

# Henkilöstöhallinnon tietojärjestelmän käyttöönottotestaus



Rantanen, Riitta

**Laurea-ammattikorkeakoulu**  
Laurea-Kerava

## **Henkilöstöhallinnon tietojärjestelmän käyttönottotestaus**

Riitta Rantanen  
Tietojenkäsittelyn koulutusohjelma  
Opinnäytetyö  
Huhtikuu, 2010

**Laurea University of Applied Sciences**  
Laurea-Kerava

# **The Cutover Testing of Human Resource Management Data System**

Riitta Rantanen  
Information Technology Programme  
Thesis  
April, 2010

Riitta Rantanen

### **Henkilöstöhallinnon tietojärjestelmän käyttöönottotestaus**

Vuosi 2010 Sivumäärä 26

---

Opinnäytetyön tavoitteena oli kerätä tietojärjestelmän käyttöönottotestauksessa tarvittavat tiedot helpottamaan vastaavanlaisen testauksen läpivientiä. Työn esimerkkinä on henkilöstöhallinnon tietojärjestelmän käyttöönottoprojektin testauksen toteutus. Projektin yksilöintitiedot on poistettu.

Esimerkkiprojektissa käytössä olevan palkanlaskentaohjelman rinnalle otetaan käyttöön erillinen henkilöstöhallinnon tietojärjestelmä. Työssä kerrotaan projektista ja sen käyttöönottotestauksen eri vaiheista sekä testitapauksista ja -tuloksista.

Teoriaosuudessa käsitellään tietojärjestelmähanketta, tietojärjestelmän laatuvaatimuksia, testauksen eri vaiheita ja erilaisten käyttäjien muutokseen suhtautumisen vaikutusta käyttöönoton onnistumiseen.

Johtopäätöksissä käydään läpi vastaavanlaisen käyttöönoton onnistuneessa toteutuksessa huomioitavia asioita ja kriittisiä kohtia.

Asiasanat: henkilöstöhallinnon tietojärjestelmä, käyttöönotto, järjestelmätestaus

Riitta Rantanen

**The Cutover Testing of Human Resource Management Data System**

Year 2010 Pages 26

---

In this Bachelor´s thesis the aim was to gather information in the cutover testing of data system to facilitate the testing of a similar implementation. An example in this thesis is the cutover testing project of a Human Resource Management data system. All project designation has been removed from it.

In the example project a separate Human Resource Management data system is being used alongside an existing Payroll. The thesis describes the project as well as cutover testing, test cases and results.

The theoretical part deals with data system projects, standards for data qualities, the various stages of testing and different user´s attitude to change towards the successful introduction.

The conclusion discusses important elements and critical points for to carry out some similar cutover

Key words: Human Resource data system, cutover, system testing

## Sisällys

1	JOHDANTO.....	6
2	HEHA-PROJEKTI.....	7
	2.1 Tausta ja tavoitteet .....	7
	2.2 Organisaatio.....	7
	2.3 Laatuavoite.....	8
	2.4 Aikataulu ja tiedottaminen.....	8
3	MUUTTUVAT TOIMINTATAPAPROSESSIT.....	8
	3.1 Osoitetietojen muuttaminen.....	9
	3.2 Poissaolon hakeminen ja hyväksyminen.....	10
	3.3 Työsopimukset .....	10
4	LIITTYMÄ .....	11
5	TIETOJÄRJESTELMÄN KÄYTTÖÖNOTTO .....	12
	5.1 Tietojärjestelmähänke .....	12
	5.2 Laatuvaatimukset .....	13
	5.3 Käyttöönottotestaus .....	14
	5.3.1 Testauksen tavoitteet.....	14
	5.3.2 Testauksen eri vaiheet.....	15
	5.4 Käytettävyys .....	16
6	HENKILÖSTÖHALLINNON TIETOJÄRJESTELMÄN KÄYTTÖÖNOTTOTESTAUS .....	17
	6.1 Konversion testaus .....	17
	6.2 Integrointi ja järjestelmätestaus.....	18
	6.3 Hyväksymistestaus .....	19
7	JOHTOPÄÄTÖKSET .....	20
	7.1 Testaus.....	20
	7.2 Käyttäjien koulutus ja tuki .....	22
	7.2.1 Koulutus .....	22
	7.2.2 Käyttäjien tuki .....	23
8	ARVIOINTI JA LOPPUYHTEENVETO.....	23
	LÄHTEET .....	25
	KAAVAT JA TAULUKOT .....	26

## 1 Johdanto

Vanha tietojärjestelmä voi toimia ja täyttää sille asetetut tehtävät ja tavoitteet, mutta lisääntyvä tiedontarve ja liiketoimintaprosessien nykyaikaistamisvaatimukset asettavat järjestelmän kehitykselle joskus liian suuret haasteet. Tällaisessa tilanteessa joko kehitetään vanhan rinnalle uusi tietojärjestelmä tai integroidaan olemassa oleva, uudemmalla tekniikalla kehitetty tietojärjestelmä täyttämään uudet vaatimukset.

Tässä opinnäytetyössä käydään läpi 1990-luvulla käyttöönotetun, työasemakohtaisen asennuksen vaativan palkanlaskentaohjelman rinnalle liitetyn selainpohjaisen henkilöstöhallinnon osion käyttöönottoprojektia ja -testausta. Työn taustana käytetään toteutettua henkilöstöhallintotietojärjestelmän käyttöönottoprojektia (jatkossa HEHA-projekti). Sen osapuolina on Yritys A tilaajana ja Ohjelmistoyritys X käyttöönotettavan tietojärjestelmän toimittajana sekä vanhan ja uuden tietojärjestelmän välisen liittymän kehittäjänä.

Osallistuin työssä käsiteltävään HEHA-projektin eri testausvaiheisiin viiden asiantuntijan muodostamassa päättestaajaryhmässä. Opinnäytetyön aihe on siis valittu omasta työelämästä.

Opinnäytetyössä esitellään toteutuneen projektin sisältö, tavoite ja aikataulu yleisellä tasolla sekä kuvataan Yritys A:n kahta henkilöstöhallinnon toimintatapaprosessia ennen ja vastaavasti HEHA-projektin tavoiteprosessin mukaan toteutuneena. Työssä selostetaan käyttöönottestauksen eri vaiheita sekä testitapauksia ja -tuloksia.

Teoriaosuudessa käsitellään:

- tietojärjestelmähanketta
- käyttöönottestausta
- testaussuunnitelmaa
- testauksen tavoitteita
- testauksen eri vaiheita.

Lopuksi käydään läpi vastaavanlaisen projektin onnistuneessa toteutuksessa huomioon otettavia tärkeitä osatekijöitä. Johtopäätöksissä kartoitetaan myös mahdollisia uhkia ja kriittisiä kohtia.

Opinnäytetyön toteutuksessa käytetyt menetelmät ovat olleet yritysten tietohallinnon kehittämiseen, projektityöskentelyyn ja testaukseen liittyvän kirjallisuuden, artikkeleiden ja www-sivujen lukemista. Toteutettu projekti ja tavoiteprosessit on esitelty projektiin osallistumisen pohjalta.

## 2 HEHA-projekti

Projekti toteutettiin perinteisen vesiputousmallin mukaisesti. Projektisuunnitelmaan oli selvästi määritelty hankkeen eri vaiheet.

