietojärjestelmän käyttäjästä voi tuntua houkuttelevan helpolta lähettää muutostarpeen toimeksianto ylläpitotaholle esimerkiksi soittamalla tai sähköpostilla. Tällaisen tehtävänannon epämääräisyys voi kuitenkin aiheuttaa ongelmia ylläpidossa. Muutostarpeesta ei välttämättä selviä tarpeeksi tarkasti, mitä muutoksia halutaan tehtävän tai tarvetta ei ole kuvattu tarpeeksi selkeästi. Tässä kohden sekä käyttäjä, että ylläpitotehtävän tekijä voi ajatella, että muutoksen tekijä tuntee käyttäjien tehtävät ja tarpeet niin hyvin, että hän osaa toteuttaa tarpeisiin vastaavan muutoksen ilman tarkempaa määrittelyä. Kuitenkin lopputulosta testatessa voi käydä ilmi, ettei lopputulos olekaan sellainen, mitä muutoksen pyytjä tarkoitti. Pahimmillaan voi käydä niin, että muutoksen toteuttamisen jälkeen käykin ilmi, että muutospyyntö oli tehty liian hätäisesti eikä muutos olisikaan ollut tarpeellinen. (Koistinen 2002, 49.)

Muutostoiveiden esittäminen ylläpito-organisaatiolle tulee olla käyttäjille mahdollisimman helppoa, mutta se ei kuitenkaan saa olla hallitsematonta tai dokumentoimatonta.

Ylläpidolle voi saapua runsaasti erilaisia muutostarpeita. Haasteita voi aiheuttaa myös niiden käsittely ja priorisointi. Etenkin jos tietojärjestelmällä on paljon käyttäjiä, voivat eri käyttäjät toisistaan tietämättä pyytää saman kaltaisia muutoksia samaan asiaan. Näiden samaan asiaan liittyvien muutostarpeiden toteuttaminen yksitellen kuluttaa enemmän resursseja kuin niiden toteuttaminen samanaikaisesti. Hallitulla muutosten käsittelyllä voidaan tunnistaa tietyllä aikavälillä saapuneet samaan asiaan liittyvät muutostarpeet ja tehostaa resurssien käyttöä toteuttamalla ne samanaikaisesti. Eri käyttäjiltä tai tahoilta saapuneet samaan asiaan liittyvät muutostarpeet voivat kuitenkin olla myös ristiriidassa keskenään. Toteuttamalla kaikki saapuneet muutostarpeet ilman niiden arviointia ja käsittelyä, voidaan helposti ajautua tilanteeseen, jossa ensin toteutetaan yksi muutos ja myöhemmin toinen, joka kumoaa ensimmäisen, koska käyttäjät ovat toivoneet eri asioita. Tällaiset muutokset ovat turhaa resurssien käyttöä. Muutostarpeet tulee käsitellä ja priorisoida, jotta voidaan varmistua siitä, että ylläpidossa tehdään oikeita asioita. (Koistinen 2002, 51.)

Muutostarpeiden käsittely ja priorisointi on hankalaa, mikäli tietojärjestelmän omistajuutta ei ole osoitettu organisaatiossa kenellekään tai määritelty tarpeeksi selkeästi. Tietojärjestelmän sisällön oikeellisuus ja se, että tietojärjestelmä todella tukee käyttäjien toimintaa, tulisi selkeästi osoittaa tietyn henkilön vastuulle. Tällä henkilöllä tulisi olla riittävästi tietoa ja ymmärrystä käytännön työskentelystä ja koko tietojärjestelmästä, jotta hän pystyy priorisoimaan muutostyöt tehokkaasti välttämättä turhaa ja päällekkäistä työtä. Lisäksi henkilöllä tulee olla oikeus tehdä tarvittavat tietojärjestelmän toimintaan liittyvät päätökset. (Koistinen 2002, 52.)

5.2 Ylläpidon organisointi

Kuten uuden tietojärjestelmän kehittämisessä, myöskään kaikki ylläpidon haasteet eivät liity tietoteknisiin asioihin. Ylläpito tuleekin organisoida, kuvata toimintamallein ja ohjeistuksin, jotta ylläpito olisi mahdollisimman tehokasta. (Koistinen 2002, 19.) Ylläpidon organisointia tulee tarkastella ylläpidon tavoitteiden kautta, jotta voidaan varmistua siitä, että organisaatiolla on edellytykset saavuttaa ylläpidon tavoitteet. Ylläpidon organisointi vaatii myös jatkuvaa tarkastelua ja kehittämistä, minkä vuoksi onkin kannattavaa nimetä ylläpidon kokonaiskehittämisestä vastaava henkilö. (Koistinen 2002, 76-77.)

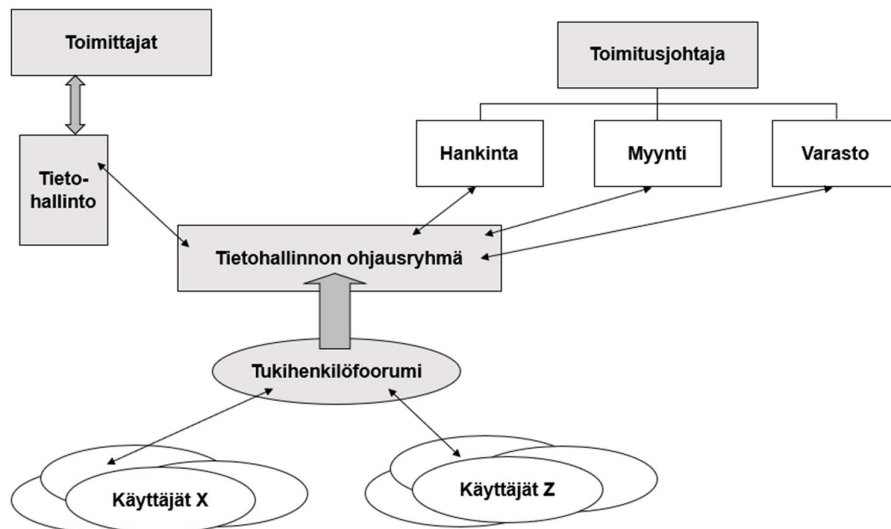
Tietojärjestelmän koko elinkaaren kustannuksista vain noin 30 prosenttia kuluu tietojärjestelmän kehittämiseen ja 70 prosenttia sen ylläpitoon. Ylläpidon resurssit on siis tärkeää kohdentaa oikeisiin asioihin ja siksi panostaa myös ylläpidon tehokkaaseen

organisointiin. (Koistinen 2002, 28.) Nykyään tietojärjestelmän kehittämisen ja ylläpidon raja ei ole enää niin selkeä, vaan kehittämistä ja ylläpitoa tapahtuu paljon myös rinnakkain.

Ylläpidon toimintatavat on hyödyllistä kuvata prosesseina, jotta muutokset tapahtuvat hallitusti ja jotta kaikki ymmärtävät ylläpidon eri osapuolien roolit ja vastuut samalla tavalla. Ylläpidon prosessikuvauksissa kuvataan, miten yksittäinen ylläpitotehtävä etenee organisaatiossa tarpeesta tuotantokäyttöön. Prosessikuvauksia on hyödyllistä tehdä eri tarpeisiin eri tarkkuustasoilla eri prosessien osakokonaisuuksista. Ylläpitoprosesseja on myös jatkuvasti kehitettävä. (Koistinen 2002, 125.)

Tietojärjestelmän ylläpidon tehtävissä on paljon samoja elementtejä kuin uuden tietojärjestelmän kehittämisessä, joten ylläpitoon ei tarvitse suhtautua uuden tietojärjestelmän kehittämisestä täysin poikkeavalla tavalla. Uuden tietojärjestelmän kehittämisen menetelmät voivat olla liian raskaita etenkin pienempiin ylläpitotehtäviin, mutta menetelmiä on mahdollista soveltaa ja käyttää joustavasti myös ylläpidon tehtäviin. Myös ylläpitovaiheessa on tärkeää huolehtia, ettei mikään oleellinen tehtävä jää tekemättä ja toimivan menetelmän käyttäminen voi edesauttaa tätä tavoitetta. (Koistinen 2002, 115.) Tietojärjestelmän ylläpitovaiheeseen pätee sama periaate kuin sen kehittämiseen: Jostain lähteestä nousee esiin tarve tehdä muutoksia (mitä), sen jälkeen suunnitellaan, miten tarpeeseen vastataan, sitten toteutetaan ja lopuksi testataan. Eri systeemityömallien hyödyntäminen myös ylläpitotehtävissä vähentää ylläpidon sidonnaisuutta tiettyihin henkilöihin, jotka juuri nyt tuntevat tietojärjestelmän, koska mallien tarjoama systemaattisuus varmistaa tiedon siirtymisen ja säilymisen organisaatiossa (Koistinen 2002, 108-109).

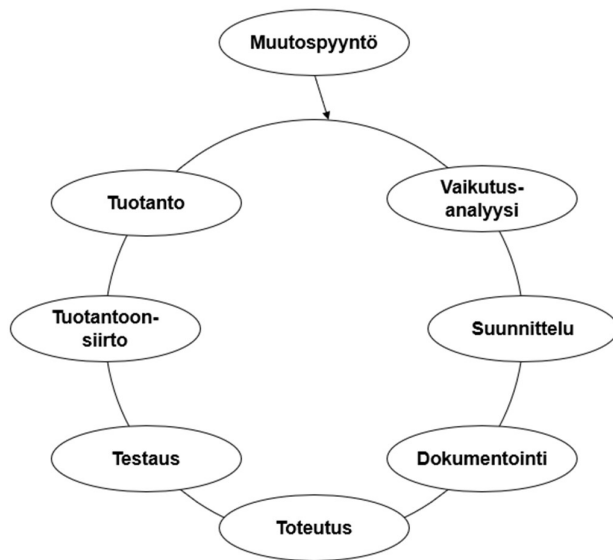
Kuviossa 4 on kuvattuna yksi tapa organisoida sovelluksen ylläpito. Siinä käyttäjiltä sovellusalueittain kerätyt muutos- ja lisäpiirteet käsitellään alustavasti tukihenkilöfoorumissa. IT-yhdyshenkilöt keräävät muutostarpeet. Lopulliset päätökset siitä, mitä toteutetaan, tehdään tietohallintofoorumissa, jossa on edustettuna myös liiketoimintayksiköt. (Stenberg 2006, 115.)



Kuvio 4. Ylläpidon organisointi (mukaillen Stenberg 2006, 115)

Ylläpitovaihe on olennainen osa systeemyön elinkaarta ja sen organisoimiseen on käytettävä tarpeeksi resursseja, jotta se on tehokasta. Ylläpidon organisoimisen perustana voidaan tarkastella ylläpitäjän työn eri vaiheita yksittäisen ylläpitotehtävän toteuttamisen näkökulmasta. Tämä auttaa hahmottamaan ylläpitäjän tehtäviä ja niiden järjestystä. Ylläpitotehtävä käynnistyy, kun tulee ilmi tehdä muutos tietojärjestelmään. Muutostarpeet tehdään kirjallisena ja sen tekee useimmiten käyttäjä. Oleellista on, että kaikki tarpeelliset osapuolet tietävät, miksi muutosprosessi on käynnistynyt ja siksi se myös perustellaan muutospyyntönsä. Muutostarpeen ilmitulon jälkeen sen vaikutukset analysoidaan. Analysoinnissa kartoitetaan, mihin kaikkialle eri järjestelmissä sekä käyttäjän toiminnoissa muutoksella on vaikutusta, jotta voidaan välttyä suunnittelemattomilta muutoksilta. Analyysissa myös vertaillaan eri ratkaisuvaihtoehtojen kustannuksia ja laaditaan mahdollinen toteutusaikataulu. Mikäli muutos päätetään toteuttaa, muutoksen toteutus, testaus, koulutus ja ohjeistus suunnitellaan. Muutos voidaan dokumentoida jo ennen sen toteutusta. Suunnitellun muutoksen dokumentaatio voidaan käydä käyttäjien kanssa yhdessä läpi ja näin varmistaa ennen sen toteuttamista, että muutos on toivotunlainen ja vastaa tarpeisiin. (Koistinen 2002, 105-107.)

Toteutuksen jälkeen muutos testataan. Testauksessa on huolehdittava, että sovellus toimii teknisesti oikein ja määrittelyjen mukaisesti ja että muutokset vastaavat sisällöltään haluttua lopputulosta. Testausta on siis tehtävä useammasta näkökulmasta. Mikäli toteutettu tuotos hyväksytään, se siirretään tuotantoon sovitun proseduurin mukaisesti. Sovittuun proseduriin vaikuttaa muun muassa ylläpidon projektointi tai versiointi. (Koistinen 2002, 107-108.) Prosessi on kuvattuna kuviossa 5.



Kuvio 5. Ylläpitomalli (mukaillen Koistinen 2002, 106)

Ylläpidon tulisi olla hyvin organisoitua ja sen prosessit kuvattuna jo ennen ylläpitovaiheen alkua, sillä muutostarpeiden määrä (sekä korjaavien että kehittävien) on suurimmillaan heti tietojärjestelmän käyttöönoton jälkeen (Koistinen 2002, 149).

5.3 Muutostarpeiden aiheuttajat

Jotta tietojärjestelmään voidaan tehdä muutoksia, pitää muutoksen taustalla olla tarve. Tarve muutokselle voi tulla organisaation ulkopuolelta tai organisaation sisältä. Organisaation ulkopuolisiin muutostarpeisiin organisaatio voi vaikuttaa vähemmän kuin sisäisiin. Sisäiset muutostarpeet voivat tulla esiin yllättäen, mutta organisaatio voi myös käyttää resursseja sisäisten muutostarpeiden selvittämiseen ja esiin tuomiseen. Tyypillisimpiä tietolähteitä tietojärjestelmän muutostarpeille ylläpitovaiheessa ovat käytettävyystudkimukset, help desk -toiminnan tukipyynnöt sekä työntekijöiltä ja käyttäjiltä tulevat aloitteet ja palautteet (Stenberg 2006, 115).

Tietojärjestelmään kohdistuvien organisaation ulkopuolelta tulevien muutostarpeiden taustalla voivat olla esimerkiksi lainsäädännön tai muun organisaation toimintaa ohjaavien määräyksien muuttuminen. Organisaatioilla on vain rajalliset mahdollisuudet vaikuttaa näihin muutoksiin ja niiden aiheuttamia muutoksia tietojärjestelmään ei voi jättää toteuttamatta. Tästä syystä on tehokasta mahdollisuuksien mukaan tunnistaa jo tietojärjestelmän toteutusvaiheessa sellaiset osiot, joihin todennäköisesti kohdistuu lainsäädännön muutosten tuomia vaikutuksia. Tunnistamalla kyseiset kohdat ne voidaan toteuttaa ohjelmistossa kevyesti muutettaviksi ja ylläpidettäviksi. Näin pystytään parantamaan ohjelmiston ylläpidettävyyttä. (Koistinen 2002, 85.) Organisaation

ulkopuolelta tulevia muutostarpeita voivat aiheuttaa myös kilpailevien yritysten toiminta ja siihen vastaamisen tarve tai muutokset yhteistyöorganisaatioiden tietojärjestelmiin ja niiden yhteensovittaminen omaan tietojärjestelmään. Nämä muutokset ovat tarpeellisia, mutta eivät niin välttämättömiä kuin lainsäädännön muutosten aiheuttamat muutostarpeet. Organisaation on kuitenkin hyvä tiedostaa, minkä ulkopuolisten tahojen toiminta voi aiheuttaa muutostarpeita heidän tietojärjestelmäänsä ja seurata näiden toimintaa, jotta muutostarpeisiin pystytään varautumaan hallitusti. (Koistinen 2002, 85-87.)

Organisaation oma toiminta saattaa myös aiheuttaa muutostarpeita tietojärjestelmään. Kun organisaatio siirtyy uudelle toiminta-alueelle, on uuden toiminnan oltava mahdollista tietojärjestelmässä ja usein uutta toimintaa halutaan myös seurata raportein. Myös aiempaa toimintaa voidaan haluta tehostaa uudella tavalla, johon tarvitaan tietoja, joita ei välttämättä ole lainkaan tallennettu tietojärjestelmään. Organisaation omat toimintatapa- ja organisaatiomuutokset aiheuttavat kuitenkin eniten muutostarpeita tietojärjestelmiin, joten erityisesti niihin on syytä varautua jo tietojärjestelmän suunnitteluvaiheessa. Varautumalla muutoksiin jo etukäteen voidaan varmistua, etteivät ne aiheuta isoja muutostöitä tietojärjestelmään. Vaikka organisaation tulevia muutoksia ei olisi vielä julkistettu tai niitä ei olisi suunniteltu loppuun asti, on organisaation johdolla useimmiten tieto niistä jo hyvissä ajoin. Onkin tärkeää, että tietojärjestelmän suunnittelijoilla on tiedossaan tulevat muutokset tarpeellisella tasolla, jotta he osaavat ottaa ne tietojärjestelmän suunnittelussa huomioon. (Koistinen 2002, 82-83.)

Tietojärjestelmään voi kohdistua muutostarpeita myös siksi, ettei se vastaa käyttäjien tarpeita eli tietojärjestelmä toimii virheellisesti. Usein tietojärjestelmän virheet mielletään vain ohjelmointivirheiksi, mutta käyttäjien tarpeita vastaamaton toiminta voi olla seurausta myös puutteellisesta määrittelystä tai puutteellisesta tietoteknisestä suunnittelusta. Puutteellinen määrittely voi johtua siitä, etteivät suunnittelijat ole olleet varmoja, miten tietojärjestelmän tulisi toimia sillä hetkellä tai tulevaisuudessa tai he eivät ole tienneet kaikkia tietojärjestelmän toivottuun toimintaan vaikuttavia seikkoja. Mikäli tietojärjestelmän toiminta ei vastaa määrittelyä, on kyse ohjelmointivirheestä. Boehmin tutkimuksen mukaan valtaosa eli 56 prosenttia järjestelmän tuotantokäytössä esiin tulleista virheistä oli seurausta virheistä järjestelmän määrittelyn aikana. (Koistinen 2002, 87-89.)

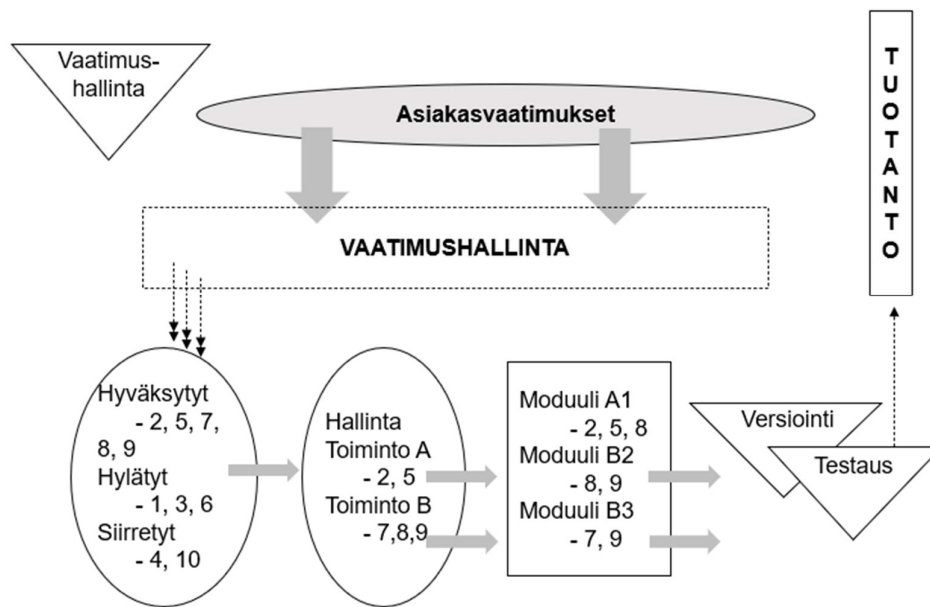
Riippumatta siitä, mistä tietojärjestelmään aiheutunut muutostarve johtuu, on tarpeen esiintuojalla tarve saada muutos tietojärjestelmään toteutettua mahdollisimman nopeasti. Näin on erityisesti silloin, jos tietojärjestelmä toimii virheellisesti. Muutoksissa ja virheiden korjauksissa on kuitenkin tärkeää tunnistaa, mistä virheellinen toiminta johtuu ja korjata sekä virheellinen toiminta että sen aiheuttamat seuraukset. Erityisesti mikäli

tietojärjestelmä on laaja, monimutkainen tai kytköksissä toisiin tietojärjestelmiin, on tärkeää tunnistaa virheellisen toiminnan kaikki vaikutukset ja huolehtia, ettei virheen korjaus luo uusia virheitä. (Koistinen 2002, 88.)

5.4 Muutostarpeiden hallinta ja analysointi

Ylläpidon organisoinnin yksi oleellisimmista asioista on tietojärjestelmään kohdistuvien muutostarpeiden analysointi, luokittelu ja tilastointi. Organisaation on oltava tietoinen, millaisia muutostarpeita sen tietojärjestelmiin kohdistuu ja millaisia muutoksia niihin tehdään. Muutostarpeita seuraamalla voidaan analysoida tietojärjestelmän toimintakuntoa ja analyysin perusteella pystytään tekemään tietoon perustuva päätös siitä, milloin tietojärjestelmä tulisi uusia osittain tai kokonaan. (Koistinen 2002, 20.) Muutostarpeita seuraamalla voidaan myös mitoittaa ja suunnitella ylläpitoon osoitetut resurssit paremmin (Koistinen 2002, 87). Tuotteesta kerätyn palautteen analysointi on yksi keino tuoda esiin käyttäjien kehitysideat sekä tilaisuus käydä keskustelua eri muutosten tärkeydestä (Hyysalo 2011, 136).

Kaikissa ylläpidon osa-alueiden organisoinnissa on suunniteltava, kuka hallinnoi ja käsittelee tietolähteistä kerättyä dataa ja miten. Tyypillisesti esiin noussut muutostarve ensin dokumentoidaan muutostarpeiden hallintajärjestelmään. Mikäli muutostarve vaatii toimittajan resursseja, toimittajalta pyydetään arvio muutoksen toteuttamisesta. Toimittajan arvion lisäksi on otettava huomioon muutoksen vaikutukset ja sillä saavutettavat hyödyt. Käsitteilyn ja kerättyjen lisätietojen perusteella voidaan tehdä päätös toteuttamisesta tietyssä foorumissa. Muutostarpeiden hallintajärjestelmässä ylläpidetään tietoa esiin tulleista muutostarpeista ja siitä, miten niitä on käsitelty sekä onko ne päätetty toteuttaa, hylätä tai jättää odottamaan. (Stenberg 2006, 115–116.) Kuviossa 6 on kuvattu yksi tapa ohjata, hallita ja toteuttaa sovelluksen muutostarpeita ja virhetilanteita.



Kuvio 6. Vaatimushallinnan organisointi (mukaillen Stenberg 2006, 116)

Muutostarpeiden hallinnan ja dokumentoinnin tehostamiseksi on hyödyllistä linjata, että muutostarpeet tulee esittää tai ne ainakin tallennetaan muutostarpeiden hallintajärjestelmään kirjallisena, vaikka muut keinot voivat tuntua vähemmän byrokraattisilta. Kirjallisista muutostarpeista voidaan myöhemmin tarkastaa, kuka muutosta on pyytänyt ja millä perustein. Muutostarpeeseen tulee kirjata ainakin, mihin sovellukseen muutostarve kohdistuu, millainen muutos halutaan tehtävän mahdollisimman tarkasti, mitä hyötyjä muutoksella saavutetaan, päivämäärä sekä muutoksen kiireellisyyttä. (Koistinen 2002, 130-131.)

