



VAASAN AMMATTIKORKEAKOULU  
VASA YRKESHÖGSKOLA  
UNIVERSITY OF APPLIED SCIENCES

Tero Juhani Lyly

LEMONSOFT  
TOIMINNANOHJAUSJÄRJESTELMÄN  
KÄYTTÖÖNOTTO-OPAS

Tekniikka ja liikenne  
2010

## **ALKUSANAT**

Tämä opinnäytetyö kirjoitettiin Vaasan ammattikorkeakoulun tietotekniikan osaston opinnäytetyönä lukukauden 2009-2010 välisenä aikana. Opinnäytetyön toimeksiantaja oli Lemonsoft Oy. Työ toteutettiin itsenäisesti sekä Lemonsoft Oy:n tuella.

Lemonsoft Oy:n puolesta yhteyshenkilönä ja ohjaajana toimi yrityksen toimitusjohtaja Kari Joki-Hollanti. Vaasan ammattikorkeakoulun puolesta työn valvojana toimi yliopettaja Ghodrat Moghadampour.

Kiitokset tahdon esittää työni valvojalle, luokkatovereille, ystäville sekä Tekun-kellarille. Erityiset kiitokset haluan esittää Lemonsoft Oy:stä saadulle tuelle ja ohjaukselle.

Lämpimät kiitokset avopuolisolleni Sofialle, kun olet jaksanut tukea minua ja potkinut eteenpäin työn vaikeina hetkinä.

Vaasassa 3.6.2010

Tero Lyly

## VAASAN AMMATTIKORKEAKOULU

Tietotekniikan koulutusohjelma

**TIIVISTELMÄ**

Tekijä	Tero Lyly
Opinnäytetyön nimi	Lemonsoft toiminnanohjausjärjestelmän käyttöönotto-opas
Vuosi	2010
Kieli	suomi
Sivumäärä	74
Ohjaaja	Ghodrat Moghadampour

Tämä opinnäytetyö tehtiin Vaasassa sijaitsevalle Lemonsoft Oy:lle. Lemonsoft Oy on helmikuussa 2006 perustettu ohjelmistotalo, jonka tuotteena ovat toiminnanohjausratkaisut pk –sektorin yrityksille. Opinnäytetyön tavoitteena oli toteuttaa opas toiminnanohjausjärjestelmän käyttöönotosta. Opas on suunnattu Lemonsoftin asiakkaille, jotka ovat käyttöönottamassa Lemonsoft -toiminnanohjausjärjestelmää.

Työn teoriaosuudessa käsitellään perinteistä ohjelmistokehitysprosessia ja nykyajan ketteriä menetelmiä. Teoriaosuuden tarkoituksena on, että lukija ymmärtää mihin ohjelmistokehitysprosessin vaiheeseen tämä opinnäytetyö ajoittuu.

Työn toteutusosuudessa opasta lähdettiin toteuttamaan olemassa olevan käyttöönottoprosessin aktiviteettien pohjalta, jotka Lemonsoft oli aiemmin tunnistanut ja luovuttanut opinnäytetyön tekijän käyttöön. Toteutusosuuden tarkoituksena on, että asiakas saa kokonaiskuvan käyttöönottoprojektin eri aktiviteeteistä ja sisällöistä. Lisäksi opas on tukena silloin, kun asiakas on ottamassa käyttöön Lemonsoftin toiminnanohjausjärjestelmää.

Toteutetun oppaan tavoitteet saavutettiin hyvin. Aikataulullisesti projekti venyi alkuperäisestä suunnitelmasta

Asiasanat	ERP, Enterprise Resource Planning, toiminnanohjausjärjestelmä, käyttöönotto
-----------	---

VAASAN AMMATTIKORKEAKOULU

UNIVERSITY OF APPLIED SCIENCES

Tietotekniikan koulutusohjelma

## ABSTRACT

Authors	Tero Lyly
Title	Implementation of Lemonsoft Enterprise Resource Planning System Guide
Year	2010
Language	Finnish
Pages	74
Name of Supervisor	Ghodrat Moghadampour

---

This research was done for Lemonsoft Oy, a software company located in Vaasa. Lemonsoft Oy was established in the February 2006. Lemonsoft produces software applications for small and middle size companies. The objective of this thesis was to create a guide for customers, who are going to deploy the enterprise resource planning developed by Lemonsoft Oy.