### 2.1 Tausta ja tavoitteet

Noin kaksi vuotta kestäneen projektin tehtävänä oli saada integroitua käytössä oleva palkka-laskentaohjelma ja nykypäivän henkilöstöjohtamisen vaatimukset täyttävä henkilöstö-hallinnon tietojärjestelmä. Yritys A:n tavoitteena oli saada koko henkilökuntansa käyttöön sähköiset itsepalvelutoiminnot henkilöstöhallinnon asioissa, esimerkiksi vuosiloman anominen. Yrityksen toinen tavoite oli saada sähköinen henkilöstöjohtamista ja esimiestyötä tukeva työkalu, jossa kaikki tarvittava tieto on yhdessä tietokannassa.

Käyttöön otettavan henkilöstöhallinnon tietojärjestelmälle yksi asetettu tavoite oli vähentää paperilomakkeiden ja -tulosteiden käyttöä, joten muutoksella tuli olemaan myös ekologinen vaikutus. Kun muutettava/tallennettava tieto tuli mahdolliseksi syöttää suoraan vain yhteen järjestelmään ilman välikäsiä, tiedon sisältö pysyi muuttumattomana eli virhemahdollisuus väheni. Lisäksi Yritys A halusi syötetyn tiedon olevan nopeasti käytettävissä personoidusti sitä tarvitsevalle. Esimerkiksi työntekijä näkee oman vuosilomansa lisäksi muiden samaan tiimiin kuuluvien vuosilomat, kun taas esimies näkee kaikkien alaistensa vuosilomat.

### 2.2 Organisaatio

HEHA-projektia johti johtoryhmä, jossa oli sekä Yritys A:sta että Ohjelmistoyritys X:stä edustus. Johtoryhmällä oli projektissa päätös- ja johtamisvastuu. Sen tehtävänä oli seurata koko projektin edistymistä aikataulun mukaisesti.

Johtoryhmän alaisuudessa toimi hankeryhmä, joka tiedotti johtoryhmälle alaisuudessaan olevien projektiryhmien edistymisestä ja mahdollisista uusista hankkeeseen liittyvistä tavoitteista tai toimintatarpeista. Projektiryhmiä oli yhteensä neljä, joilla jokaisella oli oma alueensa kokonaisprojektin toteutumisesta. Tässä opinnäytetyössä tarkastellaan vain henkilöstöhallinnon projektiryhmälle kuuluvaa aluetta.

Lisäksi tarvittaessa kutsuttiin koolle tuki- ja asiantuntijaryhmiä, joiden tehtävänä oli käydä läpi eri vaiheiden tulokset ja tehdä tarvittavat korjausehdotukset. Projektiryhmillä oli myös käytössään Yritys A:n eri sidosryhmien edustajat.



### 2.3 Laatumavoite

Koko HEHA-projektin laatumavoitteena oli aikataulun, arvioitujen työmäärien ja kustannusten pitävyys. Näiden paikkansa pitävyyttä valvoi projektin johtoryhmä.

Koko projektin tuloksilla oli omat laatumavoitteet:

- virheetön toiminta
- käyttäjäystävällisyys
- kaikkien osapuolien hyväksymä ratkaisu
- selkeä projektin dokumentointi

Laadunohjaus toteutettiin käynnistyskatselmuksilla, tulostarkasteluksilla ja loppukatselmuksella. Projektin alkuvaiheessa edistymiskatselmuksia suoritettiin kuukausittain ja myöhemmin tarpeen mukaan.

### 2.4 Aikataulu ja tiedottaminen

HEHA-projekti oli aikataulutettu niin, että tietojärjestelmän räätälöinnin määrittelyt ja suunnittelu olivat valmiit vuoden päästä projektin aloituksesta. Varsinainen käyttöönotto-testaus pystyttiin aloittamaan puoli vuotta ennen käyttöönottoa.

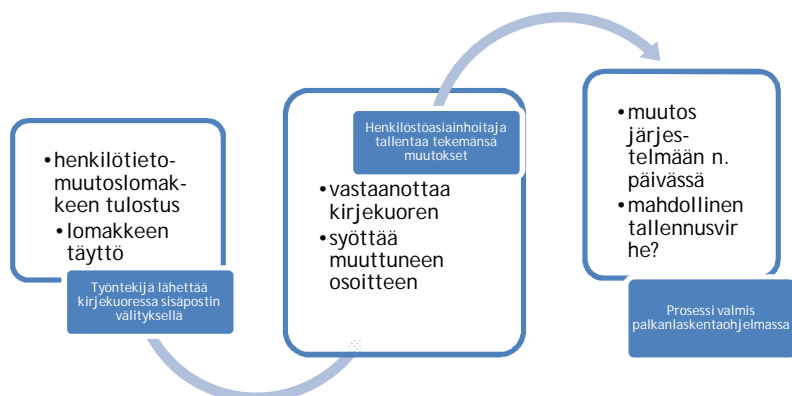
Neljän projektiryhmän välinen tiedotus tapahtui sähköpostitse. Tuleville järjestelmän loppukäyttäjille (yli 10 000 henkilöä) aloitettiin tiedottaminen puoli vuotta ennen suunniteltua käyttöönottoa. Tiedottaminen järjestettiin pitämällä useita infotilaisuuksia, joissa pyrittiin tiedon antamisen lisäksi vähentämään muutosvastarintaa. Yritys A koulutti 60 vapaaehtoista työntekijää pitämään tiedotustilaisuuksia.

## 3 Muuttuvat toimintatapaprosessit

HEHA-projektin jälkeen henkilöstöhallintoon liittyvät prosessit muuttuivat oleellisesti. Koko henkilökunta siirtyi käyttämään sähköistä työkalua henkilöstöhallinnon asioissaan. Muutokset koskivat mm. työntekijän osoitetiedon muuttamisprosessia samoin kuin poissaolon hakemis- ja myöntämisprosessia.

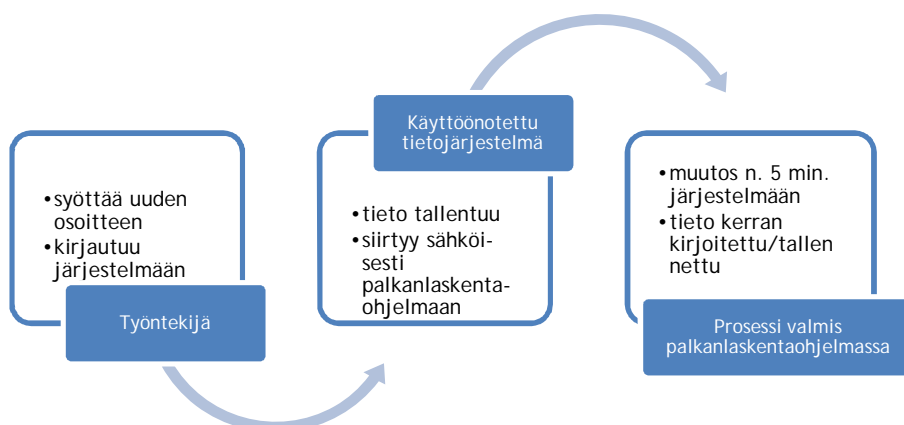
### 3.1 Osoitetietojen muuttaminen

Ennen käyttöönottoa työntekijä ilmoitti osoitemuutoksensa seuraavasti:



Kaavio 1: Vanha osoitemuutosprosessi.