Esiin tulleiden muutostarpeiden käsittelyä helpottamaan ne voidaan luokitella tärkeyden ja vaikutusten mukaisesti. Muutostarpeet voivat olla erityyppisiä virheitä tai niillä voidaan esimerkiksi pyrkiä lisäämään tietojärjestelmän käytettävyyttä tai vähentämään virheellisen käytön mahdollisuuksia. Muutosehdotuksia käsitellessä tulisi ottaa kantaa siihen, millaisia haitallisia vaikutuksia muutoksen toteuttamatta jättäminen aiheuttaa ja millaisia hyötyjä muutoksen toteutuksella voidaan saavuttaa. (Stenberg 2006, 117.) Mahdollisesti toteutettavan muutoksen työmäärää arvioitaessa ei voida arvioida ainoastaan teknistä ohjelmamuutoksen toteuttamista, vaan on otettava huomioon myös muut muutoksen vaikutukset eli esimerkiksi vaikutukset erilaisiin dokumentteihin, testaukseen sekä ylläpitovaiheessa olevan ohjelmiston käyttäjiin ja heidän koulutukseensa. Merkittävä osa kustannuksista, jotka muutostyö aiheuttaa, johtuvat muutostyön hallinnasta ja erilaisista sivuvaikutuksista. Siksi muutoksen ehdottajan näkökulmasta pieneltä vaikuttava muutos voi aiheuttaa yllättävän suuria kustannuksia. Muutoksen sivuvaikutusten kustannuksia ei

voi jättää huomiotta, koska niiden huomiointi voi vaikuttaa kokonaisarviointiin niin merkittävästi, ettei muutosta ryhdytä tekemään, koska analyysin perusteella muutoksen toteuttaminen aiheuttaa enemmän kustannuksia kuin säästöjä. (Lehtimäki 2006, 50.) Muutosta harkittaessa on huomioitava myös tietojärjestelmän liittymät muihin järjestelmiin ja vaikuttaako muutos myös niihin. Tämä voi viedä aikaa erityisesti, jos liittymiä on paljon. (Koistinen 2002, 174.) Käyttäjiltä kerätyn palautteen ja kehitysideoiden systemaattinen analysointi on keino osallistaa käyttäjät tietojärjestelmän kehittämiseen sekä tuottaa keskustelua eri muutosten hyödyistä (Hyysalo 2011, 136).

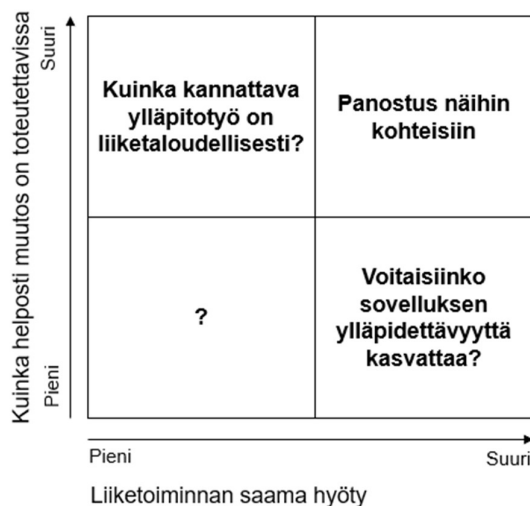
Myös muutostarpeet, jotka kuluvalle hetkellä syystä tai toisesta hylätään, on hyödyllistä tallentaa sekä dokumentoida niihin jo kootut tiedot eli esimerkiksi hyödyt, vaikutukset ja työmääräarviot. Vaikka muutostarvetta ei kuluvalle hetkenä voidakaan toteuttaa, voidaan se tallentaa kehitysideana ja toteuttaa jonain muuna sopivampana hetkenä, jolloin sen työmääräarvio voi mahdollisesti olla pienempi tai hyödyt suurempia. (Lehtimäki 2006, 51.) Yhtä tärkeää kuin on pyrkiä keräämään käyttäjien näkemyksiä ja palautetta, on palautteen säännöllinen ja säännönmukainen käsittely. Palautteen kerääminen on vain kuluera, mikäli sitä ei käytetä hyväksi päätöksenteossa ja kehittämisessä. (Merisalo-Rantanen 2011, 16.)

Yksi muutostenhallinnan keino on projektointi. Projektoinnilla tarkoitetaan sitä, että kaikki tietojärjestelmään kohdistuneet muutostarpeet kerätään yhteen tietyin väliajoin, esimerkiksi kaksi kertaa vuodessa. Organisaation tietojärjestelmämuutoksista päättävä ryhmä arvioi muutostarpeet ja päättää, mitkä toteutetaan seuraavaksi eli otetaan projektiin mukaan. Päätöksiin voivat vaikuttaa esimerkiksi muutostarpeen kiireellisyys tai synergiaetujen saavuttaminen toteuttamalla tietyt muutokset samanaikaisesti. Muutostarpeet, joita ei valita mukaan, jäävät odottamaan seuraavaa projektia. (Koistinen 2002, 97.)

Ylläpidon tehokkuutta parantaa projektoinnin lisäksi tehtävien muutosten versiointi, mikä tarkoittaa, että tietojärjestelmästä otetaan säännöllisin väliajoin käyttöön uusi versio, joka sisältää muutoksia ja uusia toiminnallisuuksia tietojärjestelmään. Projektointi ja versiointi tekevät ylläpidosta hallitumpaa ja selkeyttävät aikatauluja. (Koistinen 2002, 97.) Ylläpidon suunnittelussa on sovittava, kuinka usein uusi versio otetaan käyttöön ja kuinka isoja uudet versiot ovat eli kuinka paljon niihin otetaan uusia asioita mukaan kerrallaan. Liiketoiminnan ja tietojärjestelmän kehittämisen edustajat päättävät yhteistyössä, mitä muutoksia tietojärjestelmään toteutetaan ja missä versiossa. Versiointi parantaa yhteistä näkemystä tietojärjestelmän kehittämisen seuraavista toimenpiteistä ja edesauttaa tietojärjestelmän suunnitelmallista ja systemaattista kehitystä. Mikäli ylläpito on versioitu,

tulisi ylläpidossa välttää muutoksien vientiä tuotantoon versioiden välissä, jotta versioinnin tuottamia hallinnan etuja ei menetetä. Tietyissä tilanteissa muutokset on kuitenkin vietävä tuotantoon mahdollisimman nopeasti ja silloinkin muutosten vienti on oltava yhdessä harkittu ja sovittu sekä huolehdittava muutoksen dokumentoinnista. (Koistinen 2002, 123-125.)

Yksi osa muutostarpeiden hallintaa on päätösprosessi koskien sitä, mitä muutoksia tietojärjestelmään lopulta toteutetaan ja missä järjestyksessä. Yhtenä apukeinona päätöksenteossa voidaan käyttää kuviossa 7 esitettyä nelikenttää, jonka avulla voidaan arvioida muutosten toteutuskannattavuutta. Pystyakselilla oleva muutoksen helppous tarkoittaa sitä, kuinka paljon resursseja muutoksen toteuttaminen vaatii eli kuinka ylläpidettävä sovellus on. Vaaka-akselille on asetettu liiketoiminnan saama hyöty. Nelikentän vasempaan alakulmaan asetettujen muutosten toteutus vaatii paljon resursseja ja niiden tuottama hyöty on pieni, joten niiden toteuttaminen ei ole tehokasta. Poikkeuksen muodostavat esimerkiksi lain muutosten aiheuttamat muutostarpeet. Alaoikealle ja ylävasempaan asetetut muutokset voi olla tehokasta toteuttaa niiden vaatimien pienten resurssien tai suurten hyötyjen vuoksi. Mikäli merkittävä osa muutostarpeista asettuu alaoikealla olevaan neliöön, on aiheellista pohtia, onko sovelluksen ylläpidettävyyttä kasvattaa vai pitäisikö sitä kasvattaa. Yläoikealla olevan neliön muutosten toteuttaminen on kaikkein kannattavinta. (Koistinen 2002, 173.)



Kuvio 7. Ylläpitötöiden valinta (mukaillen Koistinen 2002, 173)

5.5 Dokumentaation merkitys ylläpidossa

Riittävän tarkan ja ajantasaisen dokumentaation merkitys korostuu entisestään tietojärjestelmän ylläpitovaiheessa. Kun ylläpitäjälle annetaan toimeksianto tehdä muutos

tietojärjestelmään ilman tarpeeksi kattavaa dokumentaatiota, on riskinä, ettei ylläpitäjä pysty suunnittelemaan, mihin järjestelmän kohtaan muutos olisi kohdistettava tai tiedä, mihin kaikkeen muutos vaikuttaa. Muutos voidaan pahimmassa tapauksessa tehdä väärään kohtaan ja näin aiheuttaa odottamattomia ongelmia. Erityisesti jos kyseessä on monimutkainen tietojärjestelmä, jossa on paljon liitoksia eri toiminnallisuuksien välillä, on muutokset harkittava tarkasti ja huolehdittava, että kaikki muutoksen vaikutukset on otettu huomioon. (Koistinen 2002, 54.) Kaikkien ratkaisujen dokumentointi on tärkeää, mutta erityisen tärkeää on dokumentoida poikkeavat ratkaisut perusteluineen. Kun ratkaisut on dokumentoitu laadukkaasti, ei ylläpitäjän aikaa kulu asian selvittämiseen. Perustelemalla poikkeava ratkaisu voidaan välttyä myös turhilta muutoksilta. (Koistinen 2002, 177.)

Useimmiten syynä puutteelliselle dokumentaatiolle on ajanpuute, mikä voi johtua siitä, ettei dokumentoinnille varata aikaa tai sitä sisällytetä työmääriin. Dokumentoinnille tulisikin laatia standardit, joilla huolehditaan dokumentoinnin toteutumisesta ja dokumentoinnin tasosta niin, ettei dokumentin tarkkuus ole riippuvainen sen tekijästä. Riittävän tarkka ja laaja dokumentaatio auttaa myös varautumaan etukäteen henkilövaihdoiksiin ja niissä olemassa olevaan riskiin oleellisen tiedon katoamisesta organisaatiosta lähtevien henkilöiden mukana. (Koistinen 2002, 53-54.)

Yksittäisen asiantuntijan mukaan jokin tietojärjestelmän toiminnallisuus tai tekninen yksityiskohta voi olla suunniteltu tai toteutettu väärin. Tällaisessa tilanteessa voi olla houkuttelevaa korjata virhe ilman byrokratiaa ja monimutkaiselta tuntuvaa muutoksenhaallintamenettelyä. Se, mikä on oikein tai väärin, ei kuitenkaan ole itsestäänselvyys ja se on aina riippuvainen asiantuntijan tulkinnasta. Se, miten asiat tulee toteuttaa, on sovittava yhteisesti, jotta muodostuu yhteinen käsitys siitä, mikä on oikein. Muutosten tekeminen ilman asianmukaista muutoksenhaallintamenettelyä ja siihen liittyvää dokumentaatiota hankaloittaa kokonaisuuden hallintaa ja kokonaiskuvan säilyttämistä tietojärjestelmästä. (Lehtimäki 2006, 52-53.) Laadukasta dokumentointia on jatkettava ja dokumenttien ajantasaisuudesta huolehdittava myös ylläpitovaiheessa. Yhtenä ratkaisuna dokumenttien ylläpidon varmistamiseen on dokumentoida suunniteltu työ ennen toteutuksen aloittamista. (Koistinen 2002, 127.)

Tietojärjestelmän pienimmälläkin muutoksella ylläpitovaiheessa on suora vaikutus tietojärjestelmän käyttäjiin ja siellä käsiteltävään tietoon. Jotta muutostarpeiden analysointi ja päätöksenteko toteutettavista muutoksista perustuisi oikeaan tietoon, on tärkeää, että tietojärjestelmän dokumentaatio on ajan tasalla. Jos muutoksen analysoinnin apuna käytetään päivittämätöntä dokumentaatiota, voidaan tietojärjestelmään toteuttaa muutoksia, joilla on hallitsemattomia muutoksia tietojärjestelmän muihin osiin. Myös

dokumentaation perusta luodaan jo ylläpitoa edeltävissä vaiheissa. Kattava dokumentaatio koko kehittämisen ajalta varmistaa sen, että muutostarpeen kriittisyyttä arvioivilla henkilöillä on käytössään riittävät tiedot muutoksen vaikutuksista. Muutosten dokumentointi tulee olla kirjattuna ylläpidon muutoksenhallintaprosessiin, jotta dokumentaation ylläpito jatkuu myös ylläpitovaiheessa.

5.6 Käyttäjakeskeisyys tietojärjestelmän ylläpitovaiheessa

Käyttäjakeskeisessä kehittämisessä käyttäjän näkökulma huomioidaan kaikissa tietojärjestelmän kehittämisen vaiheissa. Käyttäjien tarpeita ja palautetta kannattaa selvittää mahdollisimman aikaisessa vaiheessa tietojärjestelmän kehittämistä, jotta palautteeseen ehditään reagoida. Käyttäjäpalautetta kannattaa kerätä kuitenkin myös valmiista tuotteesta, jolloin palautetta voidaan käyttää tuotteen seuraavaa versiota suunnitellessa (Jokela 2010, 50). Vaikka tietojärjestelmä olisi testattu laajasti kuvitteellisilla tilanteilla ja siitä olisi saatu palautetta, on tuotantokäytössä olevasta tietojärjestelmästä saatu palaute arvokkaampaa ja jäsennellympää. Kyselemällä ja haastattelemalla käyttäjiä voidaan kartoittaa käytössä olevan tietojärjestelmän vastaavuus työtehtäviin nähden ja pystytään keräämään muutos- ja kehittämistarpeita (Stenberg 2006, 91). Organisaation tietojärjestelmän ylläpitostrategiassa on otettava huomioon käyttäjien mielipiteet ylläpidon kehittämisestä (Koistinen 2002, 69).

Teknologia on keino saavuttaa inhimillisen toiminnan päämäärät (Saariluoma 2011, 59), joten palaute teknologiasta on aina myös suhteessa toimintaan, jota teknologialla yritettiin toteuttaa. Tietojärjestelmän käytön perustana on aina inhimillinen toiminta.

Testaustilanteessa toiminnan päämääränä on testin loppuunsaattaminen, joten palaute voi koskea sellaista tietojärjestelmän toimintaa, jota käyttäjä käyttäisi aivan eri tavalla, mikäli toiminnan päämääränä ja teknologian käytön perustana olisi jokin todellinen työtehtävä. Käyttäjä testaa tietojärjestelmää kuvitteellisella tilanteella ja käyttötapauksella ja voi nopeasti tietojärjestelmää testatessaan esimerkiksi unohtaa kirjata jonkin tiedon, joka oikeassa työtilanteessa ei todennäköisesti jäisi kirjaamatta. Tällöin käyttäjä voi ehdottaa tietojärjestelmään toteutettavaksi jonkinlaista muistutusta tai herätettä asian kirjaamiseksi, vaikka tuotantokäytössä asian kirjaaminen unohtuu paljon pienemmällä todennäköisyydellä. Testitilanteessa tietojärjestelmässä myös viedään muutamissa tunneissa läpi tilanteita, jotka tosielämässä tapahtuvat useamman päivän tai viikkojen aikavälillä, jolloin käyttäjälle tulee tunne työläästä ja monenlaisia kirjauksia vaativasta tietojärjestelmästä. Testitilanteen taustalla oleva inhimillinen toiminta on siis keinoitekoinen.

Mikään tietojärjestelmä ei ole täydellinen. Harva on käyttöönottohetkellään edes hyvä. Myös hyvää tietojärjestelmää joudutaan kehittämään ja räätälöimään vähintään muuttuneiden tilanteiden ja uusien tarvevaatimusten takia (Hyysalo 2011, 133). Prototyyppien testaaminen voi myös olla haastavaa, jos testattava tuote on monimutkainen. Kun ”kaikki on mahdollista”, käyttäjä ei usein tiedä, mitä hän haluaa (Ries 2011, 118). Jos käyttäjä ei ymmärrä tai osaa hahmottaa, miten prototyyppi toimisi, jos sitä käytettäisiin tosielämässä, ei palautteen antaminenkaan ole mahdollista. Kun käyttäjän on pakko ryhtyä käyttämään uutta tietojärjestelmää, voi hänen olla helpompaa antaa siihen myös kehitysehdotuksia. Vasta teknologian todellinen käyttäminen synnyttää käyttäjyyttä ja tuo esiin teknologian ominaisuudet. Tietojärjestelmää koskevat yksityiskohtaiset tarpeet, sen mahdollisuudet ja puutteet ja tosiasiallinen käyttö kaikkine käytänteineen tulee näkyväksi ja koettavaksi vasta kun sitä käytetään arjen työkaluna todellisissa tilanteissa. (Hyysalo 2011, 133.)

Organisaation työntekijöillä on valtavasti osaamista ja tietoa kaikesta organisaation osa-alueilla tehtävästä työstä ja peilaten omaan tietämykseensä he huomaavat myös paljon käytännön kehittämistarpeita. On tärkeää kannustaa ja motivoida työntekijöitä tuomaan ideansa ja kehittämistarpeensa esiin, jotta heidän tietämystään ja ideoitaan pystytään hyödyntämään koko organisaation hyväksi. Tämä koskee myös organisaation käytössä olevien tietojärjestelmien kehittämistarpeita. Jatkuvassa kehittämisessä on olennaisen tärkeää, että kaikki saavat mahdollisuuden esittää ideansa ja viedä aloitteita eteenpäin. Ideoiden ja kehittämistarpeiden käsittelyä tulee myös seurata järjestelmällisesti ja säännöllisesti. Ideoiden elinkaari voi olla läpinäkyvä ja kaikkien työntekijöiden nähtävissä. On kannattavaa luokitella, seurata ja tilastoida esitettyjä ideoita ja miten niitä on käsitelty, jotta ideointijärjestelmän tilastojen perusteella voidaan ryhtyä toimenpiteisiin. (Stenberg 2006, 53.)

Tietojärjestelmän ylläpitoa tarvitaan pääasiassa siksi, että tietojärjestelmän tuotantokäytön alkaessa käyttäjät ovat alkaneet käyttää tietojärjestelmää. Monien ulkoisten muutostarpeiden aiheuttajien lisäksi käyttäjät aiheuttavat muutostarpeita suunnitellusti ja suunnittelemattomasti. Käyttäjien tarpeet voivat poiketa siitä, millaisia tietojärjestelmän suunnittelijat ajattelivat niiden olevan tai käyttäjien tehtävät voivat muuttua ja tietojärjestelmän on vastattava muuttuneisiin tilanteisiin. (Koistinen 2002, 35.) Tästä syystä käyttäjät on otettava ylläpitovaiheessa entistä tiiviimmin mukaan tietojärjestelmän ylläpitoon ja jatkokehittämiseen.

Sen lisäksi, että käyttäjien näkemyksiä selvitetään ja heidät otetaan mukaan tietojärjestelmän kehittämiseen sen ylläpitovaiheessa, tulisi käyttäjäkeskeisyyden toteutua

myös varmistamalla, että ylläpidosta vastaavat henkilöt tuntevat ylläpitämänsä tietojärjestelmän tarpeeksi hyvin. Ylläpidosta vastaavien tulisi tuntea tietojärjestelmän sovelluksen toiminnan lisäksi sen toimintaympäristö, periaatteet, tavoitteet ja prosessit, jotta he voivat paremmin ymmärtää myös käyttäjien tarpeita. Tällaisen käyttäjätiedon kerääminen helpottaa yhteisen kielen löytymistä ylläpitäjän ja käyttäjän välillä, parantaa ylläpidon osuvuutta ja kasvattaa ylläpidon tuottavuutta. (Koistinen 2002, 96.)

5.7 Tietolähteet

Valtaosa eri yhtiöiden tuottamien teknologioiden, esimerkiksi tietokoneiden, käyttöliittymät muistuttavat peruseriaatteiltaan toisiaan. Tähän on syynä osin se, ettei pyörää kannata keksiä uudelleen, mutta osin myös pyrkimys madaltaa ihmisten kynnystä vaihtaa tuttua käyttöjärjestelmää tai ohjelmistoa uuteen. Ihmisen oppimisen pohjana on aina aiemmin opitut asiat ja sen vuoksi käyttäjät voivat olla vastahakoisia kokeilemaan täysin uudenlaista teknologiaa. Oppimisen perustamista aiemmin opituille asioille kutsutaan opitun siirtovaikutukseksi ja sen vuoksi uuden toiminnan oppiminen on helpompaa, jos toiminta muistuttaa aiemmin tuttua toimintamallia. Uuden teknologian käytön opiskelu on käyttäjälle sitä helpompaa, mitä enemmän käyttöliittymää suunnitellessa on pohjana pystytty käyttämään aiemmin käytössä olleita rakenteita esimerkiksi toimintoketjujen osalta. (Saariluoma ym. 2010, 61–62.)

Uusi teknologia voi tuntua käyttäjistä vaikealta. Etenkin jos teknologian käytettävyyks on alhainen, se voi saada käyttäjän kokemaan, että käyttämisen vaikeus on heidän oma syynsä ja he tuntevat olonsa avuttomiksi. Tällaista tunneilmiötä kutsutaan teknostressiksi. Teknostressin takia käyttäjät eivät myöskään pysty näkemään uuden teknologian hyötyjä tai heidän vaikutusmahdollisuuksiaan siihen. (Saariluoma ym. 2010, 67–68.)

Käyttäjäpalautteen aktiivinen kerääminen heti tietojärjestelmän käyttöönoton jälkeen ei ole tehokkain ajankohta, kun tavoitteena on saada rakentavaa ja jäsenneltyä palautetta. Käyttäjien kyky nähdä ja sanoittaa konkreettisia muutoskohteita tietojärjestelmässä voi olla alhainen, kun he eivät vielä tiedä, miten sen on tarkoitus edes nyt toimia.

Käyttäjät kuitenkin oppivat tietojärjestelmän käyttöä koko ajan, joten käyttäjäpalautteen systemaattinen kerääminen tulisi aloittaa kohtalaisen pian käyttöönoton jälkeen. Tietojärjestelmän käyttö ei ole työn arjesta irrallaan oleva työtehtävä, vaan uusi teknologia asettuu omalle paikalleen osaksi työn tekemistä ja organisaation käytänteisiin. Todelliset hyödyt ja ongelmat tulevat näkyväksi vasta tässä kohdin. Kun käyttäjät ovat käyneet käyttökoulutuksen, he useimmiten jonkin aikaa sen jälkeen käyttävät tietojärjestelmää siten, miten sitä oletettiin käytettävän, mutta tekevät sen jälkeen lukuisia muutoksia käyttötapoihinsa. Tutkimusten mukaan kuuden kuukauden kuluttua käyttöönotosta

teknologian käyttötapoja tai ominaisuuksia ei enää kyseenalaisteta. Ihmiset siis ottavat teknologian haltuun niin, että se sopii heidän tapoihinsa. Tiedon kerääminen käytöstä tulee aloittaa pian käytön aloittamisen jälkeen ja sitä on jatkettava systemaattisesti niin kauan, kuin tuote on käytössä, koska tuotteen kehittämisen ja käytön yhteensovittamisella voidaan saavuttaa merkittäviä hyötyjä. (Hyysalo 2011, 145–147.)

Tietojärjestelmän ja sen käyttöliittymän toiminnallisuudelle tulee sokeaksi, kun sen kanssa tekee tarpeeksi kauan töitä (Saariluoma ym. 2010, 227). Jokaisen tietojärjestelmän käyttö on ajan kanssa mahdollista oppia, mutta otollinen kohta kehitysehdotusten keräämiseen on ajankohta, jolloin käyttäjät ovat oppineet sen käytön periaatteet, mutta eivät vielä tottuneet siihen niin, että eivät kykene tai halua nähdä siinä muutoskohteita. Kun käyttäjäpalautetta ryhdytään keräämään tarpeeksi nopeasti tietojärjestelmän käyttöönoton jälkeen ja sen tärkeydestä viestitään käyttäjille, luodaan käyttäjille alusta alkaen sellainen käyttämisen mielentila, että he pystyvät näkemään tietojärjestelmässä kehittämisen kohteita ja ovat motivoituneita olemaan mukana sen kehittämisessä. Kun käyttäjillä on motivaatio kehittämiseen, se lykkää heidän sokeutumistansa tietojärjestelmän käytölle ja sopeutumista sen ongelmiin.