In the theoretical part of the thesis the most popular classical and modern software development methods, like pure waterfall and agile are presented. The purpose of the theoretical part is to point out in what stage of the software development process, the deployment resides. The implementation part of the thesis is based on the preliminary information provided by Lemonsoft Oy and further research on the activities carried out at the company. The purpose of the implementation part was to give a broad perspective of the deployment phase of Lemonsoft Oy's enterprise resource planning system to the customers.

The research met all specified requirements and the customer is satisfied with the guide. Implementation part of the thesis was stretched from the original project planning.

---

Keywords	ERP, Enterprise Resource Planning, Software Project, Implementation
----------	---

## KÄYTETYT MERKINNÄT JA LYHENTEET

AD	on lyhenne englanninkielisistä sanoista <i>Active Directory</i> , jolla tarkoitetaan suomeksi keskitettyä käyttäjienhallintaa.
APS	on lyhenne englanninkielisistä sanoista <i>Advanced Scheduling and Planning</i> , jolla tarkoitetaan suomeksi tuotannonkuormituksen suunnittelemista.
CIM	on lyhenne englanninkielisistä sanoista <i>Computer Integrated Manufacturing</i> , jolla tarkoitetaan suomeksi tietokoneellisesti ohjattua tuotannonohjausta.
CRM	on lyhenne englanninkielisistä sanoista <i>Customer Relationship Management</i> , jolla tarkoitetaan suomeksi asiakkuudenhallintaa.
ERP II	on kehitetty versio ERP:stä, jonka toimintamalleihin on lisätty sovellukset toimitusketjunhallinta, asiakkuudenhallinta ja sähköisen kaupan välineitä sekä sosiaaliseen verkottumisen työkaluja.
ERP	on lyhenne englanninkielisistä sanoista <i>Enterprise Resource Planning</i> , jolla tarkoitetaan suomeksi toiminnanohjausjärjestelmää.
FTP	on lyhenne englanninkielisistä sanoista <i>File Transfer Protocol</i> , jolla tarkoitetaan suomeksi tiedonsiirtomenetelmää kahden laitteen välillä.
HRM	on lyhenne englanninkielisistä sanoista <i>Human Resource Management</i> , jolla tarkoitetaan suomeksi henkilöstönhallintaa.

JIT	on lyhenne englanninkielisistä sanoista <i>Just In Time</i> , jolla tarkoitetaan suomeksi teollisuudessa käytettyä varastonhallintaa ja tuotannonohjausstrategiaa, jonka tarkoituksena on parantaa tehokkuutta.
Lean	on Toyotan kehittämä johtamisfilosofia, jonka tarkoituksena oli parantaa mm. asiakastytyväisyyttä, laatua ja pienentämään kustannuksia.
Moduuli	on toiminnanohjausjärjestelmän osa. Esimerkiksi tässä tapauksessa taloushallinto, joka sisältää käytettävät sovellukset laskutus- ja myyntireskontran, rekisterit sekä verkkolaskutuksen.
MRP I	on lyhenne sanoista <i>Material Requirements Planning</i> , jolla tarkoitetaan suomeksi materiaalinhallintajärjestelmää.
MRP II	on lyhenne englanninkielisistä sanoista <i>Manufacturing Resource Planning</i> ja tarkoittaa suomeksi tuotannonohjausjärjestelmää.
Ohjelmisto	on toiminnanohjausjärjestelmä kokonaisuudessa, joka koostuu käyttöönotetuista moduuleista, esimerkiksi taloushallinnosta, henkilöstöhallinnosta ja asiakkuudenhallinnasta.
Palvelin	on tietoliikenteen yhteydessä tietokoneessa suoritettavaa palvelinohjelmistoa sekä tällaista ohjelmistoa suorittavaa tietokonetta. Tunnetaan myös yleisesti käsitteillä <i>servu</i> tai <i>serveri</i> .
Palvelu	on toiminnanohjausjärjestelmän rajapinnasta ajettava palvelu, jonka kautta voidaan välittää tietoa järjestelmän ulkopuolisille sovelluksille.