Alla kuvataan, miten osoitemuutos siirtyy palkanlaskentaohjelmaan uuden prosessin mukaan:

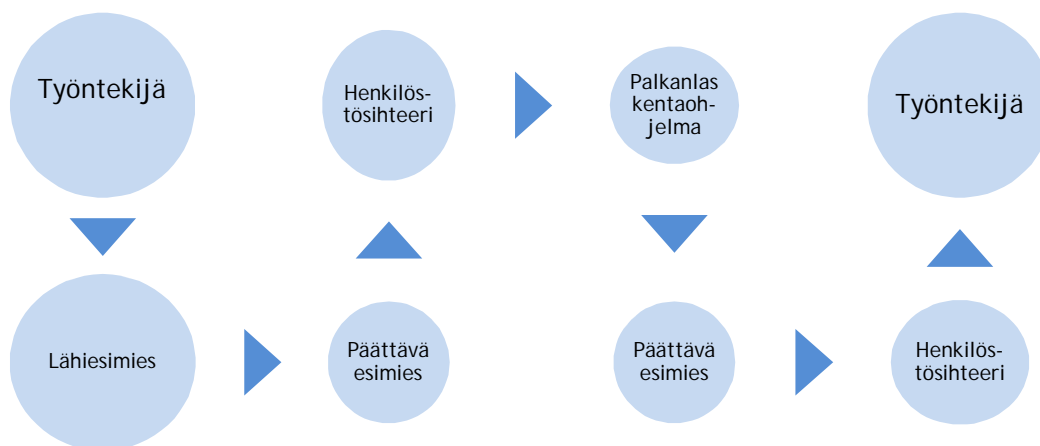


Kaavio 2: Uusi osoitemuutosprosessi.

Uusi prosessi nopeuttaa huomattavasti työntekijän ilmoittamien henkilötietomuutosten siirtymistä palkanlaskentaohjelmaan. Tämä tulee vähentämään postitusvirheitä sekä pankkitilin muutoksen ilmoittamisen nopeutumisen ohjaa nettopalkan oikealle tilille, ja näin vähenevät maksatuksen käsinkorjaukset.

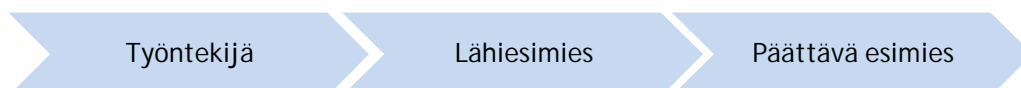
### 3.2 Poissaolon hakeminen ja hyväksyminen

Ennen käyttöönottoa työntekijä anoi esimerkiksi yhden kalenteriviikon palkatonta vapaata lomakkeella, joka "liikkui" kaavion 3 osoittamalla tavalla.



Kaavio 3: Vanha poissaolon hakuprosessi.

Koko prosessi vei keskimäärin kaksi vuorokautta anomuksen lähettämisestä siihen hetkeen, kun työntekijä sai tiedon hyväksytystä poissaolosta. Käytön jälkeen koko prosessi työntekijältä palkanlaskentaohjelmaan vie keskimäärin maksimissaan puoli vuorokautta. Tieto siirtyy liittymän siirtotauluun heti päättävän esimiehen hyväksynnän jälkeen, mutta liittymä siirtää poissaolot palkanlaskentaohjelmaan kaksi kertaa vuorokaudessa.



Kaavio 4: Uusi poissaolon hakuprosessi.

Koko prosessin työnkulku muuttui sähköiseen muotoon. Joka vaiheen jälkeen järjestelmä on määritelty lähettämään sähköposti-ilmoitus automaattisesti tapahtuman osapuolille.

### 3.3 Työsopimukset

Kun koko käyttöönotto on valmis, esimiehet tekevät itse työsopimukset sähköisesti. Tarkoituksena on, että työhönottotilanteessa työhön ottava esimies syöttää tiedot suoraan järjestelmään käyttämättä mitään paperilomaketta välikätenä. Tämän jälkeen hän välittää tietojärjestelmän välityksellä työsopimuksen päättävälle esimiehelle, joka vahvistaa ja allekirjoittaa tulostamansa sopimuksen. Päättävä esimies toimittaa paperitulosteet

työntekijän allekirjoitettavaksi. Uudet työsopimustiedot siirtyvät käyttöön otettavasta tietojärjestelmästä liittymän kautta palkanlaskentaohjelmaan.

#### 4 Liittymä

Käyttöön otettavan henkilöstöhallinnon tietojärjestelmästä piti saada tallennettu tieto siirtymään samansisältöisenä ja oikeamuotoisena palkanlaskentaohjelmaan. Tähän Ohjelmistoyritys X kehitti liittymän järjestelmien väliseen rajapintaan. Liittymässä piti myös pystyä siirtämään tarvittaessa tietoa toisinpäin.

Liittymän tehtävänä on tarkistaa siirrettävän tiedon sisältö ja muoto sen tallentuessa siirtotauluun. Tarkistussäännöissä on myös uuden ja vanhan tiedon erottaminen. Jos tieto on uusi eli sitä ei löydy vielä palkanlaskentaohjelmasta, se tallentuu ohjelmaan automaattisesti. Liittymän tulkitessa syötetyn tiedon olevan korjaus palkanlaskentaohjelmassa jo olemassa olevaan tietoon, jää tieto siirtymättä. Tällaiset korjaukset ohjeistettiin syöttämään erikseen palkanlaskentaohjelmaan niin kauan, kunnes valmius automaattiseen tiedon korjauksen siirtoon on olemassa. Katsottiin tarpeelliseksi, että käyttäjät saavat ensin lisää kokemusta järjestelmien välisen tiedonsiirron tarkistamisesta. Siirtymättömät tiedot tulostuivat huomautuslistalle.

Alla olevassa taulukossa esitetään kaksi esimerkkiä liittymän tiedon siirtymisen käsittelystä:

Tieto	HEHA-järj.	Tarkistus		Liittymä	Palkanlaskenta
Työsuhde	tallennus	uusi?	on	siirtyy	autom.
			ei	ei siirry	käsin korj.
Viikkotyö	tallennus	uusi?	on	siirtyy	autom.
			ei	ei siirry	käsin korj.

Taulukko 1: Liittymän tiedon siirto.

Osa palkanlaskentaohjelman tarvitsemista tiedoista oli sellaisia, joita ei tarvittu henkilöstöhallinnon tietojärjestelmässä. Näiden tietojen muodostuminen määriteltiin liittymän tehtäväksi.

Henkilöstöhallinnon tietojärjestelmässä ei tarvittu palkanmaksupäivätietoa näkyviin. Se määriteltiin tulostumaan työsopimukselle automaattisesti 1. maksupäivänä, jos työsuhde on lajiltaan vakituinen. Kun työsuhteen tietoihin tallennetaan päättymispäivä, määrittelyjen mukaan tulostuu sopimukseen 2. maksupäivä. Palkanlaskentaohjelmaan vastaavan tiedon siirtyminen määriteltiin liittymän pääteltäväksi. Samankaltaisia liittymän pääteltäviä tietoja on kirjanpito (vakituinen/ei vakituinen) sekä tapaturmaluokitus, jonka liittymä päättelee ammattinimikeryhmän perusteella.

## 5 Tietojärjestelmän käyttöönotto

Yritys voi päättäessään kehittää liiketoimintaprosessejansa integroida samaan tietojärjestelmän kehittämishankkeen. Järjestelmän käyttöönottotestausvaiheissa on huomioitava prosessien muutostavoitteet.

### 5.1 Tietojärjestelmähänke

Kirjassa Yrityksen tietohallinto esitetään kaavion muodossa, kuinka liiketoimintaprosessien nykytila kartoitetaan ensin, jonka jälkeen prosesseille asetetaan tavoitteet, sekä aloitetaan tietojärjestelmän kehittämishanke esikartoituksella ja määrittelyllä. Seuraavassa vaiheessa sekä uusi prosessi, että järjestelmä on suunnitteluvaiheessa. Prosessien muutoksia valmistellessa tietojärjestelmä on toteutusvaiheessa. Molempien käyttöönotto tapahtuu samanaikaisesti. (Ruohonen & Salmela 2005, 81.) TIEKE Tietoyhteiskunnan kehittämiskeskus ry:n oppaassa todetaan integroitujen tietokantojen tehostavan yritysten liiketoimintaprosesseja kerran syötetyn tiedon siirtyessä automaattisesti toiseen järjestelmään. (Sähköisen kaupankäynnin aapinen 1999.)