Kun käyttäjät ovat käyttäneet tietojärjestelmää jonkin aikaa, voidaan sen käytettävyyttä selvittää kyselyllä. Kyselyllä voidaan selvittää käyttöön liittyviä tekijöitä, tietojärjestelmän vastaavuutta tehtäviin, joita sillä pyritään tekemään ja näiden mahdollisia ristiriitoja sekä tietojärjestelmän kehittämistarpeita. Tietojärjestelmän käytettävyyttä tulisi tutkia systemaattisesti ja jatkuvasti, koska sekä tehtävät että myös tietojärjestelmän käyttö muuttuu ajan kuluessa. (Stenberg 2006, 52.) Tietojärjestelmän kehittämisen ja ylläpidon yleisen tason pohdintaa varten voidaan perustaa myös käyttäjäryhmiä- tai kerhoja (Koistinen 2002, 203).

Tietojärjestelmän kehittämistarpeita voidaan selvittää myös tekemällä johtopäätöksiä tukiprosessien toiminnasta. Organisaatioilla on omat toimintamallinsa tilanteisiin, joissa käyttäjät tarvitsevat jonkinlaista tukea tietojärjestelmän käytössä. Tuen tarve voi liittyä käyttäjän omaan virheelliseen toimintaan, tietojärjestelmän virheelliseen toimintaan tai siihen, ettei käyttäjä tiedä, miten käyttää tietojärjestelmää. Tilastoimalla ja luokittelemalla tukipyynnöitä ja sitä, mikä on aiheuttanut tukipyynnön, voidaan tehdä tarpeellisia johtopäätöksiä tietojärjestelmän kehittämiskohdista ja muutostarpeista. Tukipyynnötilastoja tulkitsemalla voidaan tehdä päätelmiä myös käyttäjien kouluttamisen tarpeista. (Stenberg 2006, 52.)

5.8 Ylläpidettävyystekijät

Kaikilla tietojärjestelmillä on ylläpitovaihe. Joidenkin tietojärjestelmien ylläpitovaihe voi olla niin lyhyt, ettei sen ylläpidettävyyteen ole kannattavaa panostaa paljon resursseja, mutta mitä pidempi ylläpitovaihe on, sitä enemmän sen ylläpidettävyyteen kannattaa panostaa. Tietojärjestelmän sisältämästä koodista muutetaan vuosittain jollain tasolla 10-15 prosenttia, joten etenkin laajojen tietojärjestelmien kohdalla on erityisen tärkeää, että ylläpitotehtävät voidaan toteuttaa helposti ja virheettömästi. Ylläpidettävyyden tasoa arvioidaan yleisesti sen perusteella, kuinka helppo järjestelmän ominaisuuksia on lisätä, poistaa tai muokata käyttäjien tarpeiden mukaisesti. (Koistinen 2002, 38.)

Vaikka tietojärjestelmän ylläpito voi tietojärjestelmää kehitettäessä tuntua kaukaiselta vaiheelta, ei ylläpidon suunnitteluun voi havahtua vasta, kun se on lähellä. Tietojärjestelmää kehitettäessä on lyhytnäköistä keskittyä ainoastaan kesken olevaan vaiheeseen, jos on tiedossa, että tietojärjestelmän kallein ja pisin vaihe, ylläpitovaihe, on tulossa. (Koistinen 2002, 31.) Usein tietojärjestelmän kehittämisen haasteena on kiire, tiukat resurssit ja tarve saada tietojärjestelmä tuotantokäyttöön. Kun suunnittelijoiden fokus on tietojärjestelmän käyttöönotossa, ei sen suunnittelussa välttämättä ole aikaa tehdä ratkaisuja, joissa on otettu huomioon myös tietojärjestelmän ylläpito. Se, ettei tietojärjestelmän suunnittelussa huomioida tarpeeksi sen ylläpidettävyyttä, konkretisoituu ylläpitovaiheessa ylläpidon haasteina. Pahimmillaan ylläpidossa joudutaan tekemään isojaakin korjauksia ja muutoksia tietojärjestelmän ratkaisujen laadun parantamiseksi. (Koistinen 2002, 37.) Ylläpidettävyystekijät on otettava huomioon ja huolehdittava tietojärjestelmän ylläpidettävyydestä myös ylläpitotöissä eli ylläpitotehtäviäkin tehdessä on nopeusvaatimuksista ja kiireestä huolimatta varmistettava, etteivät tehdyt ratkaisut ja muutokset sovellukseen vaikeuta tulevia ylläpitotöitä (Koistinen 2002, 94).

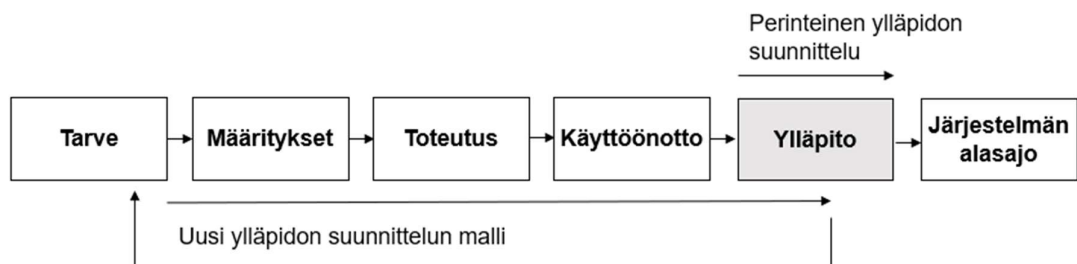
Tietojärjestelmän ylläpitovaihe ja ylläpidettävyyds tuleekin ottaa huomioon jo tietojärjestelmän suunnittelu- ja toteutusvaiheissa, eikä vasta käyttöönoton jälkeen tai sen lähestyessä (kuvio 8). Käytännössä tietojärjestelmän suunnittelun aikana tulisi arvioida, kuinka todennäköisesti tietojärjestelmään suunnitellut asiat tulevat tulevaisuudessa muuttumaan ja suunnitella niiden toteutus sen perusteella. Mikäli toteutettavaan asiaan todennäköisesti kohdistuu tulevaisuudessa muutostarpeita, on tehokasta toteuttaa asia järjestelmään niin, että muutokset on helppo tehdä. (Koistinen 2002, 37.)

Tietojärjestelmän hyvällä suunnittelulla voidaan välttyä turhilta ylläpitotöiltä (Koistinen 2002, 55). Tietojärjestelmän suunnitteluvaiheessa kannattaa myös huomioida, että ylläpidon aikana joudutaan palaamaan aikaisempiin vaiheisiin, koska korjaavan ja kehittävän ylläpidon tehtävien vuoksi tarpeita, määrittämiä ja toteutuksia voidaan joutua

muuttamaan. Useimmiten tietojärjestelmän kehittämistä tapahtuu ylläpidon kanssa rinnakkain, joten ylläpidon aikanaikin toteutetaan aikaisempia vaiheita uusien toiminnallisuuksien osalta (kuvio 8).

Tuleviin muutoksiin voidaan varautua esimerkiksi parametroimalla muutoksen kohteena oleva arvo tai toiminnallisuus ohjelmakoodin sijasta käyttäjätauluun tai moduloimalla sovellus (Stenberg 2006, 116). Tietojärjestelmän ylläpidettävyyttä voidaan kasvattaa myös noudattamalla sovelluksen rakenteen suunnittelussa ja toteutuksessa kurinalaisuutta. Ohjelmoinnissa avainasemassa on järjestelmällisyys ja ns. ”spagetti-koodin” välttäminen. Ohjelmakoodia tuottaessa tulisi koko ajan huolehtia, että joku muukin kuin koodin alkuperäinen tuottaja pystyy ymmärtämään, mitä koodissa tapahtuu. Tästä voidaan huolehtia koodin tarpeeksi tarkalla dokumentoinnilla, noudattamalla kuvaavia ja yhdenmukaisia nimeämiskäytäntöjä ja tekemällä koodista luettavaa. (Pohjonen 2002, 35.) IBM:n tutkimuksen mukaan yksittäiseen ylläpitotehtävään kulutetusta ajasta jopa 50 prosenttia kuluu koodin ymmärtämiseen eli siihen, kun ylläpitäjä etsii sovelluksesta kohdan, johon muutos tulisi tehdä (Koistinen 2002, 43).

Vaikka suunnittelu- ja toteutusvaiheiden kiire voi aiheuttaa sen, että ylläpidettävyyssasiat helposti ohitetaan turhana resurssien käyttönä, saadaan ylläpitoasioiden huomioimiseen käytetty panostus moninkertaisena takaisin tehokkaana ylläpitoa. Ylläpitotehtävät voidaan tehdä virheettömämmin ja muutostarpeet saadaan käyttäjien käyttöön nopeammin. (Koistinen 2002, 37.) Kun suunnittelijat arvioivat tulevia muutostarpeita, heidän tulee aktiivisesti tiedustella organisaatiolta mahdollisista tulevista muutoksista. Organisaatiolla saattaa olla tiedossaan muutoksia, joita ei ole vielä julkistettu, mutta joihin tulisi varautua. (Koistinen 2002, 83.) Mahdollisia muutoksia kannattaa kartoittaa avoimesti ja laajasti, koska henkilö, joka suunnittelee tulevaa muutosta, ei välttämättä hahmota tai ymmärrä muutoksen vaikutusta tietojärjestelmän toimintaan eikä siksi osaa tuoda tulevaa muutosta oma-aloitteisesti tietojärjestelmän suunnittelijoiden tietoon.



Kuvio 8. Ylläpidon uusi malli (mukaillen Koistinen 2002, 37)

5.9 Tietojärjestelmän laatu

Tietojärjestelmän laatua on arvioitava ja mitattava ylläpitovaiheessa. Ylläpitotoiminnalle on laadittava konkreettiset mittarit, joiden perusteella ylläpidon onnistumista arvioidaan. Mittareita laadittaessa on huomioitava myös käyttäjien toimintaa kuvaavat mittarit. (Koistinen 2002, 75.) Laatua voidaan tarkastella tilaajan, toimittajan, ylläpitäjän sekä loppukäyttäjän näkökulmista ja siinä on kyse odotusten täyttymisestä. Jo tietojärjestelmän määrittely- ja suunnitteluvaiheissa tulisi huomioida tietojärjestelmän laatu, joka koostuu mukautumiskyvystä, muutosvalmiudesta, käyttöominaisuuksista sekä toiminnallisesta ja tiedollisesta kattavuudesta. (Stenberg 2006, 90-91.)

Yleisimpiä laatutekijöitä ovat virheettömyys, luotettavuus, tehokkuus, eheys, käytettävyys, ylläpidettävyys, testattavuus, joustavuus, siirrettävyys, uudelleenkäytettävyys ja yhteensopivuus. Laatutekijät voivat olla päällekkäisiä tai sisältyä toisiinsa. Ne ovat myös riippuvaisia toisistaan. Esimerkiksi ylläpidettävyyttä tukevat virheettömyys, luotettavuus sekä käytettävyys. Toisinaan laatutekijöiden välillä joudutaan tekemään myös kompromisseja. (Harsu 2003, 52-54.) Laadun käsitettä on kuitenkin vaikea määritellä, mikä tekee sen arvioinnista haastavaa. Perinteisesti laatu on merkinnyt sitä, että järjestelmä on vaatimusmäärittelyn mukainen, mutta vaatimusmäärittelyitä on vaikea kirjoittaa kattavasti, joten vaikka järjestelmä toteutettaisiin vaatimusmäärittelyjen mukaisesti, käyttäjät eivät silti välttämättä pidä sitä laadukkaana. (Harsu 2003, 61.)

Ylläpidon laatua voidaan arvioida ja mitata esimerkiksi muutosten luokittelun avulla sekä seuraamalla hyväksytyjen, hylättyjen, siirrettyjen ja virheellisten toimitusten määriä ja kustannuksia. Laadun arvioinnissa voidaan ottaa huomioon myös muutosten välttämättömyyden ja kiireellisyyden tila. Toimittajan sovellustoimitusten laatua on syytä arvioida myös omana kokonaisuutenaan. Sovellustoimitusten laatuun on laadittava yhteisesti sovitut ja jatkuvat mittarit, joilla voidaan seurata muun muassa toimitusaikoja, virheiden määrää ja asiakaspalvelun laatua. (Stenberg 2006, 116-117.)

6 Kehittämistyön toteutus

Opinnäytetyön tutkimuksen vaiheet olivat aiheeseen perehtyminen, suunnittelu, toteutus ja raportointi. Tutkimusprosessin aikana toteutettiin myös tutkimuksellinen kehittämistyö.

6.1 Aiheeseen perehtyminen

Opinnäytetyön tekijä on osallistunut kohdetietojärjestelmän suunnitteluun ja opinnäytetyöprosessi käynnistyi opinnäytetyön tekijän havainnosta käyttäjäkeskeisyyden haasteellisuudesta. Tietojärjestelmäprojektin aikataulu- ja resurssihaasteista johtuen ei tietojärjestelmän suunnittelussa pystytty osallistamaan sen tulevia käyttäjiä tietojärjestelmän suunnitteluun niin laajasti, kuin olisi ollut tarpeellista. Tulevia käyttäjiä oli niin paljon ja heidän teknologiset taitonsa ja käyttötapansa niin erilaisia, että suunnittelussa ei ollut mahdollista ottaa kaikkien tarpeita huomioon. Lisäksi käyttäjien toimintatavat vaihtelivat yksiköittäin. Yksittäisiltä käyttäjiltä hankittiin tietoa nykytilan toiminnasta ja tarpeista koskien tietojärjestelmän toimintaa, mutta oli selkeää, ettei käyttäjien otanta ollut tarpeeksi laaja koko käyttäjäkuntaan suhteutettuna.

Opinnäytetyön tekijän ja kohdeorganisaation edustajan välisissä keskusteluissa todettiin, että käyttäjien osallisuutta kehittämiseen tulisi lisätä tietojärjestelmän ylläpitovaiheessa ja organisoida kehittävä ylläpito niin, että sen lähtökohta on käyttäjäkeskeisyydessä. Kohdeorganisaation edustaja toi lisäksi esiin, että prosessikuvauksessa olisi tarve huomioida esiin tulleiden muutostarpeiden analysointi niiden priorisoinnin helpottamiseksi. Priorisoinnin tulisi perustua aiempaa rakenteellisempaan käsittelyyn eikä yksittäisten henkilöiden tietotaitoon ja tarveharkintaan.

Opinnäytetyön tutkimuksellinen kehittämistyö aloitettiin aihepiiriin tutustumisella ja tietoperustan koostamisella, jonka avulla muodostettiin alustavat tutkimuskysymykset. Aiheeseen liittyvään kirjallisuuteen perehtymisen perusteella käyttäjäkeskeisyyden lisäämisellä on useimmiten positiivinen vaikutus tuotteen käytettävyyteen. Käyttäjien osallistamisesta tuotteen kehittämiseen on sitä enemmän hyötyä, mitä aiemmin heidät otetaan mukaan tuotteen suunnitteluun. Tehtäessä päätöksiä käyttäjien osallistumisen määrästä ja laajuudesta on kuitenkin pyrittävä siihen, että siihen käytettävät resurssit ovat järkevässä suhteessa sillä saavutettaviin hyötyihin. Asiakastietojärjestelmän kehittävä ylläpidon prosessikuvaus suunniteltiin pohjautuen sekä kirjallisuuteen että käyttäjien näkemyksiin.

6.2 Suunnittelu

Kehittämistyön lähestymistavaksi valittiin konstrukttiivinen tutkimus, koska kehittämistyönä oli suunnitella kehittävän ylläpidon prosessikuvaus, jonka mukaisesti kehittävä ylläpito organisoitaisiin. Konstrukttiivisessa tutkimuksessa uuden rakenteen luomiseen käytetään teoreettista ja empiiristä tietoa. Kehittämistyön toteuttamiseen sovellettiin Järvisen ja Järvisen innovaation toteuttamisprosessia, joka koostuu lähtötilasta, toteuttamisesta ja tavoitetilasta.

Alustavia tutkimuskysymyksiä tarkennettiin lopulliseen muotoon useamman iteraatiokierroksen kautta. Ensimmäisten versioiden tutkimuskysymykset keskittyivät tietojärjestelmän käytön haasteisiin ja siihen, millä keinoin käyttäjien näkemykset selvitetään. Myöhemmissä versioissa tutkimuskysymykset liittyivät käyttäjäkeskeisyyden haasteisiin ennen tietojärjestelmän ylläpitovaihetta, kehittävän ylläpidon organisoimiseen sekä käytettävyyden kasvattamiseen. Lopulliset tutkimuskysymykset eivät käsitelleet tietojärjestelmän käytössä koettuja haasteita, koska ne eivät opinnäytetyöntekijän näkemyksen mukaan lopulta olleet käyttäjäkeskeisyyden toteuttamisen kannalta oleellista. Tutkimuksen tavoitteiden ja tulosten yleistettävyyden kannalta oleellisempaa oli, miten tietojärjestelmän käytössä koetut haasteet saadaan jäsennellysti ylläpito-organisaation tietoon, jotta niiden perusteella tietojärjestelmää voidaan jatkokehittää eikä niinkään se, mitkä asiat oikeastaan tuottivat käytössä haasteita. Tietojärjestelmän käytössä koetut haasteet eivät ole pysyviä eli yhdessä hetkessä selvitettyjen haasteiden ajantasaisuus ja todenmukaisuus vähenee melko nopeasti etenkin, kun haasteita olisi selvitetty ennen tietojärjestelmän käyttöönottoa. Tutkimalla sitä, miten haasteet saadaan selville, pystytään tuottamaan pysyvämpää tietoa organisaation käyttöön. Lopulta aiheen teoriaan perehtymisen myötä lopullisiksi tutkimuskysymyksiksi muotoutuivat: 1) Miten asiakastietojärjestelmän kehittävän ylläpidon osa-alue organisoidaan käyttäjäkeskeisesti? ja 2) Miten kehittävän ylläpidon resurssit kohdennetaan tehokkaasti? Molempiin voidaan vastata suunnittelemalla kehittävän ylläpidon prosessikuvaus.

6.3 Toteutus

Kehittämistyön toteutukseen kuului ideointipaja sekä haastattelut. Aikataulullisesti ensimmäisenä toteutettiin ideointipaja.

6.3.1 Kehittävän ylläpidon tiedonhankinnan suunnittelun työpaja

Tietojärjestelmän kehittävän ylläpidon suunnittelu aloitettiin ideointipajalla, johon osallistui tietojärjestelmäprojektissa työskenteleviä henkilöitä. Ideointipaja toteutettiin toukokuussa 2021 ja siihen osallistui:

- Organisaation prosessien asiantuntija 1
- Organisaation prosessien asiantuntija 2
- Tietojärjestelmäprojektin tuoteomistaja 1
- Tietojärjestelmäprojektin tuoteomistaja 2
- Tietojärjestelmäprojektin testaaja 1
- Tietojärjestelmäprojektin testaaja 2
- Tietojärjestelmäasiantuntija
- Opinnäytetyön tekijä fasilitoijan roolissa

Ideointipajan tavoitteena oli tuoda esiin tietojärjestelmäprojektissa tiiviisti työskennelleiden henkilöiden ideoita ja ajatuksia käyttäjien motivaatiosta osallistua kehittämiseen ja heidän osallistumistavoistaan ylläpitovaiheessa. Ideointipajaan kutsuttiin projektissa työskennelleitä henkilöitä kahdella perusteella: 1) He ovat työskennelleet Rikosseuraamuslaitoksella, joten heillä on hyvä käsitys kohdetietojärjestelmän käyttäjäkunnasta tai 2) Heillä on laaja kokemus tietojärjestelmien kehittämisestä ja käyttäjistä yleisesti. Lisäksi tiedettiin, että osa kutsutuista tulisi työskentelemään tietojärjestelmän jatkokehittämisessä ja ylläpidossa. Kutsutuille lähetetyssä kutsussa painotettiin, ettei tavoitteena ole selvittää, mikä tietojärjestelmän käyttämisessä koetaan haastavaksi, vaan ideoida tapoja saada haasteet selville. Ideointipajaan osallistuminen oli vapaaehtoista.

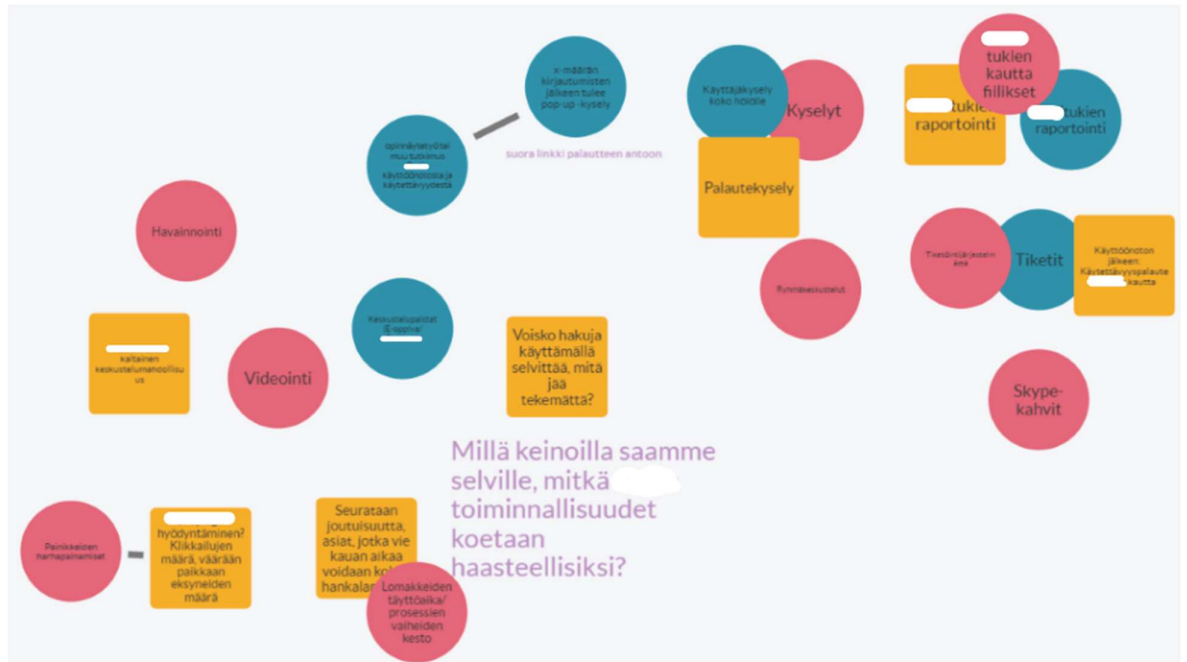
Opinnäytetyön tekijä halusi saada kaikki ideointipajaan osallistujat ideoimaan rohkeasti sekä motivoida heitä myös yhteiseen ideointiin, joten ideointipajan rakenne suunniteltiin etukäteen. Ideointipajasta suunniteltiin kaksiosainen ja molemmissa käytettiin fasilitointimenetelmää me-we-us. Me-we-us on menetelmänä yksinkertainen, mutta tehokas ja sen tarkoituksena on saada kaikki ryhmäläiset jakamaan ideansa muille. Menetelmän vaiheet ovat yksilötyöskentely, pienryhmätyöskentely ja koko ryhmän työskentely. Ensin osallistujat miettivät käsiteltävää asiaa yksin ja mahdollisesti kirjoittavat ajatuksiaan ylös. Sen jälkeen osallistujat jakautuvat pienryhmiin ja jakavat ajatuksensa muille. Pienryhmässä tarkastellaan ideoita ja valitaan yhdessä ne, mitkä halutaan viedä koko ryhmän tietoon. Lopuksi pienryhmäkohtaiset ideat esitellään koko ryhmälle ja esitetyistä ideoista valitaan ne ideat, joiden kanssa edetään. (Kankaanpää 18.1.2021.)