Ruohonen ja Salmela (2005, 83) mittaavat tietojärjestelmän onnistumista eri näkökulmista, jotka ovat järjestelmän tekninen laatu, järjestelmän tuottaman informaation laatu, järjestelmän vaikutus työhön ja päätöksentekoon, järjestelmän vaikutus yrityksen liiketoimintaprosesseihin ja kilpailukykyyn. Onnistumista uhkaavina riskeinä he luettelevat tekniset, resurssi- ja ympäristön muutokseen liittyvät riskit, muutosvastarinta ja poliittiset riskit sekä kilpailutilanteeseen liittyvät riskit.

Tietojärjestelmän toteutushankkeessa voi käyttää perinteisen vesiputousmallin sijasta jotain muotoa agile- eli ketterämenetelmistä, joissa on omat prosessimallit projektien toteuttamiseksi. Kainulainen luettelee lopputyössään ketterämenetelmistä, esimerkiksi ASD (=Adaptive Systems Development), XP (=Extreme-Programming), Scrum, Crystal-menetelmät, FDD (=Feature-Driven Development), DSDM (=Dynamic Systems Development Method) ja TDD (=Test-Driven Development). Kainulainen toteaa opinnäytetyössään mm., että Scrumin kattavasta prosessimallista puuttuu toimintotestaus, XP:ssä testaus aloitetaan aina kirjoittamalla yksikkötesti, ja sen jälkeen ohjelmakoodi, ja Crystal-menetelmissä ei ole valmiita malleja prosessien kehittämiseen. FDD-menetelmässä kahta viimeisestä aliprosessia, omaisuuden toteutuksen suunnittelua ja toteuttamista toistetaan niin kauan, että kaikki luetteloidut ominaisuudet on saatu toteutetuksi. (Kainulainen 2008.)

## 5.2 Laatuvaatimukset

Tietojen laatuongelmat paljastuvat testausvaiheessa. Laadun parantamiseksi yrityksellä pitäisi olla laatustrategia, josta käy ilmi yrityksen laatutavoitteet. Laadun parantamisen vaikutuskeinoja ovat myös eri menetelmät:

- profilointi, jolla kyselyjen avulla selvitetään esimerkiksi testauksen jäljiltä jäämiä vääriä tiedon arvoja
- yhdistäminen, jolla tarkoitetaan eri lähteiden samaa tarkoittavan tiedon vertailua ja integrointia
- monitorointi, joka on tietojen laadun jatkuvaa valvontaa. (Hovi, Hervonen & Koistinen 2009, 68 - 69.)

Tietojärjestelmälle on aina asetettava tietyt laatuvaatimukset, joilla arvioidaan järjestelmän kokonaislaatua:

- Tekninen laatu arvioidaan järjestelmälaatuna. Se sisältää mm. järjestelmän luotettavuuden ja tarkkuuden, käytön helppouden sekä hankkeen hyödyntämisen. Se vaikuttaa järjestelmän käyttöön ja käyttäjien suhtautumiseen käyttöönotettavaan järjestelmään.
- Tiedon tärkeys, tarkkuus, virheettömyys, käyttökelpoisuus ja oikea-aikaisuus arvioidaan tietosisällön laatuna. Käyttäjien tietotarpeet täyttävä järjestelmä lisää käyttäjien tyytyväisyyttä.
- Käyttäjätyytyväisyydellä saadaan arvioitua tietojärjestelmän hyödyllisyyttä. Laatumittareina käytetään kokonais- tai päätöksentekotyytyväisyyttä. Mittarina voi olla myös tietojärjestelmästä halutun ja saadun tiedon eron arviointi. Yleisin käytetty mittari on kokonaistyytyväisyyden arviointi, koska yritykselle on tuottavampia ne käyttäjät, jotka ovat järjestelmään tyytyväisiä.
- Tiedon käyttö voi olla vähäistä. Käyttöä voi mitata esimerkiksi seuraamalla kokonaiskäyttöaikoja, käyttäjille tehdyillä kyselyillä tai yhdistämällä nämä tavat.
- Yksiköiden toimintaan vaikutuksen mittareina on esimerkiksi tehtävän suorittamiseen kuluva aika, tulkinnan tarkkuus ja päätöksentekoprosessin kesto.
- Järjestelmän vaikutuksina organisaatioon on esimerkiksi kilpailuaseman muuttuminen, työntekijöiden työn kehittyminen ja yhteistyön kehittyminen. (Ruohonen & Salmela 2005, 183 - 185.)

Vuori kirjoittaa laadunvarmistuksen kehittymisestä, että nykypäivänä pitäisi joka projektissa olla teemana käytettävyys ja tietoturva sekä järjestelmän välttämättömänä testaus-tehtävänä kuormitustestaus. Käytettävyys on usein projekteissa mukanaoleva keskeinen laatutekijä. Hän muistuttaa, että käyttäjäkeskeisen suunnittelun osaamisen ja soveltamisen lisäksi on varmistettava aina ohjelmistokehityksessä lopputuloksen laatu. Käyttäjäkokemus

kattaa ergonomiapohjaista käytettävyyttä laajemmin käyttäjän kokema laatua. Kyseessä on käyttäjän ja tuotteen eheästä suhteesta, tunnesiteistä ja kokonaisvaltaisesta kokemuksesta. (Vuori 2009, 4.)

### 5.3 Käyttöönottotestaus

Testauksen tarkoituksena on löytää tietojärjestelmästä virheitä. Pohjonen kertoo, että testaus tapahtuu yleensä V-mallin mukaisesti, jossa testaus on jaettu moduuli-, integrointi- ja järjestelmätestauksiin. Moduulitestauksessa testataan yksittäisiä moduuleja, integrointi-testauksessa taas moduulien yhteistoimintaa ja järjestelmätestauksessa testataan kaikkia toimintoja ja järjestelmän suorituskykyä. (Pohjonen 2002, 35.)

Testausvaiheeseen liittyvät riskit on hyvä tiedostaa ja minimoida ennakkoon. Toteutettavissa projekteissa on tullut esille mm., ettei testaajilla ole aikaa testaukseen, testaussuunnitelmaa ei ole laadittu tai sitä ei noudateta, ei edetä systemaattisesti, yritys ei ymmärrä testauksen merkitystä, testaajat eivät tiedä syytä testaukseen, testiaineisto eikä -tapaukset ole kattavia, ohjelmiston toimittajan korjaukset tulevat myöhässä, ja testaus suoritetaan ylimalkaisesti.

Ennen testauksen aloittamista on sovittava testausmenetelmä ja laadittava testaussuunnitelma, josta käy ilmi

- mitä testataan
- testausaikataulu
- testaajat
- testausaineisto
- raportointi
- testauksen hyväksyminen
- mahdolliset rajaukset. (Hovi ym. 2009, 168 -170.)

#### 5.3.1 Testauksen tavoitteet

Testauksen ensimmäinen tavoite on aina varmistaa, että käyttöön otettava tietojärjestelmä vastaa sille asetettuja vaatimuksia ja tavoitteita liiketoimintaprosessien tavoitteet. Testauksella pyritään myös löytämään ja korjaamaan järjestelmän virheitä ajoissa ennen käyttöönottoa.