Käyttäjien kokemista haasteista tietojärjestelmän käytössä voidaan kerätä tietoa monilla eri menetelmillä. Useimmat menetelmät eivät toimi ilman käyttäjän osallistumista, vaan käyttäjän pitää tuoda näkemyksensä julki jollain tavalla. Kehittävän ylläpidon prosessikuvauksen menetelmien valinnassa on hyödyllistä ottaa huomioon myös käyttäjien motivaatio palautteen antamiseen. Tästä syystä ideointipajan ensimmäisessä vaiheessa mietittiin, mikä motivoi käyttäjää antamaan palautetta ja kehitysehdotuksia tai kertomaan haastaviksi kokemiaan asioita tietojärjestelmässä. Osallistujat työstivät ideoitaan me-we-us-menetelmällä, jota opinnäytetyön tekijä fasilitoi.

Käyttäjää motivoiviksi asioiksi ideointipajassa listattiin:

- Palautetta kysytään.
- Käyttäjällä on muutoksen tarve ja usko siihen, että palautteella on merkitystä ja mahdollisuus saada aikaan muutosta.
- Jokin hankaloittaa omaa työtä.
- Tunnetila; negatiivinen tai poikkeuksellinen helppous.
- Kiinnostus asiaa ja siihen vaikuttamista kohtaan.
- Palkinto palautteen antamisesta.
- Palautteen antamisen helppous.
- Mahdollisuus antaa palautetta anonyymisti.

Ideointipajan toisessa vaiheessa mietittiin erilaisia tiedonhankinnan menetelmiä, joilla voitaisiin hankkia tietoa siitä, mitkä asiat käyttäjät kokevat hankaliksi tietojärjestelmän käytössä. Menetelmien ideointiin käytettiin me-we-us-menetelmää ja tiedonhankintamenetelmien keräämisen jälkeen niitä arvioitiin sen perusteella, kuinka paljon resursseja niiden toteuttaminen vaatii sekä kuinka hyödynnettäviä niiden tulokset ovat. Arviointi toteutettiin sijoittamalla esiin tulleet ideat virtuaalisessa valkotaulussa arvioinnin nelikenttään. Arvioinnin nelikentän avulla voidaan arvioida asioita sen mukaan, kuinka helppoja ne on toteuttaa ja kuinka tärkeitä ne ovat sijoittamalla ne oikeaan kohtaan nelikentässä (Malkavaara 2021). Ideat sijoitettiin valkotaulussa vaaka-akselille sen mukaan, kuinka paljon resursseja niiden toteuttaminen vaatii niin, että mikäli toteutus vaatii vähän resursseja, se asetettiin nelikentän oikeaan laitaan ja mikäli paljon, nelikentän vasempaan laitaan. pystyakselille ideat sijoitettiin sen mukaan, kuinka hyödynnettäviä niillä saatavat tiedot olisivat niin, että mikäli tieto on hyödyllistä, se asetettiin nelikentän ylälaitaan ja mikäli vähemmän hyödyllistä, nelikentän alalaitaan. Näin nelikentän yläoikealle asettuivat ne menetelmät, millä saadaan mahdollisimman hyödynnettäviä tuloksia mahdollisimman pienillä resursseilla. Ideointipajan lopputulos on kuviossa 9, josta on peitetty järjestelmien nimet.



Kuvio 9. Ideointipajan lopputulos

Ideointipajan jälkeen opinnäytetyön tekijä koosti menetelmät taulukkoon (taulukko 1) ja arvioi ideointipajan keskusteluun perustuen menetelmien toteuttamista, menetelmän tulosten hyödynnettävyyttä ja menetelmän suhdetta käyttäjän motivaatioon. Menetelmän toteuttamiseen kuluva resurssi arvioitiin pisteyttämällä ne asteikolla 1-4 niin, että mitä vähemmän resurssia menetelmän toteuttaminen vaatii, sitä pienempi pistemäärä sille annettiin. Resurssiksi laskettiin käyttäjien osallistuminen sekä menetelmän suunnitteluun ja toteutukseen sekä tulosten analysointiin kuluvat resurssit. Tulosten hyödynnettävyyttä arvioitiin pisteyttämällä menetelmät asteikolla 1-4 niin, että mitä hyödynnettävämmät tulokset ovat, sitä pienempi pistemäärä menetelmälle annettiin. Hyödynnettävyyttä arvioitiin siitä näkökulmasta, kuinka todenmukaista ja käyttökelpoista tietoa haastaviksi koetuista asioista tietojärjestelmässä menetelmällä voitaisiin tuottaa. Viimeiseksi pisteytettiin menetelmät suhteessa käyttäjän motivaatioon asteikolla 1-4 niin, että mitä enemmän palautteen antamiseen liittyviä motivoivia seikkoja menetelmään liittyy, sitä pienempi pistemäärä sille annettiin. Mikäli menetelmän edellytyksenä ei ollut käyttäjän erikseen osallistuminen, asetettiin pistemääräksi vakiona 1. Lopuksi laskettiin jokaisen menetelmän pisteet yhteen. Arvioinnissa pienet resurssit, hyödynnettävät tulokset ja käyttäjää motivoivat ominaisuudet arvioitiin pienillä pistemäärillä, joten menetelmistä tehokkaimmiksi arvioitiin ne, joilla on pienimmät pistemäärät. Raporttiin liitetystä taulukosta jätettiin pois tiedonhankintamenetelmistä ne, joiden pistemääräksi arvioitiin yli seitsemän eli ne, joiden toteuttaminen olisi vähiten kannattavaa. Suurimmat pistemäärät saaneet menetelmät olivat vapaamuotoiset kahvihetket käyttäjien kanssa,

ryhmäkeskustelut, havainnointi, videointi, organisaation keskustelupalstojen seuraaminen, väärin painikkeiden painamisen selvittäminen ja tilastointi sekä prosessien vaiheiden keston seuranta.

Taulukko 1. Tiedonhankintamenetelmien vertailu

Menetelmä	Toteuttamiseen kuluvat resurssit (1-4)	Tulosten hyödynnettävyys (1-4)	Käyttäjän motivaatio (1-4)	Yhteensä
Tukipyyntöjen säännöllinen tilastointi	Tikettijärjestelmässä valmis raportointimahdollisuus. Ei vaadi käyttäjältä ylimääräisiä toimenpiteitä. (1)	Todellinen tuen tarve haastavissa kohdissa (2)	1	4
Opinnäytetyö/muu tutkimus käytettävyydestä	Ulkopuolinen toteutus (1)	Tutkimus (2)	(2)	5
Käyttäjäkysely	Kyselyn toteutus ja tulosten raportointi (2)	Anonyymi kysely kaikille käyttäjille (1)	(2)	5
Tukihenkilöiden säännöllinen raportointi heidän antamastaan tuesta	Raportointimenetelmän suunnittelu ja toteutus. Ei vaadi käyttäjältä (pl. tukihenkilöt) ylimääräisiä toimenpiteitä. (2)	Käyttäjien todellinen avuntarve (2)	(2)	6
Tiettyjen kirjautumiskertojen jälkeen pop up-kysely	Suunnittelu ja toteutus sovellustoimittajalta. Tulosten raportointi. (4)	Anonyymi kysely kaikille käyttäjille (1)	(2)	7
Kesken/tekemättä jääneiden toimenpiteiden tilastointi	Seurannan suunnittelu ja säännöllinen raportointi. Ei vaadi käyttäjältä ylimääräisiä toimenpiteitä. (3)	Tulosten luotettavuus; onko kesken jääminen merkki hankaluudesta (3)	(1)	7

Alhaisimman pistemäärän sai tukipyyntöjen säännöllinen tilastointi. Kun käyttäjä kohtaa tietojärjestelmän käytössä sellaisen tilanteen, johon hän tarvitsee apua, hän ottaa ensin yhteyttä oman yksikkönsä nimettyyn tukihenkilöön. Mikäli tukihenkilö ei pysty ratkaisemaan ongelmaa, tekee käyttäjä tukipyynnön tietojärjestelmän ylläpito-organisaatiolle erillisen toimintamallin mukaisesti. Tukipyynnön tekeminen on kannattavaa toteuttaa niin, että tukipyynnöistä on helppo tehdä erilaisia tilastoja esimerkiksi tukipyyntöjen aiheen perusteella. Koska käyttäjä on kohdannut tilanteen, jota hän tai tukihenkilö ei pysty ratkaisemaan, on hänen tehtävä tukipyyntö, mutta tämä menetelmä ei vaadi käyttäjältä muuta ylimääräistä osallistumista tai toimenpiteitä. Tukipyynnön tekeminen on myös suora indikaatio siitä, että kyseinen asia tietojärjestelmässä on

haastava. Kun tiettyyn tietojärjestelmän osioon saapuu useita tukipyyntöjä, on selkeää, että asiaan on reagoitava.

Toiseksi alhaisimman pistemäärän saivat opinnäytetyön tai muun tutkimuksen tekeminen tietojärjestelmän käytettävyydestä sekä käyttäjäkysely. Opinnäytetyö tai tutkimus ei sovellu jatkuvan kehittävän ylläpidon toimintamalliin sen ulkopuolisen toteutuksen vuoksi. Käyttäjäkyselyllä voidaan hankkia tietoa käyttäjien kokemista haasteista erityisesti sellaisista tietojärjestelmän osioista, joita he osaavat käyttää, mutta joiden käyttö on hidasta tai työlästä. Käyttäjäkysely soveltuu erityisesti tuotteen käyttövaiheeseen (Väänänen-Vainio-Mattila 2011, 119). Käyttäjäkyselyn etuina voidaan pitää sen monipuolisuutta ja kohdentamisen mahdollisuutta. Tulosten raportointi ja hyödynnettävyys ovat riippuvaisia kyselyn suunnittelusta ja toteutuksesta. Kysely voidaan toteuttaa anonymiina, se voidaan osoittaa suoraan käyttäjille ja siihen osallistumisesta voidaan tehdä käyttäjille helppoa. Säännöllisesti toteutetun käyttäjäkyselyn tuloksista voidaan nähdä myös tietojärjestelmän ja sen käytön kehitys.

Kolmanneksi alhaisimman pistemäärän sai tukihenkilöiden säännöllinen raportointi heidän antamastaan tuesta. Jokaiseen yksikköön on koulutettu tietojärjestelmän käyttäjistä tukihenkilöitä käyttäjille. Tukihenkilöiden on tarkoitus olla käyttäjien ensimmäisen asteen tuki heidän käytön haasteissaan, joten heille kertyy koko ajan realistista ja ajantasaista tietoa siitä, mitkä asiat tietojärjestelmän käytössä aiheuttavat käyttäjille haasteita. Tukihenkilöt on koulutettu yksikön omasta henkilökunnasta, joten käyttäjillä on todennäköisesti matalin kynnys ottaa yhteyttä juuri tukihenkilöön.

6.3.2 Haastattelut

Opinnäytetyötä varten haastateltiin kolmea asiakastietojärjestelmän suunnitteluun osallistunutta henkilöä ja kuutta asiakastietojärjestelmän tulevaa käyttäjää. Molempien ryhmien edustajien haastatteluilla haluttiin saada tietoa käyttäjäkeskeisyyden toteutumisesta ennen ylläpitovaihetta ja sen ideaalista toteutumisesta ylläpitovaiheessa. Haastateltavaksi haluttiin eri ryhmien edustajia, koska eri ryhmien edustajilla on asiaan erilaiset näkökulmat, jotka haluttiin tuoda esiin. Molempien ryhmien haastatteluissa käsiteltiin samoja teemoja, mutta asiakastietojärjestelmien suunnitteluun osallistuneiden henkilöiden haastattelurungossa (liite 1) käsiteltiin enemmän käyttäjäkeskeisyyden haasteita ja asiakastietojärjestelmän tulevien käyttäjien haastattelurungossa (liite 2) painopiste oli käyttäjäkeskeisyydessä ylläpitovaiheessa.

Haastateltavat osallistuivat haastatteluun vapaaehtoisesti ja haastatteluun osallistuminen ei vaatinut heiltä etukäteisvalmistautumista. Haastattelut toteutettiin alle tunnin pituisina

yksilöhaastatteluina touko-kesäkuussa 2021 etäyhteydellä ja haastattelut tallennettiin haastateltavien suostumuksella. Opinnäytetyön tekijä on itse työskennellyt tietojärjestelmän suunnittelijana. Haastatteluissa opinnäytetyön tekijä kertoi työskentelystään suunnittelijana, mutta korosti haastateltaville, että haastatteluilla halutaan saada tietää käyttäjien rehelliset näkemykset ja kokemukset, joita ei ole tarvetta muuttaa tai peitellä sen vuoksi, että haastattelija on toiminut suunnittelijana.

Haastatteluiden jälkeen haastattelut dokumentoitiin kirjallisesti litteroimalla. Litterointia ei toteutettu sanatarkasti, koska haastatteluista pyrittiin tunnistamaan yhteisiä teemoja eikä tarkastella sanavalintoja. Litteroinnin jälkeen aineistoa ryhdyttiin analysoimaan. Aineisto luettiin useamman kerran ja joka kierroksella tunnistettiin enemmän teemoihin liittyviä lauseita tai sanoja ja pystyttiin tunnistamaan paremmin oleelliset teemat. Opinnäytetyön tekijä pyrki teemoittelua tehdessään pitämään fokuksen opinnäytetyön tavoitteissa, mutta suhtautui avoimesti myös teemoihin, jotka eivät suoraan vaikuttaneet liittyvän käyttäjäkeskisyyteen tai sen toteuttamiseen ylläpidossa. Analyysia ja teemoittelua tehdessään opinnäytetyön tekijä huomioi myös oman kokemuksensa tietojärjestelmän suunnittelusta. Huomiointi tehtiin sekä havaitsemalla sen vaikutus tekijän ajatteluun että lisäämällä se aineistoon.

Lopulliset teemat jaettiin kolmeen ryhmään ja jokaisen ryhmän tulokset koostettiin haastateltavittain liitteinä oleviin taulukoihin (liitteet 4-6). Ensimmäisen ryhmän teemoiksi tunnistettiin **käyttäjäkeskisyyden toteutuminen ennen ylläpitovaihetta** sekä **tietojärjestelmän monimutkaisuus**. Toisen ryhmän teemojen yhteisenä nimittäjänä on käyttäjäkeskisyyden haasteet ja siinä teemoiksi tunnistettiin **tarpeiden määrittelyn vaikeus ilman konkretiaa, erilaiset käyttäjät ja yksiköt** sekä **käsitteet ja käytettävä kieli**. Kolmannen ryhmän teemat käsittelevät käyttäjäkeskisyyttä ylläpitovaiheessa, jossa teemoiksi tunnistettiin **palautteen antamisen helppous, palautteen merkitys** sekä **palautteen hankinta**. Ryhmien lisäksi opinnäytetyöhön haluttiin nostaa yhtenä teemana **käyttäjätiedon vaikutus ja merkitys** (liite 7), mikä nousi esiin suunnittelijoiden haastatteluissa monista eri näkökulmista.

Kaikki haastatellut käyttäjät ja suunnittelijat mainitsivat, että ajankohta uuden tietojärjestelmän käyttöönotolle on hyvin haastava. Maailmanlaajuinen tilanne on vaikuttanut ja vaikuttaa käyttäjien työhön ja koko organisaatiossa on kesken meneillään oleva organisaatiouudistus. Toisaalta käyttäjät pitivät lähtökohtaisesti positiivisena asiana, että kaikki tarvittava tieto tulee olemaan uudessa asiakastietojärjestelmässä ja se yhdistää aiemmat tietojärjestelmät.

6.3.3 Käyttäjakeskeisyyden toteutuminen ennen ylläpitovaihetta

Jotta käyttäjakeskeisyys pystytään huomioimaan ja sitä jopa lisäämään asiakastietojärjestelmän ylläpitovaiheessa, on kehittämistyön pohjaksi selvitettävä käyttäjakeskeisyyden lähtötilanne, jossa kuvataan käyttäjakeskeisyyden toteuttamisessa ennen ylläpitoa tehdyt onnistuneet ratkaisut ja kehittämisen kohdat. Kehittävän ylläpidon suunnittelussa ja prosessikuvauksen teossa voidaan hyödyntää taustana aiempia onnistumisia ja haasteita. Käyttäjakeskeisyys asiakastietojärjestelmän kehittämisen aiemmissa vaiheissa oli valikoitunut myös yhdeksi haastatteluiden teemaksi, joten on luonnollista, että haastateltavat myös kertoivat siitä. Ennen ylläpitovaihetta tapahtuneen käyttäjakeskeisyyden teeman ja käyttäjakeskeisyyden haasteisiin liittyvien teemojen käsittelyn yhteydessä muodostuu kuva kohdeasiakastietojärjestelmän käyttäjakeskeisen kehittämisen lähtötilanteesta ennen käyttöönottoa ja ylläpitoa. Teemojen käsittely haastateltavittain on koottu liitteisiin 4 ja 5.

Kaikilta käyttäjiltä kysyttiin, miten he olivat kokeneet käyttäjakeskeisyyden toteutumisen ja kokevatko he, että heitä on kuunneltu asiakastietojärjestelmää kehitettäessä. **Kaksi haastateltavaa ei muistanut yhtään tilannetta, jossa he olisivat kokeneet, että heidän näkemyksiään olisi kysytty tai olisivat voineet tuoda tarpeitaan esille.** Kolmella haastateltavalla oli muistikuvia saapuneista tarvekyselyistä tai mahdollisuuksista esittää kysymyksiä ja toiveita, mutta eivät muistaneet itse osallistuneensa niihin. Yksi haastatelluista (H1) oli osallistunut testauspäiviin ja piti sitä hyvänä tilaisuutena tuoda omia tarpeitaan esille. Hän oli myös keskustellut tietojärjestelmästä muiden käyttäjien kanssa ja havahtunut siihen, ettei käyttäjien tarpeita voi aina toteuttaa suoraan tietojärjestelmään, koska sen kehittämiseen vaikuttaa moni muukin asia, kuten laki ja siihen kohdistuvat muutokset tai toimintatapojen tulevat muutokset. **Viisi kuudesta haastatellusta siis koki, ettei käyttäjien tarpeita oltu kuultu ollenkaan tai eivät ainakaan itse olleet osallistuneet siihen.** Osa mainitsi kuitenkin, että vaikka he eivät muista, että käyttäjillä olisi ollut mahdollisuus esittää tarpeita, ei se tarkoita varmaksi, etteikö niin olisi kuitenkin tehty. Yksi haastateltava (H1) totesi, että luottaa siihen, että tietojärjestelmää ovat kehittämässä ammattitaitoiset ja osaavat henkilöt ja eräs haastateltava (H4) mainitsi tietojärjestelmässä olevan sellaisia ratkaisuja, joita olisi uskonut tietojärjestelmän kehittämisen ammattilaisten osaavan tehdä paremmin. Kyseessä oli siis tietojärjestelmien yleisiin lainalaisuuksiin liittyvä asia, eikä kyseisen tietojärjestelmän käyttäjiin tai toimintaan liittyvä.

Osa asiakastietojärjestelmän suunnittelijoista oli valittu työskentelemään projektiin niiden henkilöiden joukosta, jotka käyttävät korvattavia tietojärjestelmiä ja tulisivat siis

käyttämään myös uutta asiakastietojärjestelmää. Kaikki haastatellut suunnittelijat olivat tällaisia ja kaikki toivat myös esiin, että he pitivät lähtökohtaisesti onnistuneena ratkaisuna sitä, että asiakastietojärjestelmän suunnittelijoina on ollut sen tulevia käyttäjiä. Myös opinnäytetyön tekijän kokemuksen mukaan ratkaisu on pääasiassa ollut onnistunut, muttei ongelmaton. Vaikka opinnäytetyön tekijä on edustanut suunnittelussa käyttäjää, ei hän koe edustaneensa kaikkia tai edes valtaosaa käyttäjistä. Yksi suunnittelijoista (S2) piti erityisen tärkeänä, että suunnitteluun otetaan täysipainoisesti mukaan henkilöitä, jotka tekevät työtä, jonka tekemiseen tietojärjestelmää suunnitellaan, eikä heidän tarpeitaan selvitetä suunnittelun taustaksi tai ulkopuolella.

Vaikka kaikki suunnittelijat kokivat olevansa myös käyttäjiä, oli projekti kestänyt niin pitkään, että suunnittelijat epäilivät oman käyttäjätietonsa ajantasaisuutta. Eräs suunnittelija kuvasi tilannetta näin:

"Jotenkin itse on mieltänyt itsensä myös käyttäjänä, mikä ei tietysti ehkä enää pidä paikkaansa et se oma käsitys on väärä, mut silleehän sitä on lähteny määrittelee, et miten mä käyttäisin tätä järjestelmää, mikä on mulle se toimintatapa ja miten ne käyttäjät haluais sitä käyttää ja mitä sielt pitää löytyy. Ja se et ei me voida kaikilta kysyy et miten sä haluaisit tän toimivan." (S1)

Kaksi haastatelluista suunnittelijoista (S2 ja S3) toivat vahvasti esiin sen, että suunnittelijoina toimineiden käyttäjien olisi pitänyt saada enemmän koulutusta ja ohjausta tietojärjestelmän kehittämiseen etenkin projektin alkuvaiheessa. Heidän mukaansa kehittämisen haasteet johtuvat osittain käyttäjistä suunnittelijoiksi siirtyneiden henkilöiden puutteellisista projektiosaamisesta, käytettävyyssosaamisesta ja teknisestä osaamisesta, mikä heijastuu suoraan tietojärjestelmän toiminnallisuuksiin esimerkiksi epäjohton mukaisuutena tai eri toiminnallisuuden yhteensopimattomuutena. Projektin aikana osa suunnittelijoista on myös vaihtunut, mikä näkyy erilaisina tyyleinä suunnitella tietojärjestelmää. Tietojärjestelmän yhdenmukaisuuteen olisi pitänyt panostaa enemmän resursseja. Eräs suunnittelija kuvasi tilannetta näin:

"Siellähän on sit aina se, että --- tässä asiassa tää tehdään tän logiikalla, mutta tässä toisessa tää tehään tän toisen logiikalla. Ja sit taas kun siel on se käyttäjä, joka käyttääkin näitä monen eri henkilön tekemää juttua, --- sillähän ei oo sitte enää mitään logiikkaa, koska sillähän on se oma logiikka vielä ja niis ei välttämättä mikään mee yhteen hänen kanssaan niin sit jos siel ois vaan se yks yleinen käytettävyyssiikka, --- sitä vois sitte ainaki sanoo, että tää on aina näin." (S3)

Toinen suunnittelija kuvasi haasteita käyttäjien toimimisessa suunnittelijoina näin:

"Meil ei oo käsitystä eikä aikasempaa kokemusta järjestelmän kehittämisestä juurikaan ollu, niin se on varmaan voinut vaikuttaa siihen, että meidän ajatuksia on vahvasti ohjannut ne vanhat järjestelmät ja vanhat tavat ratkaista asioita. Ja se on varmaan alkuvaiheessa ollu vähän haastava, että miten saadaan löydettyä sellaisia ratkaisuja, että ne olisi suoraviivaisempia ja helpompia ja irti niistä vanhoista malleista." (S2)

Myös opinnäytetyön tekijän kokemuksen mukaan olisi ollut hyödyllistä jo aikaisemmin panostaa resursseja yhdenmukaisiin ratkaisuihin ja myös niiden noudattamiseen. Suunnittelijoiden ja kehittäjien vaihtuvuus on nähtävissä tietojärjestelmän isoissa kokonaisuuksissa ja yksityiskohdissa. Suunnittelijoiden osaamista laajaa käyttäjäjoukkoa koskevan käyttäjätiedon hankkimiseen ja hyödyntämiseen olisi pitänyt lisätä.

Kaikki suunnittelijat mainitsivat jatkuvan tulevien käyttäjien tekemän testauksen hyvänä käyttäjäkeskeisyyttä lisäävänä tekijänä ja kokivat, että testaustapahtumista on saanut jatkuvasti reaaliaikaista tietoa käyttäjien näkemyksistä. Yksi suunnittelija toi esiin käyttäjäkeskeisyyden haasteena myös käyttäjien tavoittamisen ja sitouttamisen kehittämistyöhön.

6.3.4 Tietojärjestelmän monimutkaisuus

Koska haastattelujen aiheena oli asiakastietojärjestelmä, on luonnollista, että yhdeksi teemaksi niissä nousi myös sovellus ja sen käyttäminen. **Kaikki haastatellut käyttäjät toivat monesta näkökulmasta esiin tietojärjestelmän monimutkaisuuden ja tietynlaisen hajanaisuuden.** Käyttäjät kuvailivat, että heidän on haastavaa hahmottaa tietojärjestelmän kokonaisuuksia ja sitä, mitkä asiat liittyvät toisiinsa. Tietojärjestelmän suunnittelijat tunnistivat haastattelussa tämän ominaisuuden myös. Käyttäjät kuvailivat asiakastietojärjestelmän sovellusta sekavaksi, paljon klikkauksia vaativaksi ja siellä navigointi koettiin hankalana. Eräs haastateltava (H1) koki vaikeana sen, ettei sovellus ole tarpeeksi yhdenmukainen eli jokin sovelluksen tietty toiminnallisuus ei ole toteutettu samanlaisena kaikkialle sovellukseen, vaan niissä on pieniä poikkeuksia, jotka hankaloittavat tietojärjestelmän käytön opettelua.

Neljä käyttäjää (H2, H3, H4 ja H6) korostivat haastatteluissaan sitä, etteivät koe tietojärjestelmän käytön olevan osa heidän perimmäistä työnkuvaansa, vaan korkeintaan työn tukena oleva tehtävä. Käyttäjät kokivat, että heidän todellinen työnsä on työskentely asiakkaiden kanssa, jota tietojärjestelmään kirjattavat tiedot tukevat. Koska he kokivat, että tietojärjestelmä ja sen käytön tulisi tukea heidän todellista työtään, kokivat he tietojärjestelmän monimutkaisuuden ja sen opetteluun käytettävän ajan kohtuuttomaksi. Eräs käyttäjä (H3) mainitsi, että hänen odotuksensa uutta tietojärjestelmää kohtaan olivat, että se auttaisi ja helpottaisi hänen työtään, mutta tällä hetkellä hänestä tuntui, että tietojärjestelmä oikeastaan vaikeutti hänen työtään, koska hän ei kokenut löytävänsä sovelluksesta tarvitsemaansa tietoa tai sieltä sopivaa paikkaa tarvittavan tiedon kirjaamiseen. Toisaalta kaksi käyttäjää (H1 ja H4) totesi, että testitilanteessa tapahtuva sovelluksen käyttö on niin erilaista todellisiin tilanteisiin verrattuna, että näkemyksen muodostaminen tietojärjestelmästä on haastavaa.

Testitilanteessa voidaan tietojärjestelmässä esimerkiksi tehdä muutamassa tunnissa toimenpiteet, jotka todellisessa elämässä tehtäisiin muutamien viikkojen ajanjakson aikana. Testaustilanteen käyttötilanne ei siis vastaa todellista käyttötilannetta, mikä vaikuttaa tietojärjestelmän käyttökokemukseen ja muodostuneeseen mielipiteeseen siitä. Testitilannetta aidompi käyttökokemus muodostuu vasta tietojärjestelmän aidossa käyttöympäristössä ja -tilanteessa.

Haastatelluista käyttäjistä H2, H3, H4 ja H6 toivat esiin myös, että kokevat haastavaksi löytää aikaa opiskelulle. **Sovellus koettiin niin monimutkaiseksi, että sen käytön opetteluun tarvittaisiin enemmän aikaa, kuin nyt oli käytettävissä.** Osa ulotti huolensa myös käyttöönoton jälkeiseen aikaan ja oli huolissaan siitä, kuinka kauan vie aikaa, että uuden tietojärjestelmän käyttö olisi niin rutiinia, että se veisi saman verran tai vähemmän aikaa kuin nykyisten tietojärjestelmien käyttö. Käyttäjät kokivat, että kaikki aika, joka kuluu tietojärjestelmän käytön opetteluun ja osin myös sen käyttöön, on pois heidän perustyöstään asiakkaiden kanssa. Käyttäjistä H2 sekä H4 mainitsivat myös, että odottavat teknistä koulutusta enemmän sisällöllistä ohjausta siihen, mitä tietojärjestelmään kirjataan ja miten kirjaukset tukevat heidän työtään. Myös haastateltu suunnittelija S1 ilmaisi yhdeksi isoimmaksi huolenaiheekseen sen, kuinka kauan aikaa kuluu tietojärjestelmän käytön rutinoitumiseen ja ajan tasalle saattamiseen.

Myös tietojärjestelmän suunnitteluun osallistuneet henkilöt toivat haastatteluissa esiin tietojärjestelmän monimutkaisuuden ja silppumaisuuden, mutta he käsittelivät asiaa hieman eri näkökulmasta kuin käyttäjät. Suunnittelijat yhdistivät monimutkaisuuden suunnittelijoiden vaihtuvuuteen sekä tietojärjestelmän kehittämisen tapaan. He toivat esiin, että osin ketterän kehittämisen periaatteiden vastaisesti tietojärjestelmän kehittämisen alussa olisi pitänyt käyttää enemmän resursseja tietojärjestelmän kokonaisuuden suunnitteluun. Kaikki suunnittelijat mainitsivat, että koko tietojärjestelmän hahmottaminen ja kuvaaminen tarkemmalla tasolla olisi parantanut tietojärjestelmän loogisuutta ja tehnyt kokonaisuudesta yhdenmukaisemman ja selkeämmin hahmotettavan. Myös opinnäytetyön tekijän kokemuksen mukaan tietojärjestelmän kehittämisen alussa olisi ollut tehokasta kuvata koko tietojärjestelmä, jotta yksittäistä toiminnallisuutta määriteltessä sen liittyminen tietojärjestelmän muihin osioihin olisi ollut alusta alkaen tiedossa ja se olisi voitu ottaa huomioon. Kokonaisuuden hahmottaminen olisi edesauttanut asioiden toteuttamista yksinkertaisemmin sekä toiminnallisuuksien priorisointia, jolloin kehittämisessä olisi pystytty kohdentamaan resursseja paremmin.

Asiakastietojärjestelmää on suunniteltu ja toteutettu ketterällä scrum-mallilla, mutta suunnittelijat mainitsivat haastatteluissa, että tässä kohden projektia asiaa tarkastellessa,

ei kyseinen malli heidän kokemuksensa mukaan ole ollut ideaali valinta kehittämisen tavaksi. Kun sovellusta toteutetaan pieni osio kerrallaan, voi sovellus kokonaisuutena jäädä monimutkaiseksi ja hajanaiseksi ja osioiden yhteensovittaminen olla vaikeaa. Yhdeksi haasteeksi suunnittelijat mainitsivat myös sen, että tietojärjestelmän käyttöönotto on koko ajan ollut sen hetkisen tiedon mukaan niin lähellä, että priorisointia on jouduttu tekemään käyttöönoton lähestymisen perusteella, mutta käyttöönoton siirtyessä priorisointia olisi pitänyt tehdä eri perusteilla.

6.3.5 Käyttäjakeskeisyyden haasteet

Haastatteluissa tuotiin esiin erilaisia käyttäjakeskeisyyden toteuttamiseen liittyviä haasteita sekä suunnittelijoiden että käyttäjien näkökulmasta. Kolme haastetta, tarpeiden määrittelyn vaikeus ilman konkretiaa, erilaiset käyttäjät ja yksiköt sekä käsitteet ja käytettävä kieli, nousivat esiin omina teemoinaan. Teemojen käsittely haastateltavittain on koottu liitteeseen 5.

Kolme haastateltua käyttäjää (H2, H4 ja H5) mainitsi erikseen, että **kokevat vaikeaksi esittää toiveita tai tarpeita ”tyhjästä” ennen kuin näkevät jotain konkreettista tai pääsevät itse kokeilemaan tietojärjestelmän käyttöä.** Vaikeutta kuvattiin esimerkiksi näin:

”Harva meist kuitenkaan tommosta teknistä puolta osaa, et ei oo sellasta tuntemusta tai tietämystä et pystyiskää, en ainakaan ite ois osannu sanoo mitään ku nyt vast sitte ku ite sinne [koulutusympäristöön] pääsee.” (H2)

Myös haastateltavat H1 ja H2 mainitsivat, että ovat lähimenneisyydessä esittäneet havaintoja ja toiveita, miten toteutusta voisi muuttaa, jotta se paremmin tukisi heidän työtehtäviään tai olisi helpompi käyttää. Tilanne on kuitenkin ollut se, että havaintoja ei ole enää ollut mahdollista huomioida ennen tietojärjestelmän käyttöönottoa, vaan ne on siirretty huomioitavaksi jatkokehityksessä. Myös suunnittelija S1 oli tunnistanut tilanteen. Tilanne on ollut turhauttava ja tuo esiin tietojärjestelmän kehittämisen ajallisen ongelman. Käyttäjien tarpeita selvitetään ennen kuin mitään konkreettista on toteutettu, jotta voidaan varmistaa, että suunnittelussa tehdään heti alusta alkaen oikeita ja käyttäjien työtä tukevia ratkaisuja. Käyttäjien on kuitenkin vaikeaa sanoittaa, miten haluavat tietojärjestelmän toimivan ennen kuin näkevät tai pääsevät kokeilemaan itse, miten se voisi toimia. Projekteissa ei kuitenkaan aina ole resurssien vuoksi mahdollista toteuttaa samoja toiminnallisuuksia useaan kertaan testattavaksi ja kokeiltavaksi. Tämä tilanne johtaa siihen, että tietojärjestelmän toiminnallisuus toteutetaan käyttäjien tuottamia näkemyksiä noudattaen ja sitten, kun käyttäjät pääsevät kokeilemaan tietojärjestelmää itse ja pystyisivät antamaan siitä palautetta ja kehitysehdotuksia, ei niiden toteuttamiseen välttämättä ole tarpeellisia resursseja.

Ennen kuin mitään konkreettista on toteutettu, voivat käyttäjien tarpeet olla liian yleisluontoisia, mikä voi johtaa siihen, että tarve on toteutettu, mutta se ei kuitenkaan vastaa käyttäjän työnkuvaa. Esimerkiksi jos suunnittelun alkuvaiheessa tarpeeksi kirjataan ”Asiakastietojärjestelmään on pystyttävä kirjaamaan henkilön yhteystiedot”, ei kukaan todennäköisesti ole eri mieltä tarpeen toteuttamisen tarpeellisuudesta. Kyseinen tarve voidaan kuitenkin toteuttaa tietojärjestelmään monin eri tavoin, jolloin tarpeen perusolemus on toteutettu, mutta se ei välttämättä silti palvele työntekijää. Ratkaistavaksi jää edelleen esimerkiksi mitä tietoja tarkoitetaan yhteystiedoilla, kuka voi kirjata yhteystiedot, milloin yhteystiedot poistuvat, missä muodossa ja kenelle yhteystiedot näytetään ja niin edelleen. Sekä tarpeiden ajoituksen että tarkkuuden haasteisiin tulisi löytää ratkaisu, jolla resurssien puitteissa mahdollistetaan tarpeeksi tarkan tason protoilu käyttäjille ja mahdollisuus antaa siitä palautetta.

Käyttäjät ja suunnittelijat toivat haasteena esiin sen, että asiakastietojärjestelmän käyttäjäkunta on hyvin heterogeeninen sekä käyttö- että toimintatavoiltaan. **Käyttäjissä on hyvin erilaisia tapoja käyttää tietojärjestelmiä ja tekniikkaa yleensä ja eri yksiköt ovat toiminnaltaan ja toimintatavoiltaan erilaisia.** Tietojärjestelmän tulisi tukea käyttäjien toimintaa, mutta on haastavaa löytää tapa, joka tukee kaikkien käyttäjien toimintaa, mikäli ne poikkeavat toisistaan. Kaikki suunnittelijat toivat esiin myös, että vaikka he itse edustavat suunnittelussa myös käyttäjää, on käyttäjien joukossa niin erilaisia käyttäjiä, käyttötapoja ja myös yksikkökohtaisia toimintaeroja, että heidän näkemyksensä on vain yksi näkemys kaikkien käyttäjien joukossa.

Yhdeksi käyttäjäkeskeisyyden haasteeksi mainittiin **käsitteet ja käytettävä kieli** kahdesta eri näkökulmasta. Ensimmäinen on se, että käyttäjät kokivat, etteivät aina ymmärrä tietojärjestelmän kehittämiseen liittyvää teknistä sanastoa tai ymmärtävät sen väärin. Tästä syystä he kokevat ohjeistuksen tai saamansa vastaukset vaikeiksi ymmärtää. Kaksi haastateltavaa (H4 ja H6) mainitsi, että se, ettei ymmärrä termistöä tai pelkää, ettei osaa käyttää oikeita termejä, voi johtaa siihen, ettei lopulta anna palautetta tietojärjestelmästä, vaikka muuten olisi motivoitunut.

Käyttäjät kokivat lisäksi tietojärjestelmässä käytettävät termit vieraiksi sekä vaikeaksi ymmärtää ja yhdistää asiayhteyksiin, mikä vaikeuttaa myös käytön opettelua. Toisaalta yksi haastateltava (H4) totesi, että myös arkielämässä jokin termi voi tarkoittaa eri yksiköissä tai eri ihmisten käytössä eri asiaa, joten termistö, jonka kaikki ymmärtäisivät samalla tavalla, on haastavaa laatia.

6.3.6 Käyttäjakeskeisyys ylläpitovaiheessa

Haastatteluiden yhtenä teemana oli käyttäjien osallistuminen tietojärjestelmän kehittämiseen ylläpitovaiheessa ja tavoitteena oli selvittää, millä tavoin käyttäjien näkemyksien mukaan käyttäjien tarpeita pystyttäisiin parhaiten selvittämään ylläpitovaiheessa. Haastatellut käyttäjät kokivat suhtautuvansa perustasolla palautteen antamiseen ja kehitysehdotuksien esiin tuomiseen positiivisesti. Muutama haastateltava sanoi pyrkivänsä esimerkiksi aina antamaan palautetta koulutuspäivistä, johon on osallistunut. Tosin jo se, että on ollut halukas haastateltavaksi asiakastietojärjestelmän kehittämistä koskevaan opinnäytetyöhön voi olla osaltaan merkki siitä, että kaikki haastateltavat suhtautuvat palautteen antamiseen ja kehittämistyöhön keskimääräistä positiivisemmin. Keskeisiksi käyttäjakeskeisyydessä huomioon otettaviksi teemoiksi nousi palautteen antamisen helppous, palautteen merkitys sekä palautteen hankinnan keinot. Teemojen käsittely haastateltavittain on koottu liitteeseen 6.

Kaikki kuusi haastateltua käyttäjää sanoivat, **että tapa palautteen antamiseen tulisi olla mahdollisimman helppo**. Helpoksi tavaksi käyttäjät mainitsivat esimerkiksi sähköpostin. Palautteen antamisen helppous madaltaa kynnystä antaa palautetta, mitä käyttäjät pitivät tärkeänä. Jos palautteen antamisen vuoksi pitäisi kirjautua erilliseen ohjelmaan tai täyttää erillinen lomake, niin siinä tilanteessa palautteen antaminen jää käyttäjiltä kesken tai tekemättä. Eräs haastateltava (H4) totesi, että vaikka pitäisi palautteen antamista tärkeänä ja lähtökohtaisesti sitä tekee, niin kiireessä jättää tekemättä sellaiset asiat, joita ei ole pakko tehdä. Sama haastateltava mainitsi, että hänen motivaatioonsa vastata kyselyyn vaikuttaa suuresti se, miten helppo kyselyyn on vastata. Mikäli kyselyyn vastaaminen vie paljon aikaa tai tuntuu vaikealta, hän ei todennäköisesti silloin tee kyselyä loppuun asti. Suunnittelijoista S3 korosti sitä, että palautteen hankinnalla tulisi varmistaa palautteen saaminen mahdollisimman monelta käyttäjältä, jotta mahdollisimman ison osuuden käyttäjistä näkemykset voitaisiin huomioida.

Neljä haastateltua käyttäjää (H1, H2, H3 ja H6) mainitsivat, että he pitävät tärkeänä sitä, että tietojärjestelmän kehittämisestä vastaava organisaatio on aktiivisesti yhteydessä käyttäjiin ja **pyytävät palautetta jatkuvasti**. Myös suunnittelijat (S1 ja S3) toivat esiin, että kehittävän ylläpidon keskiössä tulisi olla käyttäjien tekemät havainnot, joiden aktiiviseen selvittämiseen myös panostetaan resursseja. Yksi haastateltava kuvasi palautteen pyytämisen merkitystä näin:

”Kyllä mä näkisin, että se olisi tärkeätä, että sieltä päin, ettei vaan heitetä luuta, et pitäkää hyvänänne ja tehkää sillä mitä teette, vaan just sitä kiinnostusta, että miten toimii ja mitä sieltä herää, ihmetystä ja että useampaan kertaan tulis sit, ettei vaan yks iso kysely vaan ois useempi kysely.” (H2)

Haastateltavat pitivät myös tärkeänä, että käyttäjille viestitään aktiivisesti, että heidän näkemyksillään on merkitystä ja että palautteella on aito vaikutusmahdollisuus tietojärjestelmään. Palautteen merkityksen korostaminen kasvattaa haastateltavien näkemyksien mukaan käyttäjien motivaatioon osallistua kehittämiseen. Eräs haastateltava (H1) kuvasi saaneensa kehittämisessä mukana olemisesta hyvän mielen, kun hän sai itse hyvää palautetta siitä, että oli asiakastietojärjestelmän testauksen yhteydessä havainnut tietojärjestelmän toimivan virheellisesti ja tuonut havaintonsa esiin. Lisäksi usea haastateltava (H2, H3 ja H6) totesi, että mikäli kehitysehdotuksen lähettämisen jälkeen siitä saisi jonkinlaisen palautteen, esimerkiksi vastauksen toimenpiteistä, joihin kehitysehdotuksen vuoksi on ryhdytty, motivoisi se lähettämään kehitysehdotuksia jatkossakin. Eräs haastateltu käyttäjä kuvasi asiaa näin:

”Tietenkin se, että sillä on jotain merkitystä, että niitä huomioita, että pyydetään tekemään huomioita ja kertomaan ja antamaan palautetta, että semmoinen kokemus siitä, että mä annan palautetta ja huomioita ja sit se ei niinku johda mihinkään, niin on kuin tuuleen huutaisi, niin semmoinenhan turhauttaa ja sit sitä lakkaa tekemästä.” (H3)

Yhtenä keinona pyytää palautetta ja kehitysehdotuksia tuotiin esiin **käyttäjäkyselyn** toteuttaminen. Useampi haastateltava toi esiin, että kyselyitä tulisi toteuttaa jatkuvasti, vaikka niiden välillä voisikin kulua pidempi aika. Kysyttäessä tarpeesta järjestää tilaisuuksia, joissa palautetta tai kehitysehdotuksia voisi tuoda esiin myös suullisesti, esimerkiksi **käyttäjahaastattelut** tai **yhteiskehittämistilaisuudet**, suhtautuivat kaikki haastatellut käyttäjät siihen positiivisesti ja näkivät omalta osaltaan hyvin mahdollisena, että voisivat osallistua kyseisiin tilaisuuksiin. Haastateltavat (H2 ja H3) perustelivat erillisten tilaisuuksien tarvetta sillä, että kyse on niin vieraasta ja monimutkaisesta asiasta, että asian esittäminen suullisesti voi joissain tilanteissa olla helpompaa ja riskit väärinymmärryksiin pienempiä kuin silloin, jos ajatuksensa esittää kirjallisesti.

Käyttäjien näkemyksien selvittämisen ja pyytämisen tulee olla lähtöisin kehittävän ylläpidon organisaatiosta, mutta kaikki haastatellut käyttäjät toivat esiin, että käyttäjillä pitäisi olla mahdollisuus tuoda näkemyksiään esiin myös oma-aloitteisesti. Haastateltavat mainitsivat, että keinon saattaa näkemyksensä kehittävän ylläpidon organisaatiolle tulisi olla mahdollisimman helppo ja kynnys toimintaan ryhtymiseen mahdollisimman matala. Haastateltavat pitivät tärkeänä, että oman viestinsä pystyisi lähettämään melko vapaamuotoisesti, jotta kynnys sen kirjoittamiseen ei kasva liian korkeaksi. Matala kynnys kehitysehdotuksen lähettämiseen tarkoittaa myös, ettei lähettämisen tulisi edellyttää oikeiden termien käyttöä tai kaikkien näkökulmien esiintuomista.

Eräs haastateltava totesi yleisesti käyttäjäkeskeisyydestä ylläpitovaiheessa näin:

*”Varmaan tärkeintä olisi se, että sitä ylipäänsä on, muodossa vaikka toisessa, että jää sitten jokaisen omaan harkintaan, että minkä verran siihen asiaan sitten tarttuu.”
(H6)*

6.3.7 Käyttäjätiedon vaikutus ja merkitys

Erityisesti suunnittelijoiden haastatteluissa käsiteltiin ja korostui se, kuinka paljon käyttäjien näkemykset ja tarpeet vaikuttavat tietojärjestelmän kehitykseen ja miten ne suhteutuvat muihin tarpeisiin, joita tietojärjestelmään kohdistuu. Käyttäjien näkemyksiä ja käyttäjakeskeisyyden toteuttamista pidettiin tärkeänä, mutta tietojärjestelmän kehittämiseen vaikuttaa moni muukin asia ja haastavana pidettiin eri tarpeiden ja erilaisen tiedon yhteensovittamista niin, että lopputulos olisi paras mahdollinen. Suunnittelijat mainitsivat tämän tarpeiden yhteensovittamisen olleen yksi haasteellisimmista asioista koko tietojärjestelmän kehittämisessä ja opinnäytetyön tekijän kokemus on samanlainen. Suunnittelijoiden näkemykset teemaan liittyen on koottu taulukkoon liitteessä 7.

Kyseisen tietojärjestelmän kehittämiseen ja sen toiminnallisuuksiin ovat vaikuttaneet laki ja lain tulevat muutokset, asetukset ja määräykset, tarve yhdenmukaistaa organisaation toimintatavat sekä tahtotila, mihin suuntaan organisaatio haluaa kehittää toimintaansa. Käyttäjät eivät välttämättä ole tietoisia etenkään kaikista tulevista muutoksista tai tavoitteista, mihin tietojärjestelmällä pyritään ja se voi aiheuttaa käyttäjille kokemuksen, ettei käyttäjien tarpeita ole kuunneltu. Erityisesti projektin resurssirajaukset voivat myös johtaa siihen, että käyttäjakeskeisyydestä ja käyttäjää helpottavista toiminnallisuuksista joudutaan karsimaan.

Suunnittelijat ovat joutuneet arvioimaan myös käyttäjätiedon ja käytettävyyden suhdetta toisiinsa tehdessään päätöksiä siitä, toteutetaanko tietojärjestelmän toiminnallisuus käyttäjien näkemyksien vai käytettävyydsiantuntijoiden näkemyksien mukaisesti. Näkemykset ovat toisinaan olleet ristiriitaisia. Käyttäjakeskeisyyden periaatteiden mukaisesti käyttäjät tulisi asettaa kehittämisen keskiöön ja he ovat oman työnsä parhaita asiantuntijoita, mutta käyttäjien näkemykset voivat perustua vanhoihin tietojärjestelmiin ja noudattaa esimerkiksi niistä johtuvia rajoitteita, joita ei uudessa tietojärjestelmässä välttämättä ole. Yhtenä haasteena onkin pystyä selvittämään käyttäjien tarpeita tietojärjestelmän toteutuksesta irrallisena asiana, mikä on käyttäjille haastavaa aiemmin todetun konkretian puutteen vuoksi.