Testauksen tavoitteita on myös selvittää järjestelmän käytettävyys sekä käyttäjien että teknisestä näkökulmasta katsottuna, ylläpidettävyys (asennus-, päivitys- ja huoltotöiden helppous), tietoturva, siirrettävyys ympäristöstä toiseen, modulaarisuus/hajautus, suorituskyky ja virhetilanteista selviytyminen. (Karkulehto & Urpi 2003.)

### 5.3.2 Testauksen eri vaiheet

Tietojärjestelmän käyttöönottestaus tapahtuu vaiheittain:

- yksikkötestauksessa tehdään tekniikkaan ja suorituskyykyyn liittyviä testejä.
- integraatiotestauksessa varmistetaan ohjelmistojen rajapintojen toimivuutta testaamalla eri osien yhteensopivuus.
- järjestelmätestaus huomioi kaikki toiminnot. Jos järjestelmää kehitetään testauksen aikana, testataan aina aikaisemmin testatut osiot. Vasta tämän jälkeen uudet testattiin uusia toimintoja, jotta varmistuttiin, ettei uudet ole aiheuttaneet virhetilanteita aikaisemmin testattuihin osioihin. Tässä vaiheessa testataan myös käyttöönotettavan ja muiden järjestelmien väliin tehtyt liittymät sekä tehdään suorituskyykytestausta.
- vastaanottotestauksessa tietotekniikan ammattilaiset testaavat asennukset ja käyttöoikeudet. Tämä tehdään ennen kuin käyttäjät pääsevät testaamaan järjestelmää.
- hyväksymistestauksessa hyväksytään tuotantoon siirtyminen.

Yksikkö- ja integraatiotestauksen suorittavat toteuttaja/ulkopuolinen toimittaja. Käyttäjät tekevät järjestelmä- ja hyväksymistestauksen. (Hovi ym. 2009, 169.)

Etenkin järjestelmätestauksessa on erittäin tärkeätä kirjata löydetyt virheet. Virheet löytyvät testaajien tehdessä testit normaalitapausten lisäksi käyttöohjeen vastaisesti, käyttäen vääriä tietoyhdistelmiä, jättämällä osan tarvittavista tiedoista pois, kokonaan pois tai tallentamalla liikaa. Esiin tulleet virheet luokitellaan käyttäjän arvion mukaan

- vähäiseksi eli esteettiseksi.
- ärsyttäväksi eli käyttäjä pitää toimintoa kummallisena.
- kohtalaiseksi eli huonontaa suorituskyykyä.
- estäväksi eli järjestelmä ei käsittele oikeamuotoista tietoa.
- vakavaksi eli tallennettu tieto katoaa.
- erittäin vakavaksi eli tietosisältö muuttaa merkitystä.
- kohtuuttomaksi eli erittäin vakavat virheet toistuvat usein.
- kestäättömäksi eli virhe aiheuttaa tietokannan toimintahäiriön, joka on työläs korjata.
- katastrofaaliseksi eli järjestelmän toiminta päättyy itseksään.
- saastuttavaksi eli järjestelmä aiheuttaa myös toisissa järjestelmissä virheitä. (tSoft 2007.)



## 5.4 Käytettävyys

Tuotetta kehitettäessä on tiedettävä tulevat käyttäjät, käyttäjän tavoite, missä käyttö tehdään, mitä käyttäjät tekevät käyttäessään tuotetta ja näistä johtuvat käytettävyysvaatimukset. Jacob Nielsen mukaan hyvä käytettävyys muodostuu käyttötilanteen opittavuudesta, virheettömyydestä, muistettavuudesta, tehokkuudesta ja miellyttävyydestä. ISO 9241-11 ”Standardi näyttöpäätetyön ergonomiasta - ohjeita käytettävyydessä” määrittää käytettävyys käyttötilanteesta riippuvaiseksi, jossa hyvä käytettävyys muodostuu tehokkuudesta, miellyttävyydestä sekä tuottavuudesta. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2002, 17, 19.)

Käytettävyyden voi mitata käytettävyydestillä, jonka esiin tuomien käytettävyysongelmien korjaaminen on yrityksen kustannusten säästöä. Käytettävyydesti tapahtuu osissa, jotka ovat testin järjestäminen ja testaussuunnitelman laatiminen, testin suorittaminen, sekä testin analysointi ja testiraportin laatiminen. Testi voidaan suorittaa joko kvantitatiivisena (käyttöliittymän laatu verrattuna annettuihin käytettävyystavoihteisiin tai vastaavanlaiseen tuotteeseen) tai kvalitatiivisena (pyritään löytämään käytettävyyden ongelmallisia kohtia, jotka voidaan korjata tai oheistaa). Käytettävyystavoihteina voi olla esimerkiksi yleinen käytettävyys, sopivuus kokeneelle käyttäjälle, sopivuus ”mene ja käytä” -käyttöön, sopivuus epäsäännölliseen käyttöön ja virheensieto. (Sinkkonen ym. 2002, 301 - 304.)

Käytettävyydestin aikana tai jälkeen voi mitata lukumääriä, joilla on merkitystä käytettävyysvaatimusten saavuttamisen toteutukseksi, mm:

- tehtävän tekemiseen kuluvan ajan
- tietynä aikana tehtyjen tehtävien määrän
- virheelliset suoritukset
- käyttäjien tekemät käytettävyysvirheet
- ohjeiden käyttökerrat
- montako kertaa käyttäjä hapuili tai eksyi kokonaan?
- ohjaajan vihjeen tai avun määrän. (Sinkkonen ym. 2002, 305.)

Käytettävyydestauksen toteuttamisen tavallisimmat testausmenetelmät ovat:

- ääneenajattelu, jossa käyttäjä tehtäviä tehdessään kertoo koko ajan, mitä on tekemässä.
- paritestit, jossa kaksi testaajaa käyttävät yhtä aikaa samaa järjestelmää keskustellen järjestelmästä keskenään.
- yhteisläpikäynti, jossa ohjaaja ja testaaja etenevät testissä keskustellen järjestelmästä.

- jälkikäteen haastattelu, joka tehdään testaajan itsenäisen työskentelyn jälkeen. Haastattelu tehdään joko suullisesti tai kyselylomakkeella.
- jälkepäin kommentointi, jossa testaaja tekee tehtävän itsekseen, jonka jälkeen hän kommentoi tilanteita nauhalta, jonka ohjaajat katsovat.
- pikkutestit, osatellit, testaaja tekee kynällä ja paperilla tehtävän ja kommentoi tekemisensä ohjaajalle.
- ryhmäläpikäynti, jossa testaajat, ohjaaja ja järjestelmän suunnittelija käyvät tehtävät läpi käyttäen kuvia, kynää ja paperia.
- vapaa läpikäynti, jossa testaaja kokeilee järjestelmään itsekseen. (Sinkkonen ym. 2002, 309 - 312.)

Testitehtävien määrä riippuu käytettävästä ajasta. Usein testin pituus on yksi tunti, joka on jo todella raskas monelle testaajalle, joten aikaa ei kannata ylittää. (Sinkkonen ym. 2002, 312.)

## 6 Henkilöstöhallinnon tietojärjestelmän käyttöönottestaus

Kun käyttöönotettavan tietojärjestelmän räätälöinnin määrittelyt ja suunnittelu sekä näiden ohella tekninen toteutustyö tehty, testiympäristö saatiin käyttöön. Käyttöönottestaus aloitettiin puoli vuotta ennen aikataulullista käyttöönottoa.

Testauksissa selvitettiin vanhasta palkanlaskentaohjelmasta siirrettävien tietojen oikeellisuus ja toimivuus, kun niitä käytettiin hyväksi uuden tiedon välittämiseen käyttöönotetusta tietojärjestelmästä palkanlaskentaohjelmaan. Samalla testattiin ohjelmien välillä toimiva liittymä. Tarkistettiin, että liittymä siirsi tiedot määrittelyjen mukaisesti ja siirtymättömät tiedot löytyivät huomautuslistalta.