Yksi suunnittelijoista (S2) toi esiin, että tietojärjestelmän käyttäjät eivät ole ikuisesti pysyvä käyttäjäkunta, joten mikäli tietojärjestelmä suunnitellaan nykyisten käyttäjien tarpeiden mukaisesti, kuinka hyvin se palvelee uusia käyttäjiä ja heidän ehkä erilaisia tarpeitaan. Käytettävyyden periaatteet ovat pysyvämpiä kuin käyttäjäkunnan tarpeet, mikä voi toimia

perusteluna käytettävyyden priorisoimiseksi korkeammalle kuin käyttäjien tarpeet. Samanlaisina toteutetut toiminnallisuudet ja yhdenmukainen logiikka kasvattavat tietojärjestelmän käytettävyyttä, mutta yhdenmukaisuutta ei välttämättä ole tehokasta noudattaa joka tilanteessa. Suunnittelija S1 toi esiin, että jotkut toiminnallisuudet on määritelty toimimaan tietyllä tavalla vain siksi, että ne olisivat yhdenmukaisia tietojärjestelmän aiemmin toteutetun toiminnallisuuden kanssa. Aina yhdenmukaisuuden toteuttaminen ei kuitenkaan tuota haluttuja hyötyjä, koska suurin osa tietojärjestelmän käyttäjistä käyttää vain tiettyjä osioita tietojärjestelmästä, joten yksittäinen käyttäjä ei välttämättä koskaan edes käytä kaikkia toiminnallisuuksia, joihin on toteutettu yhdenmukainen logiikka, jolloin hyödyt jäävät saavuttamatta. Tällaisissa tilanteissa tulisikin arvioida määrittelyratkaisuja suhteessa muuhun tietojärjestelmään.

Projektin aikana on eri aikoina painotettu eri asioita ja se, kuinka paljon tietojärjestelmän kehittäjien tulisi tietää käyttäjistä, heidän tarpeistaan ja käyttötavoistaan on vaihdellut. Aihe herätti suunnittelijoissa eri näkökulmia, mutta kukaan ei muodostanut asiasta ehdotonta mielipidettä. Suunnittelijoiden mielestä kehittäjien tulisi tietää jonkin verran käyttäjistä ja heidän käyttötavoistaan ja ennen kaikkea ymmärtää käyttäjien toiminnan taustalla oleva tarve, jotta he osaisivat esittää ratkaisuehdotuksia niihin perustuen. Toisaalta kaiken tarpeellisen tiedon pitäisi olla määrittelymateriaaleissa. Tärkeänä pidettiin myös sitä, että kehittäjä osaa asettaa toteuttamansa toiminnallisuuden yhdeksi osaksi kokonaisuutta ja hahmottavan sitä kautta sen merkityksen.

Kuviossa 10 on havainnollistettu haastateltujen tietojärjestelmän suunnittelijoiden näkemys tietojärjestelmän kehittämiseen vaikuttavista asioista, joiden prioriteettia ja painotusta tietojärjestelmän suunnittelussa joudutaan jatkuvasti arvioimaan. Arvioinnin ja suunnittelun lähtökohtaisena tavoitteena on resurssien mahdollisimman tehokas käyttö. Koska tietojärjestelmän kehittämiseen vaikuttaa käyttäjätiedon lisäksi moni muukin asia, ei käyttäjien tarpeita voi suoraan sellaisenaan toteuttaa tietojärjestelmään. Se voi aiheuttaa käyttäjille sellaisen mielikuvan, ettei tietojärjestelmän kehittämisessä olla otettu käyttäjien näkemyksiä ollenkaan huomioon,



Kuvio 10. Tietojärjestelmän kehittämisessä huomioitavat seikat

6.4 Kehittävän ylläpidon prosessikuvaus

Tutkimuksellisen kehittämistyön produktina luotiin asiakastietojärjestelmän kehittävän ylläpidon prosessikuvaus tietoperustan, ideointipajan ja haastatteluilla hankitun aineiston perusteella. Prosessikuvauksen lähtökohtana oli käyttäjakeskeisyyden lisääminen tietojärjestelmän ylläpitovaiheessa. Lähtökohta huomioitiin suunnittelemalla prosessikuvaukseen tavat, joilla käyttäjien tarpeita selvitetään ja kirjaamalla prosessikuvaukseen käyttäjakeskeisyyden periaatteet. Käyttäjakeskeisyyden suunnittelussa pyrittiin resurssien tehokkaaseen hyödyntämiseen, mutta ylläpito-organisaatiota kehoitetaan harkitsemaan muitakin opinnäytetyössä esiin tulleita tapoja käyttäjien näkemysten selvittämiseen. Käyttäjakeskeisyyden lisäksi prosessikuvauksen tavoitteena oli tehokkuus eli miten ylläpidon resurssit kohdennetaan toimenpiteisiin, jotka tuottavat eniten arvoa. Tehokkuus huomioitiin suunnittelemalla prosessikuvaukseen muutosten arviointimalli.

Kehittävän ylläpidon prosessia on kannattavaa arvioida ja kehittää iteratiivisesti sen käyttöönoton jälkeen käytännöstä saatujen kokemusten perusteella. Tulevissa muutoksissa on huomioitava prosessikuvauksen osana luodut käyttäjakeskeisyyden periaatteet ja niiden toteutuminen.

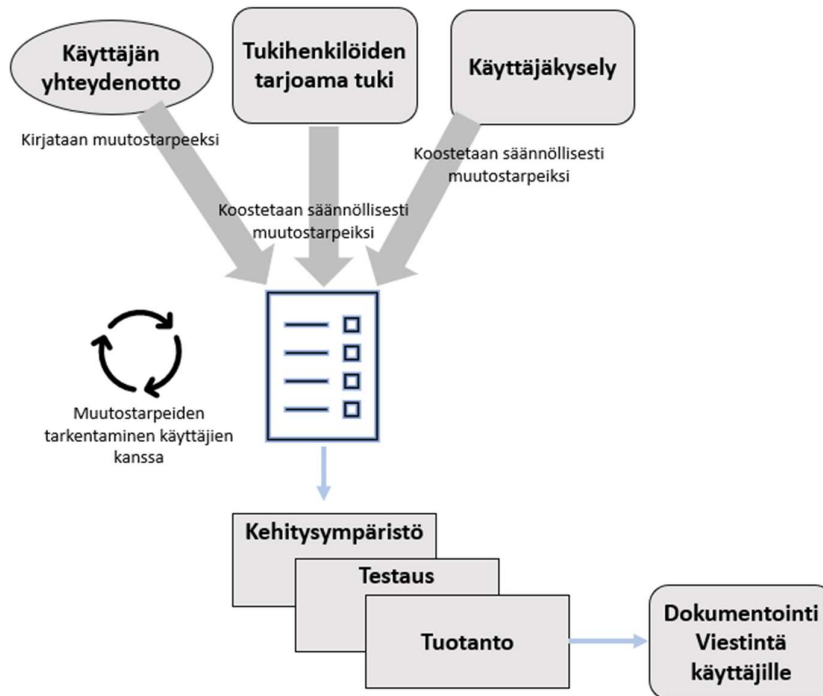
Käyttäjakeskeisen ylläpidon periaatteet:

- Käyttäjillä tulee olla mahdollisuus esittää muutostarpeita jatkuvasti ja helposti.
- Käyttäjien näkemyksiä tulee selvittää jatkuvasti ja aktiivisesti.
- Käyttäjille tulee viestiä mahdollisuuksista esittää muutostarpeita ja miten ne vaikuttavat tietojärjestelmän kehittämiseen.
- Käyttäjien esittämiä muutostarpeita ja näkemyksiä käsitellessä on ehdottomasti varmistuttava, että toteutettava muutos vastaa käyttäjien konkreettisiin tarpeisiin. Varmistaminen tehdään ensisijaisesti niin, että käyttäjät pääsevät näkemään ja

kokeilemaan muutosta ennen sen lopullista toteuttamista. Toissijaisesti yhteisestä ymmärryksestä on varmistuttava muilla tavoin.

- Käyttäjien kanssa kommunikoitaessa on varmistuttava viestin vastaanottamisesta oikean sisältöisenä. Tämä tehdään kiinnittämällä huomiota selkeään ja ymmärrettävään kieliasuun.
- Kerätyt muutostarpeet on analysoitava ja priorisoitava, jotta resurssit pystytään kohdistamaan eniten arvoa tuottaviin muutoksiin. Analysoinnissa on huomioitava laajasti kaikki muutoksen vaikutukset.

Kehittävän ylläpidon prosessikuvaus on esitetty kuviossa 11. Kehittävän ylläpidon prosessin keskiössä on muutostarvejono, joka on jatkuvasti ajan tasalla ja priorisoitu suhteutettuna käyttäjien tarpeiden ja muutoksen koon mukaisesti. Muutostarpeet koostetaan, analysoidaan ja pisteytetään myöhemmin kuvatulla tavalla. Pisteytys kuvastaa muutoksen merkitystä. Muutostarvejonosta on koko ajan nähtävissä ajantasaisesti, millä muutoksilla on korkein prioriteettiluku eli minkä toteuttamisella voidaan saavuttaa merkittävimmät hyödyt suhteessa siihen käytettyihin resursseihin. Muutostarvejono koostetaan organisaation töidenhallintajärjestelmään, josta muutokset on mahdollista nostaa helposti toteutettavaksi.



Kuvio 11. Kehittävän ylläpidon prosessikuvaus

Kuviossa 12 on kuvattu esimerkki tiedoista, jotka muutostarvelistauksen oletusnäkyessä on nähtävissä.

NIMI	VIREILLETULO-PÄIVÄMÄÄRÄ	VAIKUTUS-LUKU	LIITTYVÄT HAVAINNOT	PRIORITEETTI	STATUS
Esimerkkiosio	1.2.2022	8	12	20	Toteutettavissa
Esimerkkiosio 4	4.1.2022	15	3	18	Toteutettavissa
Esimerkkiosio 2	11.1.2022	5	2	7	Toteutettavissa
Esimerkkiosio 3	2.3.2022				Arvioitava

Kuvio 12. Muutostarvejono

Jokainen muutostarvejonon rivi edustaa muutuskorttia, josta on nähtävissä lisätietoja kyseisestä muutoksesta (kuvio 13). Muutuskortin tietoja päivitetään ja analysoidaan jatkuvasti, jotta muutostarvejono pysyy ajantasaisena.

Esimerkkiosion nimi		Perustelut/kuvaus
Vireilletulopäivämäärä	1.2.2022	
Liittyvät havainnot	12	
Käyttäjien määrä [Kuinka paljon eri käyttäjiä kyseisellä toiminnallisuudella on asteikolla 1-5]	4	Isoimman käyttäjäryhmän käytössä
Tietueiden määrä [Kuinka paljon kyseistä tietuetta käytetään asteikolla 1-5]	2	Tietueita luodaan noin 40 kuukaudessa
Toteutuksen koko [1-5*]	2	Muutos koskettaa tietueen nimeämistä sekä yhden kentän toiminnallisuutta (kopiointuminen muualle)
Vaikutusluku [edelliset kolme yhteensä]	8	
Prioriteetti [havainnot + vaikutusluku]	20	
Tila	Toteutettavissa	
Viimeisin muutospäivä	19.1.2022	
Muut ratkaisuvaihtoehdot arvioitu	19.1.2022	Tarkennettu ohjeistusta 19.1.2022.
Kuvaa lyhyesti, miten sovellus toimii tällä hetkellä		
Kuvaa, miten sovelluksen tulisi toimia		

* Arvioi toteutuksen koko asteikolla 1-5 (Isoin työmäärä on 1 ja pienin 5). Huomioi ainakin seuraavat:

- Määrittely, toteutuksen ja testauksen määrä
- Mahdolliset muutokset lokitukseen, käyttöoikeuksiin, asiakirjahallintaan ja tulosteisiin
- Mahdolliset integraatiot, vaikutukset muualle tietojärjestelmään ja muihin järjestelmiin
- Ohjeiden päivitysten ja koulutusten tarve
- Mahdollinen käyttökätkön tarve ja vaikutus kesken oleviin prosesseihin

Kuvio 13. Muutostarpeen muutuskortti

Kehittävässä ylläpidossa muutostarpeita kerätään kolmella tavalla; käyttäjän yhteydenottoina, koosteina tukihenkilöiden antamasta tuesta sekä käyttäjäkyselyinä. Kun käyttäjä ottaa yhteyttä tietojärjestelmän ylläpito-organisaatioon saadakseen apua tai

ehdottaakseen muutosta tietojärjestelmään, kirjataan käyttäjän yhteydenotto muutostarpeena organisaation töidenhallintajärjestelmään. Muutostarpeesta luodaan muutuskortti, johon kerätään muutoksen analysoinnissa tuotettu tieto. Mikäli samasta asiasta on jo muutuskortti, liitetään uusi muutostarve olemassa olevaan muutuskorttiin. Yhteydenottotavan ylläpito-organisaatioon tulisi olla mahdollisimman helppo ja ratkaisuksi suositellaan esimerkiksi suoraa linkkiä yhteydenottojärjestelmään tietojärjestelmästä.

Tietojärjestelmän tukihenkilöt toimivat käyttäjien ensimmäisen asteen tukena. He dokumentoivat antamaansa tukea ja raportoivat säännöllisesti ylläpito-organisaatiolle tilastoja antamastaan tuesta. Tukihenkilöiden antama tuki koostetaan säännöllisesti olemassa oleviin muutuskortteihin tai ne luodaan uusina. Kehittävän ylläpidon vastuulla on toteuttaa säännöllisesti käyttäjäkysely, jonka avulla kerätyt tiedot myös koostetaan säännöllisesti olemassa oleviin muutuskortteihin tai ne luodaan uusina.

Muutuskortin avulla voidaan koostaa muutostarpeen prioriteetti. Kun samaan aiheeseen tai aihepiiriin tulee uusi muutostarve, liitetään se olemassa olevaan muutostarpeeseen eli samanlaiset tai samankaltaiset muutostarpeet ryhmitellään tehokkuuden maksimoimiseksi. Liittyviin havaintoihin liitetään myös osion käytöstä aiheutuneet palvelupyynnöt eli kaikki tilanteet, jotka osion käytössä ovat aiheuttaneet ylimääräistä korjaus- tai käsittelytyötä. Näihin lasketaan esimerkiksi tilanteet, jossa tietojärjestelmään tallennettua virheellistä tietoa on jouduttu korjaamaan. Näin muutuskortin liittyvien havaintojen lukumäärä kasvaa. Muutostarpeen liitos aiempaan muutostarpeeseen on riippuvainen siitä, liittyvätkö muutokset täysin samaan aiheeseen tai onko ne muutoin tehokasta toteuttaa samanaikaisesti.

Ennen muutoksen luomista tai liittämistä olemassa olevaan muutokseen, on tärkeää varmistaa, että muutos ja tarve sen taustalla on ymmärretty oikein kommunikoimalla muutoksen esittäneen käyttäjän kanssa. Tärkeä on myös varmistaa, että muutos sisältää kaiken tarvittavan tiedon. Samoin muutoksia analysoitaessa on tärkeää jatkuvasti tarkentaa muutostarpeita käyttäjien kanssa, jotta voidaan varmistua, että tietojärjestelmään toteutetaan toivottuja muutoksia eikä niitä ole tarve myöhemmin muuttaa.

Kun muutuskortti on luotu muutostarvejonoon, on muutos analysoitava prioriteetin arvioimiseksi. Asiakastietojärjestelmän käytön asiantuntijat arvioivat, kuinka monta käyttäjää muutoksen kohteena olevalla osiolla on sekä kuinka paljon kyseistä toiminnallisuutta käytetään molemmat asteikolla 1-5. Asiakastietojärjestelmän kehittämisen asiantuntijat arvioivat toteutuksen kokoa myös asteikolla 1-5 niin, että isoin työmäärä on 1 ja pienin 5. Toteutuksen kokoa arvioitaessa on muutosta arvioitava

mahdollisimman kattavasti. Kokoa arvioitaessa on otettava huomioon ainakin toteutuksen määrittely, toteutus sovellukseen, toteutuksen testaus, muutokset lokitukseen, käyttöoikeuksiin, asiakirjahallintaan ja tulosteisiin sekä ohjeiden päivitysten ja käyttäjäkoulutuksen tarve. Lisäksi toteutuksen arvioinnissa on huomioitava mahdolliset muutokset muualle tietojärjestelmään, mahdolliset integraatiot ja muutokset muihin tietojärjestelmiin, käyttökatkon tarve ja vaikutukset sekä miten käsitellään muutoksen kohteena olevat keskeneräiset prosessit. Analysoitaessa muutosta on myös arvioitava, pystytäänkö käyttäjien tarpeisiin vastamaan jollain muulla tavalla kuin sovellusmuutoksella, esimerkiksi ohjeistuksella, koulutuksella tai sovelluksen ohjaustietojen muutoksella.

Laskemalla yhteen liittyvien havaintojen lukumäärän, käyttäjien määrän arvion luvun, tietueiden määrän arvion luvun sekä toteutuksen koon luvun saadaan muutostarpeelle prioriteetiluku. Mitä korkeampi prioriteetti on, sitä hyödyllisempi muutos on.

Käyttäjien määrän, tietueiden määrän ja toteutuksen koon arviointiin voidaan käyttää erilaisia tietojärjestelmän käyttöön perustuvia tilastoja. Niiden käytön edellytyksenä on kuitenkin niiden helppo saatavuus, jotta muutoksen arviointiin ei käytetä liikaa resursseja suhteessa toteutettavan muutoksen kokoon. Vaikka esimerkiksi tietueiden määrän arviointiin käytettäisiin tietojärjestelmästä saatavaa tilastotietoa, tulee tietojärjestelmän asiantuntijoiden määrittellä muutuskortin lukumääriä vastaavat tilastoarvot eli kuinka monta tietuetta vastaa lukuja 1-5. Kaikki muutuskortin luvut ja erityisesti toteutuksen koko perustuu kuitenkin asiantuntijoiden tietoon ja kykyyn arvioida yksittäistä toteutettavaa muutosta ja sen eri osia ja vaikutuksia. Tästä syystä arvioinnin tuottamaa prioriteetilukua ja niiden myötä luotavaa prioriteettijärjestystä tulisi tarkastella ehdotuksena, jota voidaan käyttää päätöksenteon tukena eikä ehdottomana toteuttamisjärjestyksenä. Erityisesti muutokset, joiden prioriteetilukujen erot ovat pienet, tulisi tarkastella osana laajempaa kokonaisuutta. Muutostarvejonon muutosten prioriteettia on arvioitava myös suhteessa muiden ylläpidon osa-alueiden tehtäviin.

Kun on esitetty, että tietojärjestelmän tiettyyn osioon tehdään muutos, voidaan muutoksen määrittely aloittaa tarkentamalla haluttu tavoitetilä tai haasteellinen lähtötilä. Mikäli muutoksen määrittely aloitetaan kuvaamalla tavoitetilä, voi olla hyödyllistä spesifioida tavoitetilä eli käydä läpi tavoitetilän spesifointiprosessi, jotta vältytään väärinymmärryksiltä ja tietojärjestelmän kehittämisessä voidaan varmistua siitä, että kehittämisessä kehitetään oikeita asioita. Tavoitetilän spesifointiprosessilla voidaan välttyä käyttämästä resursseja muutoksiin, jolla ei saavuteta tavoiteltua lopputulosta. Mikäli muutoksen määrittely aloitetaan kuvaamalla lähtötilä, on tavoitetilä mahdollista kuvata peilaten lähtötilän

haasteisiin. Mahdollisuuksien mukaan suunnittelun yhteydessä toteutetaan myös protoilua. Jo ennen päätöksentekoa on oleellista, että muutuskorttiin on kuvattu, mitä tietojärjestelmän osiota muutos koskee ja mikä on tarve muutoksen taustalla. Tämä tehdään kommunikoimalla käyttäjien kanssa jatkuvasti.

Jonosta voidaan ennalta sovituin väliajoin viedä organisaatorakenteen mukaiseen foorumiin päätettäväksi toteuttavat muutokset. Muutostarvejonoon on koostettuna ainoastaan käyttäjien tarpeiden perusteella havaitut muutostarpeet. Käyttäjien tarpeisiin perustuvien muutosten lisäksi sovellukseen voidaan tehdä ainakin virheitä korjaavia muutoksia ja organisaatiosta lähtöisin olevia muutoksia, jotka mahdollistavat tietojärjestelmän käytön ja tietojärjestelmästä saatavan tiedon käytön organisaation tahtotilan mukaisesti. Näiden tarpeiden priorisointi tulee tehdä suhteessa käyttäjien tarpeisiin. Lopulta organisaatorakenteen mukaisessa foorumissa tehdään päätös siitä, toteutetaanko toivottu muutos. Mikäli päätös on, että toiminnallisuutta ei toteuteta, on toiminnallisuuden suunnitelman oltava kuitenkin niin jäsentynyt ja yksityiskohtainen, että se voidaan mahdollisesti toteuttaa myöhemmin. Myös toteuttamatta jätetyt muutokset pidetään tallennettuna töidenhallintajärjestelmässä mahdollista myöhempää toteutusta varten. Muutuskorttiin kirjataan selkeästi, milloin ja millä perusteella on tehty päätös toteuttamatta jättämisestä.

Mikäli toiminnallisuuden muutokseen päädytään, se toteutetaan. Toteutus testataan ja viedään tuotantoympäristöön organisaation ylläpitomallin mukaisesti. Toteutuksen yhteydessä tehdyt muutokset päivitetään tarpeellisiin dokumentteihin. Käyttäjille on myös tärkeää viestiä, millaisia muutoksia tietojärjestelmään on tehty ja mitkä niistä on tehty käyttäjien tarpeisiin perustuen.

7 Yhteenveto ja pohdinta

Opinnäytetyön tavoitteena oli tuottaa tutkimuksellisen kehittämistyön produktina Rikosseuraamuslaitoksen asiakastietojärjestelmän kehittävän ylläpidon organisointia kuvaava prosessikuvaus. Prosessikuvaus laadittiin, mutta sitä ei ehditty opinnäytetyöprosessin aikana ottaa käytäntöön kohdeorganisaatiossa organisaation omien aikataulujen vuoksi. Prosessikuvaus on laadittu käytettäväksi kohdeorganisaatiossa, mutta osaksi prosessikuvausta laadittiin käyttäjakeskeisyyden toteuttamiseen liittyviä periaatteita, jotka ovat yleistettävissä muihin tietojärjestelmiin, joissa käyttäjakeskeisyyteen halutaan kiinnittää huomiota.

7.1 Lopputulokset

Opinnäytetyön tietoperustan ja tietojärjestelmän suunnittelijoiden ja käyttäjien haastatteluiden perusteella olennaisinta ylläpidon organisoimiseksi käyttäjakeskeisesti on käyttäjakeskeisyyden tarpeen tunnistaminen ja sen toteuttaminen jo ennen ylläpitovaihetta. Samoin jotta ylläpidon resurssit voidaan kohdentaa tehokkaasti, tulee tarve sille ottaa huomioon jo kaikissa vaiheissa ennen ylläpitoa. Parhaiten resurssien tehokas kohdentaminen varmistetaan huolehtimalla tietojärjestelmän ylläpidettävyydestä ja dokumentoinnista sekä organisoimalla ylläpidon resurssit suunnitelmallisesti ja tehokkaasti. Kun tietojärjestelmän ylläpidettavuus on korkealla tasolla, on tietojärjestelmän käyttäjakeskeinen kehittäminen ylläpitovaiheessa helpompaa toteuttaa pienemmillä resursseilla. Siksi on tärkeää jo suunnitteluvaiheessa pystyä tunnistamaan osat, joihin todennäköisimmin tulee kohdistumaan muutoksia.