### 6.1 Konversion testaus

Ennen kuin päästiin tekemään integrointi- ja järjestelmätestausta aloitettiin palkanlaskentaohjelman käyttöönotettavaan tietojärjestelmään siirrettävien tietojen konvertoinnin testaus. Ensin konvertoitiin palkanlaskentaohjelman testipuolelta 50 henkilön henkilö- ja työsuhdetiedot käyttöönotettavan järjestelmän testitietokantaan. Testitapauksien valinnan ja testitulosten tarkistamisen suoritti Yritys A:n päätestaajaryhmä.

50 testihenkilöä valittiin osittain sattumanvaraisesti, mutta osalla oli myös muutoksia työsuhdetiedoissa. Näin saatiin tarkistettua, siirtyivätkö tiedot sisällöllisesti oikein ja jäikö jotain tietoa siirtymättä. Muutoksia oli esimerkiksi työsopimukselle tehty määräaikainen muutos, kun työntekijä toimi esimiehensä sijaisena tämän loman aikana.

Testikonversion jälkeen todettiin, että

- henkilöiden perustiedot (mm. osoite- ja pankkitiedot) siirtyivät oikein.
- suomenkielen kielikoodi oli muodossa SF, kun määrittelyn mukaan kuului olla fi.
- vakituiset työsuhteet, jossa ei ollut muutoksia, siirtyivät oikein.
- vakituiseen työsuhteeseen tehdyt muutokset eivät siirtyneet oikein.
- jos palkkakustannukset jakautuivat useampaan kustannuspaikkaan, organisaatiotiedot eivät siirtyneet oikein.
- toimistohenkilökunnan työsuhteet eivät siirtyneet, koska vastaanottavasta järjestelmästä puuttui ko. ammattiryhmän työn vaativuuskoodien vastaavuus.

Testitulokset tutkittiin ja huomattiin, että tietojärjestelmien tietojen käsittelytavassa oli eroa päivämäärien kohdalla. Jos työsuhdetta kuvaa aikajanalla, niin palkanlaskentaohjelmassa vakituisella työsuhteella oli oma aikajana ilman päättymispäivää. Kun muutosesimerkin mukaisesti työntekijälle oli tehty työsuhteeseen määräaikainen muutos esimiehen tehtävien hoidosta, palkanlaskentaohjelmassa muodostui muutoksen ajalle oma aikajana, jonka perusteella työntekijä sai palkkansa. Käyttöön otettavassa tietojärjestelmässä oli vain yksi aikajana, joka katkesi aina muutosten mukaisesti. Näin ollen liittymä ei osannut siirtää oikein niiden työsuhteiden tietoja, joissa oli useampia muutoksia.

Ohjelmistoyritys X teki korjaukset konversiomäärityksiin ja tehtiin uusi testi samoilla 50 henkilöillä. Testituloksista kävi ilmi, että määrittelykorjaukset oli tehty oikein ja pystyttiin tekemään koko aineiston eli yli 10 000 työsuhteen tietojen testikonvertointi. Tietosisällön tarkistusta jatkettiin edelleen järjestelmätestauksen rinnalla.

## 6.2 Integrointi ja järjestelmätestaus

Yritys A järjesti opastuksen testauksen toteuttamisesta 40 vapaaehtoiselle järjestelmätestausta suorittavalle työntekijälleen. Testaajat toimivat yrityksen henkilöstö-asioiden hoitajina tai palkanlaskijoina. Heidät ohjeistettiin dokumentoimaan kaikki testaukseen liittävät tapahtumat ja virheet. Samalla testattiin ja dokumentoitiin uusien toimintatapaprosessien työnkulku. Testaajilla oli huomautusten dokumentointiin sähköinen tietojenkeruujärjestelmä käytössään. Tähän järjestelmään kirjattiin myös dokumentoitujen tapauksen käsittelyvaiheet ja valmistumiskuittaukset.

Järjestelmätestaus toteutettiin eri vaiheissa:

1. Uusien henkilötietojen tallennus ja siirtyminen liittymän kautta.
2. Henkilötietojen korjaus ja siirtyminen liittymän kautta.
3. Poissaolojen tallennus ja siirtyminen liittymän kautta.
4. Poissaolojen korjaus/mitätöinti ja näiden siirtyminen liittymän kautta.
5. Uusien vakituisten työsuhteiden tallennus ja siirtyminen liittymän kautta.

6. Uusien määräaikaisten työsuhteiden tallennus ja siirtyminen liittymän kautta.
7. Muiden koko projektiin liittyvien prosessien ja muutosten testaus, ja näiden tietojen käytettävyys.

Testauksessa pyrittiin saamaan selville tiedon muoto, virheettömyys, oikea-aikaisuus ja käytettävyys. Huomattiin, että tiedon tallennustavalla oli suuri merkitys sen käytettävyydellä ja virheettömyydellä palkanlaskentaohjelmassa. Tämän vuoksi ohjelmistoyritys aloitti tallennussääntöjen määrittelyn käyttöönotettavaan järjestelmään. Esimerkiksi kaikkiin tietoihin, joihin kuului tallentaa desimaalit, oli ehdottomasti erottimena käytettävä pilkkua. Pisteellä tai kahdella nolllalla tallennettu tieto siirtyi palkanlaskentaohjelmaan kokonaislukuna, jonka perään muodostui desimaalit pilkulla erottuna. Tämä aiheutti palkanmaksuvirheen, joten tietokenttiin oli määriteltävä syöttörajaus.

Testitapauksia toistettiin paljon, ja Ohjelmistoyritys X teki tarvittaessa muutoksia käyttöönotettavan tietojärjestelmän ohjaustietoihin. Muutosten jälkeen toistettiin uudelleen testitapauksia. Testausvaiheessa huomatu lisärätälöintitarpeet dokumentoitiin projekti-ryhmälle käsiteltäväksi. Järjestelmätestauksen aikana saatiin selvitettyä, korjattua ja tarkistettua konversiosta löytyneitä epäkohtia.

Järjestelmätestausvaiheeseen oli varattu neljä kalenterikuukautta aikaa. Testauksen loppuvaiheessa tehtiin myös käyttöönotettavan tietojärjestelmän pääkäyttäjätoimintojen testaus. Pääkäyttäjätoimintona testattiin käyttäjätunnusten ylläpitoa, organisaatietietojen syöttöä, organisaation hierarkian ja esimiestietojen ylläpitoa. Lisäksi aloitettiin organisaatiomuutosten aiheuttamien työsuhteiden siirtoajojen testaus. Tämä osa jätettiin kuitenkin käyttöönoton jälkeiseen aikaan, jotta testaus olisi lähempänä oikeaa ajankohtaa eli seuraavassa vuodenvaihteessa toteutettavia organisaatiomuutoksia.

Käyttöönotettavan henkilöstöhallinnon tietojärjestelmän asiantuntijatehtäviin valitut henkilöt laativat järjestelmätestauksen aikana eri työvaiheista toimintatapaohjeet, joita testattiin välittömästi valmistumisen jälkeen. Näin saatiin tärkeimpiä työvaiheita koskeva ohjeistus valmiiksi ennen käyttöönottoa.

### 6.3 Hyväksymistestaus

Kun järjestelmätestaus oli suoritettu, niin päätestäajaryhmä suoritti hyväksymistestauksen. Testaukseen oli laadittu tarkka testausuunnitelma tapauskohtaisesti. Testaus tilanteesta kirjattiin hyväksymistestausraportti, johon kirjattiin kaikki toimivat toiminto-osiot ja testaus tilanteessa toimimattomat toiminnot.