Kehittämällä tietojärjestelmää alusta alkaen käyttäjakeskeisesti voidaan varmistaa sen jatkuminen myös ylläpitovaiheessa. Käyttäjakeskeisyyden tarpeen tunnistaminen vasta ylläpitovaiheessa voi aiheuttaa suuria ongelmia tietojärjestelmän käyttöönotossa ja käytössä. Kun tietojärjestelmää kehitetään käyttäjakeskeisesti, palvelee kehittämisen aikana kertynyt käyttäjätieto myös ylläpitovaiheen kehitystä. Kun käyttäjiä osallistetaan tietojärjestelmän kehittämiseen alusta alkaen, luodaan heille luottamus siihen, että heidän näkemyksiään arvostetaan ja niillä on merkitystä. Käyttäjakeskeisyyden haasteeksi nousee käyttäjien tarpeiden sanoittaminen riittävän tarkalle tasolle sekä ajoituksen haastavuus.

Mikäli käyttäjien näkemyksiä selvitellään ennen kuin mitään konkreettista on toteutettu, voivat vaatimukset/näkemykset olla helposti liian yleisluontoisia ja käyttäjät eivät välttämättä edes hahmota, mitä heiltä toivotaan tai mitä he haluaisivat tietojärjestelmältä. Jos näkemyksiä kysytään siinä vaiheessa, kun sovelluksesta on käyttäjälle jotain

konkreettista näytettävää, pystyy käyttäjä helpommin sanomaan, onko tuote hänen mielestään tarkoituksenmukainen ja millaisia muutoksia siihen mahdollisesti pitäisi tehdä, mutta jo toteutetun sovelluksen purkamiseen tai muuttamiseen on korkeampi kynnyks, jos resurssit ja aikataulut ovat tiukat. Käyttäjä ei välttämättä osaa kuitenkaan sanoittaa toiveitaan, ennen kuin pääsee itse kokeilemaan tietojärjestelmää. Onkin kannattavaa selvittää, onko protoilu jollain tasolla resurssien puitteissa mahdollista, sillä sen avulla voidaan saavuttaa hyviä tuloksia ja saada käyttäjien tarpeet tarpeeksi ajoissa muutettua tarkoiksi vaatimuksiksi. Sekä vaatimusten ajoituksen että tarkkuuden haasteisiin tulisi löytää ratkaisu, jolla resurssien puitteissa mahdollistetaan tarpeeksi tarkan tason protoilu käyttäjille ja mahdollisuus antaa siitä palautetta.

Myös ylläpidon resurssien tehokas kohdentaminen on haastavaa, jos ylläpidon suunnitteluun havahdutaan vasta ylläpitovaiheen kynnyksellä. Tietojärjestelmän aikaisemmissa vaiheissa voidaan monin tavoin varmistaa ylläpitovaiheen kustannustehokkuus ja tietojärjestelmän pitkäikäisyys, mikäli ylläpitovaiheen tarpeet osataan priorisoida aikaisemmissa vaiheissa. Panostus ylläpidettävyystekijöihin maksaa itsensä monin verroin takaisin.

Käyttäjakeskeisyyden toteuttamiseen sekä tietojärjestelmän ylläpidettävyyteen vaikuttaa positiivisesti koko tietojärjestelmän kuvaaminen ja kokonaisuuden hallinta hyvin varhaisessa vaiheessa tietojärjestelmän kehittämistä. Ennen kuvaamista on oltava tiedossa organisaation haluttu tahtotila, jota tukemaan tietojärjestelmä suunnitellaan. Kokonaisuuden kuvaaminen mahdollistaa sen, että tietojärjestelmästä rakennetaan yhtenäinen kokonaisuus, jonka eri osat ovat loogisesti ja tarpeellisessa määrin integroituneita ja jonka prosessit on suunniteltu alusta loppuun huomioiden kaikki erilaiset käyttötapaukset ja tehtävien vastuunjako sekä sovelluksessa että sen ulkopuolella. Selkeä kokonaisuus auttaa myös eri toiminnallisuuksien priorisoinnissa ja resurssien kohdentamisessa oikeassa suhteessa tavoiteltuihin hyötyihin. Kokonaisuuden suunnitteluun tulee varata resursseja ja sen merkitys huomioitava valitusta kehitystavasta riippumatta.

Opinnäytetyön produktina on tuotettu kehittävän ylläpidon prosessikuvaus, jonka keskiössä on jatkuvasti ajan tasalla oleva priorisoitu muutostarvejono. Prosessikuvauksessa on huomioitu tietoperustasta ja haastatteluilla hankitusta aineistosta esiin nousseet huomiot; käyttäjillä tulee olla mahdollisuus esittää muutostarpeita jatkuvasti ja niitä pitää myös jatkuvasti aktiivisesti kysellä. Prosessikuvauksessa on huomioitu myös kehittävän ylläpidon resurssien mahdollisimman tehokas käyttö sekä käyttäjätiedon suhde muihin tietojärjestelmään kohdistuviin tarpeisiin. Nämä seikat liittyvät tyypillisesti julkisen

hallinnon tietojärjestelmän erityispiirteisiin. Prosessikuvauksen etuina voidaan pitää sen käyttäjäkeskeisyyttä. Prosessikuvauksessa on kuvattu eri tapoja selvittää käyttäjien näkemyksiä ja niissä on huomioitu myös käyttäjien motivaatio osallistua kehittämiseen.

Kehittämistyön lopputuloksena luodun kehittävän ylläpidon prosessikuvauksen käyttöönotto ja kehittäminen käytön kokemusten perusteella voisi olla aiheena jatkotutkimukselle. Muita mahdollisia aiheita kohdeorganisaatiossa olisivat tietojärjestelmän virheiden ja muutostarpeiden määrien kehitys käyttöönoton jälkeen sekä käytettävyyden tai käyttäjien tyytyväisyyden kehitys ja miten siihen voi vaikuttaa. Yleisesti olisi mielenkiintoista tutkia, miten paljon tietojärjestelmän toteuttajan tulee ymmärtää toteuttamastaan sovelluksesta ja sen käyttöympäristöstä, jotta tiedolla olisi merkittäviä hyötyjä toteuttamisessa.

7.2 Pohdinta

Tutkimuksellisen kehittämisen lähestymistavaksi valittiin konstrukttiivinen tutkimus, jossa ratkaisun luomiseen käytetään teorian ja empiirisen tiedon yhdistelmää. Tutkimuksen tietoperustan rakentaminen ei aiheuttanut juurikaan haasteita aiheesta olemassa olevan kirjallisuuden ja tutkimustiedon runsauden vuoksi. Tutkimuksen tutkimuskysymysten muotoilu lopulliseen muotonsa vaati useamman iteraatiokierroksen ja sen vuoksi tietoperustaa muotoiltiin ja jäsenneltiin tukemaan paremmin tutkimusprosessia. Tietoperustan rajaamisella pyrittiin löytämään oikea näkökulma tutkittavaan aiheeseen. Ainoastaan kirjallisuuden ja tutkimustiedon löytäminen käyttäjäkeskeisyydestä ja käyttäjätiedon käyttämisestä ylläpitovaiheessa tuotti alkuun haasteita. Kirjallisuuden vähäisyys voi selittyä sillä, että käyttäjäkeskeisyyden perusta rakennetaan jo ylläpitoa aikaisemmissa vaiheissa, mikä myös opinnäytetyön lopputuloksissa todetaan. Ylläpitovaihe ei useimmiten myöskään ole nykypäivän tietojärjestelmissä enää oma erillinen vaiheensa, vaan kehitystä ja ylläpitoa tehdään rinnakkain. Tietoperustan muodostamista hankaloitti käyttäjäkeskeisyyteen liittyvien samankaltaisten käsitteiden joukko, josta lopulta tietoperustan käsittelyn jälkeen opinnäytetyössä käytettäväksi käsitteeksi valittiin käyttäjäkeskeisyys.

Tutkimuksen kehittämistyö toteutettiin hyödyntäen yhteisöllisiä ja kvalitatiivisia menetelmiä. Tutkimuksen luotettavuus eli mahdollisimman korkea reliabiliteetti ja validiteetti pyrittiin varmistamaan valitsemalla tutkimuksen kannalta oleelliset tutkimusmenetelmät, perustelemalla tehdyt valinnat ja raportoimalla tutkimus niin tarkasti, että se olisi raportin perusteella mahdollista toteuttaa uudestaan. Raporttiin dokumentoitiin systemaattisesti ja yksityiskohtaisesti tutkimuksen kaikki vaiheet, aineiston tuottamisen olosuhteet sekä mihin tulosten tulkinta on perustunut. Tulkintojen perusteeksi haastattelut

litteroitiin ja raporttiin lisättiin suoria lainauksia haastateltavilta. Opinnäytetyön luotettavuutta voidaan arvioida arvioimalla sen tulosten hyödynnettävyyttä. Vaikka opinnäytetyön produktia ei prosessin aikana ehditty ottaa käyttöön ja arvioida sen käytettävyyttä, on opinnäytetyö tuottanut tietojärjestelmän ylläpidosta vastaavalle organisaatiolle huomioon otettavia periaatteita ja valmiin käyttöön otettavan prosessikuvauksen. Tulosten yleistettävyyttä on edistetty tuomalla esiin uusia, viitekehysten avulla perusteltuja näkökulmia.

Kehittämistyön aineisto hankittiin haastattelemalla asiakastietojärjestelmän suunnittelijoita ja käyttäjiä. Opinnäytetyön tekijä on itse työskennellyt kyseisen asiakastietojärjestelmän suunnittelijana toimien itse myös yhtenä aineiston lähteenä. Tekijän aikaisempi rooli saattoi vaikuttaa myös haastateltaviin ja hankittuun aineistoon. Koska haastattelujen aihe liittyi haastattelijan aikaisempaan työhön, saattoi se aiheuttaa haastateltavissa erilaisia tulkintoja viitekehuksesta ja haastattelijan tarkoituksesta ja sitä myöten tarvetta kertoa kokemuksistaan sosiaalisesti hyväksytyllä tavalla todellisuutta positiivisemmin. Käyttäjakeskeisyyden käsite ei ollut kaikille haastateltaville entuudestaan tuttu ja nopea tutustuminen uuteen aiheeseen saattoi aiheuttaa myös aiheen erilaisia tulkintoja. Haastattelujen aikana aihetta tarkennettiin ja selkiytettiin yhteisen ymmärryksen aikaansaamiseksi. Haastateltavana oli tietojärjestelmän suunnittelijoita ja käyttäjiä, joilla on erilaiset taustatiedot, lähtökohdat ja näkökulmat käsiteltävään aiheeseen. Näkemykset eri näkökulmista rikastavat aiheen käsittelyä ja niitä vertailtiin aineiston analyysivaiheessa.

Tutkimuksen aineisto analysoitiin teoriaohjaavasti, jossa aikaisempi tieto ohjaa analyysiyksiköiden valintaa ja näin teoria toimii analyysin apuna avaten uusia ajatusuria. Teoriaohjaavassa analyysissä tutkijan päättelyssä yhdistyvät aikaisemmat mallit ja aineistolähteisyys, jolla pyrittiin varmistamaan, että analyysia ohjaa aineisto eikä tutkija. Opinnäytetyön tekijän aikaisempi rooli vaikutti mahdollisesti myös aineiston analyysiin, vaikka siinä pyrittiin objektiiviseen suhtautumiseen. Tutkittaessa ihmisiä ei tutkija voi toimia täysin ulkoisena tarkkailijana, vaan tutkijan oma tausta vaikuttaa aina jonkin verran tutkimukseen. Opinnäytetyön tekijä on pyrkinyt tarkastelemaan kohdetta järjestelmällisesti ja johdonmukaisesti samalla tiedostaen oman historiansa ja sen mahdollisen vaikutuksen tulkintaan. Tekijä on toiminut myös aineiston lähteenä eli hänen ammatillista näkemystään ei ole ollut tarkoitus sivuuttaa kokonaan, vaan tiedostaa sen vaikutus tulkintaan ja sen kautta tapahtuvaan ymmärrykseen. Aineiston analyysi ei tuottanut teoriasta merkittävästi poikkeavia tuloksia käyttäjakeskeisyyteen liittyen.

Konstruktiiivinen tutkimus sopi hyvin kehittämistyön lähestymistavaksi. Uuden prosessikuvauksen pohjaksi tarvittiin ehdottomasti teoreettista tietämistä, mihin vastattiin

tietoperustan rakentamisella. Teoreettisen tietämyksen lisäksi prosessikuvauksen luomiseen käytettiin tietojärjestelmän suunnittelijoilta ja käyttäjiltä hankittua empiiristä tietoa, joka yhdistettiin tietoperustaan. Konstruktioivinen tutkimus pyrkii ratkaisemaan ongelmia käytännönläheisesti ja luotu prosessikuvaus vastaa kohdeorganisaation ongelmaan kehittävän ylläpidon organisoimisesta tehokkaasti ja käyttäjäkeskeisesti. Opinnäytetyöprosessin aikana ei pystytty toteuttamaan konstruktioiviseen tutkimukseen kuuluvaa ratkaisun toimivuuden testausta ja konstruktion oikeellisuuden osoittamista. Jo lähestymistapaa valitessa oli tiedossa, että konstruktion tieteellisen annin osoittaminen olisi vaatinut merkittävästi lisätöitä, mutta opinnäytetyön alkuperäisenä tavoitteena oli myös arvioida kehitystyön tuloksena luotu malli sen tuottamien käytännön hyötyjen perusteella, joka nyt jäi toteutumatta.

Lähteet

- Alahuhta, M. 2015. Johtajuus. Kirkas suunta ja ihmisten voima. Docendo. Jyväskylä.
- Etelä-Suomen aluehallintovirasto. 2021. Yleistä saavutettavuudesta. Luettavissa: <https://www.saavutettavuusvaatimukset.fi/yleista-saavutettavuudesta/>. Luettu: 11.12.2021.
- Forselius, P. 2013. Onnistunut tietojärjestelmän hankinta. Talentum. Helsinki.
- Günther, K., Hasanen, K. & Juhila K. 2021. Johdanto: Analyysi ja tulkinta. Teoksessa Vuori, J. (toim.) Laadullisen tutkimuksen verkkokäsikirja. Tampere: Yhteiskuntatieteellinen tietoarkisto. Luettavissa: <https://www.fsd.tuni.fi/fi/palvelut/menetelmaopetus/kvali/analyysitavan-valinta-ja-yleiset-analyysitavat/analyysi-ja-tulkinta/>. Luettu: 26.11.2021.
- Harsu, M. 2003. Ohjelmien ylläpito ja uudistaminen. Talentum. Helsinki.
- Hirsjärvi, S., Remes, P. & Sajavaara, P. 2007. Tutki ja kirjoita. Tammi. Helsinki.
- Hyysalo, S. 2009. Käyttäjä tuotekehityksessä. Tieto, tutkimus ja menetelmät. Taideteollisen korkeakoulun julkaisu. Helsinki.
- Hyysalo, S. 2011. Käyttäjätieto ja teknologian sosiaalinen muotoutuminen. Teoksessa Oulasvirta, A. (toim.) Ihmisen ja tietokoneen vuorovaikutus, s. 127-152. Gaudeamus Helsinki University Press. Helsinki.
- Interaction design foundation. Adopter Categories for New Products. Luettavissa: <https://www.interaction-design.org/literature/topics/adopter-categories-for-new-products>. Luettu: 26.12.2021.
- Jokela, T. 2010. Navigoi oikein käytettävyyden vesillä: Opas käytettävyysohjattuun vuorovaikutussuunnitteluun. Väylä-Yhtiöt Oy. Pello.
- Juhila K. 2021. Teemoittelu. Teoksessa Vuori, J. (toim.) Laadullisen tutkimuksen verkkokäsikirja. Tampere: Yhteiskuntatieteellinen tietoarkisto. Luettavissa: <https://www.fsd.tuni.fi/fi/palvelut/menetelmaopetus/kvali/analyysitavan-valinta-ja-yleiset-analyysitavat/teemoittelu/>. Luettu: 8.8.2021.
- Juvonen, R. 2018. Ohjelmistoprojektin sudenkuopat ja miten ne vältetään. Books on Demand. Helsinki.
- Järvinen, P. & Järvinen, A. 2011. Tutkimustyön metodeista. Opinpajan kirja. Tampere.

Kananen, J. 2015. Opinnäytetyön kirjoittajan opas. Näin kirjoitan opinnäytetyön tai pro gradun alusta loppuun. Jyväskylän ammattikorkeakoulu. Jyväskylä.

Kankaanpää, S. 18.1.2021. Menetelmiä ja työkaluja fasilitoinnin tueksi. Luettavissa: <https://lamia.fi/blogi/menetelmia-ja-ty%C3%B6kaluja-fasilitointiin>. Luettu: 25.11.2021.

Koistinen, H. 2002. Tietojärjestelmien ylläpito. Talentum. Helsinki.

Koskela, J. 22.11.2017. Saavutettavuus, siis mitä? Luettavissa: <https://www.celia.fi/Blogi/saavutettavuus-siis-mita/>. Luettu: 11.12.2021.

Lehtimäki, T. 2006. Ohjelmistoprojektit käytännössä. Readme.fi. Helsinki.

Malkavaara, H-L. 2021. Coaching ja fasilitoinnin taito -opintojakson kurssimateriaali. Haaga-Helia ammattikorkeakoulu.

Merisalo-Rantanen, H. 2011. Management of open-ended user feedback in the continuous development of information systems and e-services. Aalto University. Espoo. Luettavissa: https://aaltodoc.aalto.fi/bitstream/handle/123456789/8903/Aalto_DD_2011_008.pdf?sequence=1&isAllowed=y. Luettu: 8.8.2021.

Ojasalo, K., Moilanen, T. & Ritalahti, J. 2014. Kehittämistyön menetelmät. Uudenlaista osaamista liiketoimintaan. Sanoma Pro. Helsinki.

Oulasvirta, A. 2011. Mitä on ihmisen ja tietokoneen vuorovaikutus? Teoksessa Oulasvirta, A. (toim.) Ihmisen ja tietokoneen vuorovaikutus, s. 13-42. Gaudeamus Helsinki University Press. Helsinki.

Pohjonen, R. 2002. Tietojärjestelmien kehittäminen. Docendo Finland Oy. Jyväskylä.

Ries, E. 2011. Lean startup – kokeilukulttuurin käsikirja. The Crown publishing group. New York.

Rikosseuraamuslaitos. 2019. Historiaa. Luettavissa: <https://www.rikosseuraamus.fi/fi/index/rikosseuraamuslaitos/historiaa.html>. Luettu: 26.12.2021

Rikosseuraamuslaitos. 2020. Roti – toiminnan kehittämis- ja asiakastietojärjestelmähanke. Luettavissa: <https://www.rikosseuraamus.fi/fi/index/ajankohtaista/hankkeet/roti.html>. Luettu: 26.12.2021.

Saariluoma, P., Kujala, T., Kuuva, S., Kymäläinen, T., Leikas, J., Liikkanen, L. & Oulasvirta, A. 2010. Ihminen ja teknologia: Hyvän vuorovaikutuksen suunnittelu. Teknologiateollisuus ry. Helsinki.

Saariluoma, P. 2011. Käyttäjä. Teoksessa Oulasvirta, A. (toim.) Ihmisen ja tietokoneen vuorovaikutus, s. 45-61. Gaudeamus Helsinki University Press. Helsinki.

Savolainen, K. 2021. Human-Centred Design When Direct Contact with Users Is Not Possible. Luettavissa:

<https://aaltodoc.aalto.fi/bitstream/handle/123456789/107543/isbn9789526403519.pdf?sequence=4&isAllowed=y>. Luettu: 12.12.2021.

Sinkkonen, I., Kuoppala, H., Parkkinen, J. & Vastamäki, R. 2006. Käytettävyyden psykologia. Edita Publishing Oy. Helsinki.

Stenberg, M. 2006. Tieto. Tietojohtamisen arkkitehtuurit. Otava. Helsinki.

Tietosuojavaltuutetun toimisto. Usein kysyttyä EU:n tietosuojasetuksesta. Luettavissa: <https://tietosuoja.fi/gdpr>. Luettu: 11.12.2021.

Tuomi, J. & Sarajärvi, A. 2018. Laadullinen tutkimus ja sisällönanalyysi. Tammi. Helsinki.

Tutki hankintoja. 2021. Hankintakategoriat. Luettavissa:

<https://tutkihankintoja.fi/hankintakategoriat?endDate=31.12.2020&startDate=01.01.2020&lang=fi>. Luettu: 22.11.2021.

Valtiovarainministeriö. Saavutettavuus. Luettavissa: <https://vm.fi/saavutettavuusdirektiivi>. Luettu: 11.12.2021.

Vesselov, S. & Davis, T. 2019. Building design systems. Unify user experience through a shared design language. Apress. New York.

Väänänen-Vainio-Mattila, K. 2011. Käytettävyys ja käyttäjäkeskeinen suunnittelu.

Teoksessa Oulasvirta, A. (toim.) Ihmisen ja tietokoneen vuorovaikutus, s. 102-126. Gaudeamus Helsinki University Press. Helsinki.

Liitteet

Liite 1. Suunnittelijan haastattelurunko

Suunnittelijan haastattelurunko

- Miten käyttäjäkeskeisyys on mielestäsi toteutunut tietojärjestelmässä? Onko se toteutunut riittävästi?
 - o Mitkä ovat olleet onnistuneita ratkaisuja käyttäjäkeskeisyyden toteuttamisessa?
- Miten suhtaudut väitteeseen, ettei käyttäjiä ole kuunneltu riittävästi tietojärjestelmää suunniteltaessa? Mitä ajatuksia se herättää?
- Mikä on ollut vaikeinta tietojärjestelmän kehittämisessä?
- Mikä on ollut vaikeinta käyttäjäkeskeisessä suunnittelussa?
- Onko käyttäjien toimiminen suunnittelijoina ollut onnistunut ratkaisu?
- Määrittelyssä on ollut mukana myös käytettävyyssiantuntijoita. Käytettävyyssiantuntijoiden ja suunnittelijoiden (käyttäjien) näkemykset tietojärjestelmän toiminnallisuuksista eivät ole aina olleet samassa linjassa. Pystytkö arvioimaan, kumpi tieto on tärkeämpää tai kumman näkemyksien mukaan suunnittelussa tulisi edetä?
- Kuinka paljon koet, että kehittäjän tulee ymmärtää käyttäjää ja hänen toimintaansa tai kuinka syvällistä tietoa kehittäjällä pitää olla käyttäjän toiminnasta?
- Kuinka paljon koet, että tietojärjestelmämme käyttäjissä on eroja esim. käyttötapojen tai toimintatapojen osalta? Miten se on huomioitu?
- Jälkikäteen ajateltuna, mitä asioita olisit tehnyt toisin?
- Miten käyttäjäkeskeisyys tulisi toteutua ylläpitovaiheessa?

Liite 2. Käyttäjien haastattelurunko

Käyttäjän haastattelurunko

- Millaista työtä teet?
- Kuinka paljon olet osallistunut tietojärjestelmään liittyviin tilaisuuksiin tai koulutuksiin?
- Kuinka paljon olet itse kokeillut tietojärjestelmän käyttöä?
- Millaisia ajatuksia tietojärjestelmä tällä hetkellä herättää?
- Mikä tietojärjestelmän käytössä tai opettelussa on tuntunut haasteellista? Voi sanoa useamman asian.
 - o Itse tietojärjestelmän toiminnallisuudet
 - o Muu liittyvä toiminta (opiskelu, ohjeet jne)
- Mikä tietojärjestelmän toiminnallisuuksista on mielestäsi onnistunut tai hyvin toteutettu?
 - o Helppokäyttöinen toiminnallisuus?
 - o Muu toiminta?
- Miten tietojärjestelmän käyttöönottoa voitaisiin helpottaa/parantaa?
- Kuinka paljon koet, että käyttäjät on otettu huomioon tietojärjestelmää suunnitellessa?
- Miten käyttäjät kannattaisi huomioida? Mikä on sellaista, mitä tietojärjestelmän käyttäjistä pitäisi ymmärtää tai tietää?
- Kuvitellaan tietojärjestelmän käytössä tilanne, jossa tiedät, miten sinun pitäisi toimia ja osaat käyttää sitä, mutta mieleesi tulee jokin asia, miten tietojärjestelmä voisi toimia järkevämmiin tai tehokkaammin, niin mikä motivoi kertomaan sen ajatuksen eteenpäin ylläpito-organisaatioon?
- Miten käyttäjien näkemyksiä kannattaisi selvittää? Mitä tulisi ottaa huomioon?
- Miten itse haluaisit osallistua tietojärjestelmän kehittämiseen?
 - o Kysely, haastattelu, keskustelutilaisuus?
- Yleisellä tasolla kuinka helposti opit käyttämään uusia järjestelmiä tai palveluita?