Ryhmä kävi kaikki osa-alueet läpi ja neuvotteli, onko käyttöönottoa estäviä virheitä tullut esiin. Hyväksymistestausraportti toimitettiin hankeryhmän kautta koko projektin johtoryhmälle, joka päätti HEHA-projektin osalta uuden henkilöstöhallintotietojärjestelmän käyttöönotosta käytössä olevan palkanlaskentaohjelman rinnalle.

## 7 Johtopäätökset

Minkä tahansa tietojärjestelmän käyttöönottohankkeen onnistumisen mittareita ovat järjestelmän tekninen laatu, tuottaman tiedon laatu, käyttäjätyytyväisyys, järjestelmän osuus liiketoimintaprosesseihin ja yrityksen toiminnan tehostuminen käyttöönoton myötä. Suunnitteluvaiheessa on myös muistettava huomioida ja ennalta ehkäistä mahdolliset vastaan tulevat riskit, joita ovat esimerkiksi tekniset riskit, resurssiriskit ja muutosvastarinta. (Ruohonen & Salmela 2005, 83.)

### 7.1 Testaus

HEHA-projektissa olleen tietojärjestelmän tapaisissa käyttöönotoissa voidaan luetella testausvaiheen onnistumisen mittareina ja riskitekijöinä aivan samat tekijät kuin mitä edellisessä kappaleessa on lueteltu. Testausuunnitelman tueksi voidaan laatia esimerkiksi järjestelmätestauksen SWOT-analyysi, johon kirjataan tiedossa olevat testausvaiheen vahvuudet ja heikkoudet sekä testauksesta saadut mahdollisuudet ja aiheutuneet uhat käyttöönottovaiheessa.

<b>Testaus:</b>	<b>Vahvuudet:</b> - asiantuntija-apu - kirjalliset testitapaukset - tulos- ja muutosraportit - testausryhmän oman alan ammattitaito	<b>Heikkoudet:</b> - tekniset ongelmat - aikataulumuutokset - testaajien testausosaaminen - muuttuvat resurssit
<b>Käyttöönotto:</b>	<b>Mahdollisuudet:</b> - uudet toimivat prosessit - saman tiedon tallentaminen vain kerran - ajantasalla oleva tieto - kustannustehokkuus	<b>Uhat:</b> - muutosvastarinta - käyttäjien virheet - riittämätön koulutus

Taulukko 2: SWOT-esimerkki.

Testausvaiheessa testausryhmällä on käytössään tietojärjestelmän tuottajan osaaminen ja testauksessa ilmenneet järjestelmävirheet saadaan heti korjaukseen. Kun testitapaukset

laaditaan kirjallisesti etukäteen, niin testaajan on helpompi suorittaa testi käyttöohjeiden mukaan ja niiden vastaisesti. Näin saadaan kaikki toimintaketjut järjestelmällisesti testattua ja raportoitua testien tulokset. Testitulokset kirjataan myös kaikki järjestelmään tehdyt muutokset ajankohtaiseen, jotta tilanteisiin pystytään palaamaan tarvittaessa. Järjestelmätestausta suorittavien oman alansa ammattitaito auttaa heitä tulkitsemaan saatuja tuloksia ja varmistamaan siirtyvän tiedon oikeellisuutta.

Testauksen toteuttaminen voi epäonnistua, jos testiympäristössä kaikki osat eivät toimi oikein tai joku tekninen osa on ns. pois päältä. On myös mahdollista, että testaajien käyttöoikeudet eivät ole päivittyneet tai lainkaan voimassa testausajankohtana. Ennen testitilaisuuksia on hyvä tarkistaa testiympäristön ja käyttöoikeuksien toimivuus, koska kaikki myöhästymiset ja testitilaisuuksien siirrot vaikuttavat testausaikatauluun, jonka muutokset aiheuttavat mahdollisesti varsinaisen käyttöönottoajankohdan siirtymisen.

Testiryhmän jäsenille kannattaa järjestää testauksesta koulutus. Muuten voi jäädä esimerkiksi virhetilanteiden testaus liian vähälle. Ammattitaitoiset ihmiset syöttävät tarkasti oikein testaessaan, joka antaa hyvän kuvan normaalista, oikein siirtyvästä tapauksesta, mutta ei anna oikeaa kuvaa järjestelmän käyttäytymisestä virheellisesti syötetyn tiedon osalta. Tarvitavat testaajaresurssit on voitu arvioida liian vähäiseksi tai niihin tulee muutoksia muiden työtehtävien vuoksi. Tämä voi vähentää testausilaisuuksia, jolloin testaus ei ole toteutettu niin kattavasti kuin suunnitelmassa on edellytetty tai muuttaa käyttöönotto-aikataulua.

Kun onnistuneiden testausvaiheiden jälkeen otetaan tietojärjestelmä tuotantoon, on uusien toimintatapaprosessien käyttöönotto mahdollista. HEHA-projektissa oli tavoitteena tiedon syöttämisestä vain kerran käyttöönotettavaan järjestelmään, josta se siirtyi muuttumattomana muihin järjestelmiin. Tämä tavoite toteutuessaan vastaavanlaisissa käyttöönotoissa varmistaa tiedon pysymisen muuttumattomana ja tiedon oikea-aikaisuuden muissa järjestelmissä. Kaiken päällekkäistyön vähentäminen tuo oikeiden prosessien myötä yritykselle kustannussäästöjä, jotka ovat nykypäivänä yritystoiminnan kehittämiselle Suomessa perusedellytyksiä.

Kun tietojärjestelmä korvaa henkilöstön käyttämät käsintäytetyt lomakkeet uudessa toimintatapaprosessissa, tämän toteutumisen suurena uhkana on laaja muutosvastarinta. Aikaisessa vaiheessa ennen käyttöönottoa järjestetyt tiedotustilaisuudet järjestelmän tuleville käyttäjille auttavat vähentämään vastarintaa. Kun käyttäjät ymmärtävät, miksi toimintatapamuutos tehdään ja mitä hyötyä muutoksesta on, saadaan heidät suhtautumaan positiivisemmin tietojärjestelmän käyttöön.

Testausvaiheessa voi käydä niin, ettei ole osattu syöttää virheellisesti tietoa kaikilla mahdollisilla tavoilla. Kun kaikki loppukäyttäjät aloittavat järjestelmän käytön, voi tieto siirtyä virheellisessä muodossa toiseen järjestelmään aiheuttaen siellä virheellisen lopputuloksen, esimerkiksi henkilöstöhallinnon järjestelmästä palkanlaskentaohjelmaan siirtyvä virheellinen tieto aiheuttaa palkanmaksuvirheen. Järjestelmään syötetyn tiedon syöttörajoitteet kannattaa tarkistaa, jos käyttäjät toimivat saman tiedon syöttämisessä jatkuvasti väärin. Toisaalta virheellisten tietojen syöttämistavan vähentämiseksi on jo käyttöönottoaiheessa oltava mietittynä käyttäjien koulutuksen ja tuen järjestäminen. Nämä voivat olla toteutettu liian vähäisenä, kun ei ole ymmärretty käyttäjien tietoteknillisten taitojen tasoa tai kuinka kielteinen suhtautuminen järjestelmän käyttöön vaikuttaa tiedon oikeellisuuteen.

## 7.2 Käyttäjien koulutus ja tuki

Uuden tietojärjestelmän käyttöönotto onnistuu ja uudet toimintatapaprosessit omaksutaan nopeammin, jos mahdollisimman ajoissa projektin aikaisessa vaiheessa aloitetaan asiasta tiedottaminen käyttäjille. Projektin suunnitteluvaiheessa on hyvä myös tehdä alustava käyttäjien koulutussuunnitelma. Lisäksi viimeistään järjestelmätestausvaiheessa on aloitettava loppukäyttäjille annettavien käyttöohjeiden laatiminen, jotta ne saadaan testattua ennen varsinaista käyttöönottoa. Kun yrityksen kaikki työntekijät ovat tietojärjestelmän loppukäyttäjiä, on hyvä tehdä tietojärjestelmän käytettävyydestä ajoissa, jotta ennen käyttöönottoa ehditään parantaa käytettävyyttä.

### 7.2.1 Koulutus

HEHA-projektin tietojärjestelmän tapaisten järjestelmien käyttäjäkoulutusten järjestämisessä on huomioitava se, että osalla käyttäjistä on eri tilanteissa eri roolit. Esimies on loppukäyttäjä hakiessaan omaa kesälomaansa, mutta on päättäjä alaistensa vuosilomia käsitellessään. Käyttäjäkoulutukset kannattaakin järjestää käyttäjärooleittain, jotta prosessien eri vaiheet selkiytyvät käyttäjille oikein alusta alkaen. Jos koulutusresurssit ovat riittävät, niin koulutukset voi järjestää asiakokonaisuuksittain muutoksen omaksumisen helpottamiseksi.

Kirjassa Käytettävyyden psykologia on listattu keinoja, joilla autetaan käyttäjiä koulutuksessa:

- selvitä toimija, toiminnan tavoite, toimintaympäristö ja toimintamahdollisuudet.
- älä kouluta käyttäjää yli hänen tavoitteitaan.
- mieti kuuluuko työ pohdinnan tai kokemuksen toiminta-alueelle (anna pohtijalle selkeätä tietoa).

- näytä vaihtoehdot, jos käyttäjä joutuu tekemään muita kun rutiinivalintoja. Jos on vain yksi tapa edetä, selvitä voiko kone tehdä sen.
- anna selkeä palaute. Negatiivisessä palautteessa on kerrottava, mitä tapahtui ja miten pitää jatkaa.
- huolehdi, että eteneminen oikeaan suuntaan on tiedossa.
- kaikissa toimintavaiheissa on selvä aloituskohta, suoritusvaihe ja lopetusvaihe.
- työprosessia kuvaavasta käyttöohjeesta tulisi käydä ilmi työvaiheen alkupalaute, mitä pitää tehdä ja mikä on loppupalaute. (Sinkkonen ym. 2002, 73 - 75.)

### 7.2.2 Käyttäjien tuki

HEHA-projektin henkilöstöhallinnon tietojärjestelmän käyttöönoton yhteydessä järjestettiin nonstop-tukipisteet, joissa oli jatkuvasti paikalla henkilöstöasiainhoitajia asiantuntijoina auttamassa ja opastamassa tarvittaessa. Tällaisen tuen järjestäminen on hyvä keino saada uusi toimintatapaprosessi sujuvasti käyttöön ja muutosvastarinta vähenemään. Koko henkilöstön käyttämässä järjestelmässä on jatkuvasti uusia käyttäjiä, joten on hyvä varmistaa jatkuva käytöntuki esimerkiksi asiantuntijapuhelinpalveluna.

## 8 Arviointi ja loppuyhteenveto

Käyttöönoton testauksen onnistuminen riippuu ennakkoon hyvin laaditusta testaus-suunnitelmasta, testaajien kouluttamisesta ja hyvin toteutetusta sekä toimivasta testiympäristöstä. Kun otetaan koko henkilökunnan käyttöön tietojärjestelmä, kannattaa järjestelmätestauksen loppuvaiheessa ottaa loppukäyttäjiin kuuluvia henkilöitä testaukseen mukaan. Lyhyen opastuksen jälkeen he voivat testata järjestelmää itsenäisesti, ja näin saadaan mahdollisia puuttuvia syöttörajoituksia selville ennen käyttöönottoa.

Kirjassa Yrityksen tietohallinto kerrotaan, että projektin kulkuun vaikuttaa tietojärjestelmässä (tiedot) ja kehittämisprosessin (ihmiset) aikana tapahtuva vuorovaikutus. Järjestelmän käyttäjien tekemät tapahtumat, syöttötiedon kentät, tiedostot ja tietokannat, raportit ja muut tulosteet sekä käsittelysäännöt muodostavat järjestelmän määrällisen monimutkaisuuden. Projektin monimutkaisuus muodostuu esimerkiksi laitteisto- ja ohjelmistotoimittajista sekä useista käyttäjäyksiköistä. (Ruohonen & Salmela 2005, 169 - 170.)

Hyvin suunniteltuna suurikin tietojärjestelmähanke samanaikaisine liiketoimintaprosessien muutostavoitteineen on toteutettavissa, kun hankkeessa käytetään yrityksen omia osaajaresursseja ja huomioidaan ajoissa mahdolliset tarvittavat aikataululliset muutokset, tekniset vaikeudet, integrointiongelmat ja käyttäjien muutosvastarinta. HEHA-projektissa toteutuneen henkilöstöhallinnon tietojärjestelmän ja palkanlaskenta-ohjelmiston tietosisällön



ja -siirron integrointi on osoitus siitä, että tietotekniikassa voi jatkaa ohjelman elinkaarta liittämällä siihen toinen, uusien liiketoimintaprosessien vaatimuksia vastaava ietojärjestelmä.

**LÄHTEET**

Hovi, A., Hervonen, H. & Koistinen H. 2009. Tietovarastot ja Business Intelligence. Porvoo: WS Bookwell.

Kainulainen, A. 2008. Agile-menetelmät. Viitattu 17.4.2010.

<https://oa.doria.fi/handle/10024/39341>

Karkulehto, J. & Urpi, T. 2003. Opiskelijaliikkuvuuden hallinnan tuki. Viitattu 14.4.2010.

[http://www.palvelut.virtuaaliyliopisto.fi/LH\\_Testaussuunnitelma\\_versio\\_1.0.doc](http://www.palvelut.virtuaaliyliopisto.fi/LH_Testaussuunnitelma_versio_1.0.doc)

Pohjonen R. 2002. Tietojärjestelmien kehittäminen. Jyväskylä: Docendo Finland

Ruohonen, M. & Salmela, H. 2005. Yrityksen tietohallinto. 1.-3. painos. Helsinki: Edita Prima.

Sinkkonen, I., Kuoppala, H., Parkkinen J. & Vastamäki R. 2002. Käytettävyyden psykologia. Helsinki: Edita.

Sähköisen kaupankäynnin aapinen. 1999. TIEKE Tietoyhteiskunnan kehittämiskeskus ry.

Tulostettu 11.4.2010. [http://www.tieke.fi/Julkaisut/Oppaat\\_yrityksille/](http://www.tieke.fi/Julkaisut/Oppaat_yrityksille/)

tSoft-Ohjelmistotuotannon tietokeskus. 2007. Intgerointi- ja järjestelmätestaus. Viitattu

17.4.2010. <http://cs.joensuu.fi/tSoft/testaus.htm>

Vuori, M. 2009. Laadunvarmistus kehittyy. Systemityö 1/2009, 4 - 6.

**KAAVAT JA TAULUKOT**

Kaava 1: Vanha osoitemuutosprosessi .....	9
Kaava 2: Uusi osoitemuutosprosessi .....	9
Kaava 3: Vanha poissaolon hakuprosessi .....	10
Kaava 4: Uusi poissaolon hakuprosessi .....	10
Taulukko 1: Liittymän tiedon siirto .....	11
Taulukko 2: SWOT-esimerkki.....	20