Liite 3. Haastateltavat

Salattu liite. Haastateltavien nimiä ei anonyymisistä julkaista.

Liite 4. Teemat käyttäjakeskeisyyden toteutuminen ennen ylläpitovaihetta ja tietojärjestelmän monimutkaisuus haastatteluissa

Haastateltava	Käyttäjakeskeisyyden toteutuminen ennen ylläpitovaihetta	Tietojärjestelmän monimutkaisuus
H1	<p>Ollut mukana testaamassa ja sillä tavoin päässyt kertomaan näkemyksiään, vaikka toiveita ja huomioita ei ole aina voinut aikataulun vuoksi ottaa huomioon enää ennen käyttöönottoa.</p> <p>Luottaa siihen, että ammattilaiset ovat olleet suunnittelemassa ja tekemässä tietojärjestelmää, jonka kokee olevan todella monimutkainen ja iso kokonaisuus. On myös havahtunut ymmärtämään, miten monet muut asiat käyttäjien toiveiden lisäksi vaikuttavat tietojärjestelmän kehittämiseen ja toiminnallisuuksiin, esimerkiksi toimintatapojen muutokset.</p>	<p>Vaikeinta on navigointi sovelluksessa ja linkkien löytäminen. Sovelluksen tai asiakkaan kokonaiskuva on vaikea hahmottaa.</p> <p>Samanlaisina toteutetut toiminnallisuudet ovat käytön oppimista helpottava asia ja siksi on haastavaa, että sovelluksen kaikki toiminnallisuudet eivät ole keskenään yhdenmukaisia, vaan osassa on poikkeuksia. Samaan tavoitteeseen pääseminen yhdessä kohdassa vaatii erilaista toimintaa kuin toisessa kohdassa. Osin selkeä logiikka, joka toistuu monessa paikassa helpottaa oppimista. Etenkin kun harjoittelee pitkiä prosesseja lyhyessä ajassa, tulee käyttäjälle tunne, että sovellus vaatii monia klikkauksia samaan asiaan.</p>
H2	<p>Osittain tuntuu, että käyttäjät on otettu huomioon, mutta ei täysin. Käyttäjät olisi pitänyt kuulla laajemmin. Ei ole itse osallistunut kehittämiseen.</p> <p>Kokee, että kaikkia käytännön asioita ei ole suunniteltu tai otettu huomioon loppuun asti.</p>	<p>Sovellus tuntuu silppumaiselta ja kokonaisuutta on vaikea hahmottaa. Navigointi on vaikeaa. Ei löydä sovelluksen punaista lankaa. Hyvin erilainen nykyjärjestelmiin verrattuna.</p> <p>Samoja asioita kirjataan moneen eri paikkaan, mutta silti käyttäjä ei löydä sieltä etsimäänsä tietoa.</p>
H3	<p>Muistelee, että on joskus ollut mahdollisuus tuoda esiin omia toiveita ja tarpeita tietojärjestelmään liittyen, mutta ei ole tarkkaa muistikuvaa, miten tai milloin.</p>	<p>Sovellus vaikuttaa todella raskaalta, monimutkaiselta ja paljon klikkauksia ja kirjauksia vaativalta. Yksinkertaisen tehtävän tekemiseen kuluu paljon aikaa. Ei ole ohjaava. Käyttäjät ei löydä etsimäänsä asiaa.</p>
H4	<p>Kokee, että käyttäjät ei ole tarpeeksi kuultu järjestelmän suunnittelussa. Muistaa hyvin hämärästi, että joskus olisi kysytty tarpeita, mutta ei muista osallistuneensa.</p> <p>Sovelluksessa on sellaisia huonoja ratkaisuja, joita olisi olettanut kehittämisen ammattilaisten osaavan tehdä paremmin.</p>	<p>Sovellus tuntuu raskassoutuiselta ja vaatii paljon klikkauksia. Tieto on hajallaan ja on vaikea hahmottaa, minne pitäisi kirjata mitään. Aikaa menee tiedon etsimiseen, mutta oleelliselle tiedolle ei ole kunnollista paikkaa. Järjestelmä vaikuttaa todella laajalta.</p> <p>Yksinkertaisen tehtävän tekemiseen kuluu paljon aikaa. Pohtinut, mikä tieto on tarkoituksenmukaista kirjata järjestelmään.</p>
H5	<p>Muistaa, että kauan sitten on ollut mahdollisuus esittää toiveita tietojärjestelmän kehittämiseen liittyen, mutta ei muista itse osallistuneensa.</p>	<p>Yksinkertaisempikin tietojärjestelmä olisi riittänyt, vaikuttaa liian laajalta. Yksinkertaisen tehtävän tekeminen on monimutkaista ja vaatii paljon kirjauksia. Tulisi olla enemmän joustavuutta tilanteen</p>

		<p>mukaisesti. Ei ole ohjaava ja vaatii paljon klikkauksia.</p> <p>Tietojärjestelmä ja siellä oleva tieto toimii työn tukena, joten tietojen pitäisi olla helposti löydettävissä ja tilanne hahmotettavissa. Nyt ei ole.</p>
H6	<p>Ei muista, että olisi kertaakaan kysytty toiveita tai tarpeita tietojärjestelmään liittyen.</p> <p>Toivoisi, että tietojärjestelmää olisi kehittämässä henkilöt, joilla on kokemusta käytännön työstä ja osaisivat kehittää tietojärjestelmää siitä näkökulmasta.</p>	<p>Ei osaa hahmottaa, mistä pitäisi klikata tai mistä klikkaamalla tapahtuu mitään.</p> <p>Ei tiedä, missä on oikeat paikat kaikelle kirjattavalle tiedolle</p>
S1	<p>Tietojärjestelmää on ollut määrittelemässä henkilöt, jotka ovat siirtyneet käytännön työstä määrittelemään. Kokee itsensä myös käyttäjäksi. Lisäksi käytännön työtä tekevät henkilöt ovat olleet koko ajan mukana testaamassa, mistä on saatu jatkuvasti reaaliaikaista tietoa käytännön työstä. Siitä, että käyttäjät ovat määrittelemässä, olisi pitänyt tiedottaa enemmän.</p> <p>Käyttäjien tarpeita on kerätty alusta alkaen, mutta niitä olisi pitänyt kuitenkin ehkä päivittää ajan kuluessa.</p> <p>Käyttäjakeskeisyys ilmenee myös koulutuksina ja koulutuksissa.</p>	<p>Alussa olisi pitänyt paremmin määritellä kokonaisuudet hyvin yksinkertaisina ja lähteä niitä kehittämään lisää. Asioiden yhteen laittaminen on ollut vaikeaa. Jälkikäteen ajateltuna olisi pystynyt toteuttamaan yksinkertaisemmankin järjestelmän. Arveluttaa, kuinka monimutkainen se onkaan.</p>
S2	<p>Käyttäjakeskeisyyden lähtökohtana on ollut, että määrittelijät ja asiantuntijat ovat käyttäjiä ja käyttäjien näkökulma on koko ajan mukana. Käyttäjillä, jotka ovat määrittelleet tietojärjestelmää, ei toisaalta ole ollut riittävää osaamista kehittämisestä tai käytettävyydestä, mikä on varmasti vaikuttanut tietojärjestelmään. Pitää kuitenkin tärkeänä, että määrittelytyössä on mukana ihmisiä, jotka tekevät tietojärjestelmän tukemaa työtä eli käyttäjiä.</p> <p>Määrittelijänä toimineet käyttäjät ja muut käyttäjät eivät ole osanneet muuttaa tarpeitaan puhtaiksi tarpeiksi, jotka eivät ole riippuvaisia vanhoista tietojärjestelmistä.</p> <p>Koko ajan säännölliset testauspäivät, joissa kentän edustajia. Testauspäivillä tulisi keskittyä siihen, vastaako tietojärjestelmä testaajien työtehtäviä ja tukeeko se toimintaa. Testaus on toisaalta ollut pitkälti jälkikäteishuomioimista.</p>	<p>Kokonaisprosessit olisi pitänyt olla tarkemmin määritelty etukäteen projektin alussa ja käyttää pohjatöihin enemmän resursseja. Tällöin lopputulos olisi voinut olla selkeämpi ja käytettävämpi. Nyt osa toiminnallisuuksista on määritelty kokonaisuudesta erillisenä ja myöhemmin havaittu, ettei sovi kokonaisuuteen.</p> <p>Kun kokonaisuus on selvillä, on helpompi toteuttaa myös käyttäjakeskeisyyttä ja osallistaa käyttäjiä.</p> <p>Ketterät menetelmät eivät olleet ehkä paras ratkaisu kehittämiseen jälkikäteen ajateltuna.</p>
S3	<p>Käyttäjakeskeisyys ei ole onnistunut niin hyvin kuin mahdollista osittain sen takia, että käyttäjien sitouttaminen kehittämiseen ei ole onnistunut esim.</p>	<p>Ensin olisi pitänyt tehdä laajat ja yksinkertaiset prosessikuvaukset, tarkat toimintamallit ja yhteiset linjat, joita olisi noudatettu koko kehittämisen ajan ja</p>

	<p>aikataulupaineiden vuoksi. Käyttäjien tavoittaminen nopealla aikataululla on ollut haastavaa. Yhden asian käsittelyyn ei ole ollut käytettävissä niin paljon aikaa, kuin käyttäjät tai heidän tavoittamisensa olisi vaatinut.</p> <p>Testaus toiminut koko ajan ja siellä ollut mukana käyttäjiä ja saatu palautetta. Osa testaaajista olleet mukana jo pidemmän aikaa. Pidetty myös ideariihä, joista saatu hyviä ideoita ja tarpeita laajasti.</p> <p>Suunnittelijoina on ollut käyttäjiä, jotka ovat itselleen suunnitelleet. Suunnittelijoilla olisi kuitenkin pitänyt olla parempi tietämys tietojärjestelmien kehittämisestä, käytettävyydestä ja miten käyttäjätietoa käytetään. Heidän olisi pitänyt saada asiaan enemmän koulutusta ja ohjausta.</p>	<p>kattaen koko tietojärjestelmän. Isot linjat ensin ja siitä edetä yksityiskohtiin kaikki dokumentoituna ja noudatettuna. Tästä olisi luotu kriittinen kehityspolku.</p> <p>Projekti on kestänyt niin pitkään, että olisi pitänyt paremmin pitää kiinni punaisesta langasta ja yhtenäisyydestä. Henkilöt ovat myös vaihtuneet ja heidän tyyliinsä näky.</p> <p>Ketterät menetelmät ei ehkä ole ollut se oikea kehittämisen tapa.</p>
--	--	---

Liite 5. Käyttäjakeskeisyyden haasteiden teemat haastatteluissa

Haastateltava	Tarpeiden määrittelyn vaikeus ilman konkretiaa	Erilaiset käyttäjät ja yksiköt	Käsitteet ja käytettävä kieli
H1	<p>Kun käyttäjät ovat päässeet testaamaan tietojärjestelmää, he ovat esittäneet kehitysehdotuksia, mutta niitä ei ole enää ollut aikataulun vuoksi mahdollista toteuttaa. Tämä on turhauttanut.</p> <p>Käyttäjä osaa antaa palautetta vasta, kun pääsee testaamaan, ja siinä kohden "myöhäistä".</p>		
H2	<p>Harva käyttäjistä osaa teknistä puolta niin hyvin, että osaisi kommentoida ilman, että pääsee itse kokeilemaan. Ei itse osannut ainakaan sanoa ennen testausta mitään.</p> <p>On kysynyt, voisiko toteutusta muuttaa, mutta muutoksia ei pysty tekemään tässä vaiheessa, mikä turhauttaa. Käyttäjien näkemyksiä olisi pitänyt selvittää aikaisemmin.</p>		<p>Sovelluksessa on paljon uusia termejä, joiden opetteleminen vie aikansa, jotta ymmärtää, mistä puhutaan.</p>
H3		<p>Tietojärjestelmä on toteutettu yhdellä tavalla, mutta eri yksiköissä on erilaiset toimintamallit.</p>	<p>Sovelluksessa on todella paljon uusia termejä. Sen lisäksi, että järjestelmä muuttuu, niin muuttuu myös kaikki käsitteet, joiden kanssa on aiemmin työskennellyt.</p> <p>Myös ohjeistuksissa ja koulutuksissa on käytetty uusia termejä, joita ei ole ymmärtänyt. Käsittää, että tietojärjestelmää on kehittänyt IT-alan henkilöt, mutta toivoo, että viestinnässä viestin vastaanottajan tausta otetaan myös huomioon.</p>
H4	<p>Vaikea ottaa kantaa, millaisen haluaisin järjestelmän olevan tai miten haluaisi sen toimivan, ennen kuin on päässyt itse</p>	<p>Kokee, ettei eri yksiköiden eri toimintatapoja ole otettu huomioon tai mietitty loppuun asti.</p>	<p>Termistö tuntuu vielä todella epäselvältä ja vaikeuttaa viestintää. Ei ole löytänyt uusille</p>

	<p>kokeilemaan. Ei ole hahmottanut sisältöä, ennen kuin testaa itse.</p>	<p>Tuo esiin myös sen, että eri yksiköissä ja eri ihmisille voivat samat termit merkitä eri asioita.</p>	<p>termeille paikkaa tai hahmottanut työn kautta niiden merkitystä.</p> <p>Käytössä on myös paljon teknisiä termejä, joita ei ymmärrä.</p> <p>Epävarmuus siitä, osaako käyttää oikeita termejä.</p>
H5	<p>Ei usko, että käyttäjät osaavat esittää tarpeensa tai kuvata prosesseja tietojärjestelmän kehittämisen kannalta tarpeeksi tarkalla tasolla, jotta käyttäjäkeskeisyys toteutuisi.</p> <p>On odottanut, että kokonaisuus valmistuu ja sitä pääsisi testaamaan, jotta voi esittää toiveita ja antaa palautetta.</p>		
H6			<p>Käytettävä termistö on hyvin erilainen. Tietojärjestelmässä puhutaan sanoilla, joita ei ole koskaan työssä käytetty. Kokee käytön vaikeaksi, kun ei ymmärrä, mitä käyttäjältä odotetaan. Ei osaa silloin antaa myöskään palautetta.</p> <p>Kun on pyytänyt apua, on saanut vastauksen termeillä, joita ei ole ymmärtänyt. Tuntee itsensä hölmöksi.</p> <p>Palautetta ei välttämättä anna, jos ei ymmärrä termejä.</p>
S1	<p>Havaintoja on saatu, mutta niitä ei pysty enää ennen käyttöönottoa toteuttamaan. Aikataulullinen vaikeus.</p>	<p>Kaikilta ei voi kysyä, miten haluaisi tietojärjestelmän toimivan. Käyttäjien toiveet ovat osittain ristiriitaisia, mikään ratkaisu ei ole kaikille täydellinen. Osin taustatyökin on tehty rönsyillen.</p> <p>Alussa olisi pitänyt tunnistaa käyttäjäryhmät. Toisaalta tehty tietojärjestelmää monille erilaisille yksiköille, toiminnolle, ryhmille ja käyttäjille.</p>	<p>Sovelluksessa käytettävät termit ovat uusia, mikä varmasti vaikeuttaa oppimista.</p>

S2		Testaamassa on käynyt aika pieni otos koko käyttäjäkunnasta.	
S3		Käyttäjät esittävät toiveet oman yksikkönsä ja toimintansa näkökulmasta, joka voi olla toiselle yksikölle väärä tai vaikea.	

Liite 6. Ylläpitovaiheen käyttäjäkeskeisyyden teemat haastatteluissa

Haastateltava	Palautteen antamisen helppous	Palautteen merkitys	Palautteen hankinnan keinot
H1	Palautteen antamiseen mahdollisimman matala kynnyks, ei esimerkiksi erillistä sovellusta, jonne pitäisi erikseen kirjautua, vaan joku tuttu keino, esim. sähköposti.	Antanut palautetta ja saanut siitä kiitosta, mistä on tullut hyvä mieli. Kehittämisessä mukana olo mielekästä. Käyttäjille tulisi korostaa, ettei kynnyks palautteen antamiseen ole korkea, vaan palautetta halutaan ja tarvitaan.	Kyselyitä tulee niin paljon, ettei käyttäjillä välttämättä ole motivaatiota vastata. Haastattelut voisivat olla toimiva tapa. Yhtenä tapana voisi olla palautteen antaminen tukihenkilölle, joka veisi sen eteenpäin. Toimiva tapa voisi olla myös se, että palautteen kerääminen olisi esimiesten vastuulla.
H2	Muutosehdotuksien ja kysymyksien esittämiseen mahdollisimman helppo tapa ja matalalla kynnyksellä, esimerkiksi sähköpostilla. Olisi valmis osallistumaan myös keskustelutilaisuuksiin, jos osallistuminen ei teetä ylimääräistä työtä.	Kokee tärkeänä, että ylläpito-organisaatio on aktiivinen pyytäessään palautetta ja näin viestii, että palautetta halutaan ja sillä on merkitystä.	Palautetta tulisi pystyä antamaan koska vain, eikä joutua odottamaan oikeaa hetkeä. Palautetta voisi kerätä myös yksiköittäin ja laittaa kootusti eteenpäin. Uskoo, että käyttäjät ovat motivoituneita vastaamaan kyselyihin ja olisi itsekkin. Kyselyitä tulisi laittaa systemaattisesti ja usein. Pitäisi hyödyllisenä, että järjestettäisiin tilaisuuksia, joissa pääsisi ääneen kertomaan ajatuksiaan tai kysymään kysymyksiä. Monimutkaisista asioista helpompi puhua ääneen.
H3	Pitää tärkeänä, että huomion tai palautteen saisi helposti ja nopeasti merkittyä ylös ja laitettua eteenpäin, eikä pitäisi täyttää esimerkiksi erillistä lomakkeistoa tai jäädä odottamaan kyselyä.	Ajatus siitä, että annetulla palautteella on merkitystä, motivoi antamaan palautetta. Jos antaa palautetta, joka ei johda mihinkään, turhauttaa ja aiheuttaa sen, että ei enää anna palautetta.	Palautetta pitäisi pystyä antamaan koko ajan ilman odottelua. On kiinnostunut osallistumaan. Mieluummin kertoisi palautetta suullisesti esimerkiksi keskustelutilaisuudessa kuin kirjallisesti. Näiden asioiden avaaminen on helpompaa puhumalla.
H4	Mahdollisimman selkeä ja yksinkertainen tapa, jolla		Voisi olla ja osallistua myös palautesessioon, jossa saisi

	<p>viestin saa eteenpäin tarkemmin sitä muotoilematta.</p> <p>Niin paljon tehtävää, että kaikki, mikä ei ole pakko, jää tekemättä.</p>		<p>antaa palautetta ääneen puhuen.</p> <p>On motivoitunut vastaamaan käyttäjäkyselyyn, jos se on tarpeeksi selkeä. Jos kyselyä ei saa yhdellä istumalla täytettyä, jää vastaamatta.</p> <p>Kyselyn ja siinä käytettävien termien tulisi olla valittu työnteon näkökulmasta, eikä tietojärjestelmän toteutuksen näkökulmasta.</p>
H5	<p>Kynnys palautteen tai kehitysehdotuksen antamiseen pitäisi olla mahdollisimman pieni.</p>		<p>Toivoo jatkuvaa kyselyä palautteen osalta. Käyttäjien pitäisi voida myös itse antaa jatkuvasti palautetta. Mikäli useampi käyttäjä antaa palautetta samasta asiasta, voisi sen ottaa kehitettäväksi asiaksi.</p> <p>Jos antaa palautteen kirjallisena, voisi palautteen ohkeen liittää pyynnön saada myös keskustella asiasta.</p>
H6	<p>Mahdollisimman yksinkertainen tapa olisi paras, jottei palautteen antaminen jää tekemättä sen vaikeuden vuoksi tai koska ei ymmärrä termejä.</p>	<p>Miettii, onko palautteella jotain merkitystä. Toivoo, että olisi vaikutus käytännössä, mutta ei ole varma. Miksi antaisi palautetta, jos sillä ei ole vaikutusta.</p>	<p>Kyselyt ja keskustelutilaisuudet ovat hyviä tapoja, mutta tärkeintä olisi, että palautetta kysytään ja hankitaan aktiivisesti.</p>
S1		<p>Käyttäjillä tulisi olla aito mahdollisuus tuoda havaintojaan esiin, eikä niin, että palautetta kerätään, mutta niihin ei koskaan palata.</p>	<p>Käyttäjien havaintojen tulisi olla keskiössä. Uudistuksia lähdetään tekemään havaintojen perusteella.</p>
S2			
S3	<p>Mahdollisimman helppo tapa, jolla saa hankittua palautetta mahdollisimman monelta.</p>	<p>Palautteen merkitystä tulisi korostaa koko ajan ja jatkuvasti, ettei se unohdu työn kiireessä.</p>	<p>Käyttäjien näkemyksiä ja palautetta tulisi hankkia mahdollisimman laajasti, helposti ja monelta käyttäjältä.</p>

Liite 7. Käyttäjätiedon vaikutus ja merkitys haastatteluissa

Haastateltava	Käyttäjätiedon vaikutus ja merkitys
S1	<p>Määrittelyyn vaikuttaa niin moni muukin asia kuin vain käyttäjien tarpeet. Ohjaavuudesta on jouduttu karsimaan. Määrittelyn pitäisi olla niin hyvää, ettei kehittäjän tarvi tietää käyttäjien käyttäytymisestä. Käyttäjien toiveet ristiriidassa keskenään ja ristiriidassa muiden tarpeiden kanssa.</p>
S2	<p>Kehittämistä on ohjanneet vahvasti muutkin asiat kuin käyttäjien tarpeet, kuten toimintatapojen yhdenmukaistaminen ja toiminnan kehittäminen. Käytettävyyssääntöjen ja käyttäjätiedon yhdistäminen ollut vaikeaa ja ristiriitaista. Toisaalta käyttäjätietoa ei ole kerätty tarpeeksi systemaattisesti tai riittävästi. Vaikeinta on löytää ratkaisu, joka palvelisi suurinta osaa edes hyvin ja täyttäisi myös muualta tulevat tarpeet. Käyttäjät ei ehkä ole tietoisia tulevista muutoksista ja osaa peilata tietojärjestelmää uuteen toimintamalliin.</p> <p>Kehittäjän tulee ymmärtää kokonaisuus ja miten oma osuus asettuu kokonaisuuteen. Käyttäjätkin muuttuvat ja vaihtuvat, joten on arvioitava, tehdäänkö nykykäyttäjille vai yleisiä sääntöjä noudattaen.</p>
S3	<p>Käyttäjakeskeisyydestä on jouduttu karsimaan resurssipaineiden vuoksi. Tietojärjestelmän kehittämiseen vaikuttaa niin moni muukin asia kuin käyttäjät, kuten laki ja organisaation linjaukset, joita käyttäjät eivät välttämättä aina osaa ottaa huomioon. Vaikeinta on löytää ratkaisu, joka palvelisi käyttäjiä ja täyttäisi kaikki muut tietojärjestelmälle osoitetut tavoitteet. Toiminnan muutos tietojärjestelmän ohella on vaikeaa. Miten voi esittää tarpeita jostain, mitä ei vielä edes tiedä?</p> <p>Yhtenäinen logiikka tärkeämpää kuin käyttäjätieto. Kehittäjän on ymmärrettävä juuri sen tietojärjestelmän käyttäjää ja ymmärrettävä käyttäjän toiminnan taustalla oleva tarve, jotta voi tarjota ratkaisuja. Ensin tulisi kuvata muutos ja sitten suunnitella tietojärjestelmä vastaamaan sitä.</p>