Partneri	on yrityksen yhteistyökumppani, esimerkiksi kouluttaja.
PK –yritys	tarkoittaa pientä ja keskisuurta yritystä.
SCM	on lyhenne englanninkielisistä sanoista <i>Supply Chain Management</i> , jolla tarkoitetaan suomeksi toimitusketjunhallintaa.
Sovellus	on toiminnanohjausjärjestelmän moduulista ajettava aliohjelma esimerkiksi laskun kirjaaminen järjestelmään.
SQL	on lyhenne englanninkielisistä sanoista <i>Structured Query Language</i> , jolla tarkoitetaan suomeksi kyselykieltä, jonka avulla relaatiotietokantapalvelimelle voidaan tehdä erilaisia hakuja, muutoksia ja lisäyksiä.
Toimittaja	tarkoittaa järjestelmän toimittavaa tahoa.
TVM	on lyhenne englanninkielisistä sanoista <i>Total Value Management</i> , jolla tarkoitetaan suomeksi organisaation arviointi tapaa pitkän-ajan vaikutuksista.
Työasema	on tietokone, jonka avulla käyttäjä voi työskennellä esimerkiksi tuottaa materiaalia tai hakea tietoa.
WEB	on lyhenne englanninkielisistä sanoista <i>World Wide Web</i> , jolla tarkoitetaan suomeksi maailmanlaajuista tietoverkkoa, jossa voidaan jakaa informaatiota käyttäjille.

## SISÄLLYS

ALKUSANAT .....	2
TIIVISTELMÄ .....	3
ABSTRACT .....	4
KÄYTETYT MERKINNÄT JA LYHENTEET .....	5
1 JOHDANTO .....	13
1.1 Opinnäytetyön tavoite .....	13
1.2 Opinnäytetyön rajaus .....	13
2 LEMONSOFT OY .....	14
2.1 Yritys .....	14
2.2 Tuotteet .....	14
2.3 Asiakkaat .....	15
3 TOIMINNANOHJAUSJÄRJESTELMÄT .....	16
3.1 Historia .....	16
3.2 ERP II .....	17
3.3 Toiminnanohjausjärjestelmän hyödyt .....	18
3.4 Toiminnanohjausjärjestelmän käyttöönotto .....	18
3.5 Toiminnanohjausjärjestelmän käyttöönoton ongelmat .....	19
4 PERINTEINEN OHJELMISTOKEHITYSPROSESSI .....	20
4.1 Ohjelmistokehitysprosessin elinkaari .....	20
4.2 Ohjelmistokehitysprosessin yleiset aktiviteetit .....	21
4.2.1 Vaatimukset .....	21
4.2.2 Määrittely .....	22
4.2.3 Suunnittelu .....	22
4.2.4 Toteutus .....	23



4.2.5	Integrointi.....	23
4.2.6	Testaus .....	23
4.2.7	Käyttöönotto .....	24
4.2.8	Ylläpito .....	24
4.3	Ohjelmistokehityksen prosessimalleja .....	24
4.3.1	Vesiputousmalli .....	25
4.3.2	Prototyypimalli .....	25
4.3.3	Inkrementaalinen malli .....	27
4.3.4	Spiraalimalli.....	28
4.3.5	RUP –malli.....	29
5	KETTERÄT MENETELMÄT .....	31
5.1	Yhteiset piirteet .....	31
5.2	Ketteriä menetelmiä .....	31
5.2.1	Scrum .....	32
5.2.2	Agile Modeling.....	33
6	HANKINTAPROSESSI.....	38
6.1	Valintaprosessi .....	39
6.1.1	Hankintamenettelyt.....	39
6.1.2	Toimittajien kanssa asiointi .....	40
6.1.3	Julkiset hankinnat.....	40
6.2	Käyttöönottoprosessi.....	41
6.3	Varsinainen käyttö .....	42
6.3.1	Käyttöönottamattomat moduulit.....	42
6.3.2	Ohjelmistopäivitykset .....	42
6.3.3	Räätälöinnit.....	42

7	KÄYTTÖÖNOTTOVAIHE.....	43
7.1	Käyttöönottosuunnitelma .....	43
7.1.1	Henkilöstöressurit ja tehtävät .....	43
7.1.2	Aikataulu.....	43
7.1.3	Hankkeen vaiheistus ja päätöksentekopisteet .....	43
7.1.4	Budjetti.....	44
7.1.4.1	Kustannusarviointi.....	45
7.1.4.2	Projektibudjetti .....	45
7.1.4.3	Kustannusvalvonta .....	46
7.1.5	Tiedottaminen ja tiedotussuunnitelma .....	46
7.1.6	Riskien hallinta .....	47
7.1.7	Koulutussuunnitelma .....	47
7.2	Projektiorganisaatio.....	48
7.2.1	Työryhmä ja hanke .....	48
7.2.2	Hanke .....	48
7.2.2.1	Hankkeen omistaja .....	48
7.2.2.2	Hankkeen ohjausryhmä .....	49
7.2.2.3	Projektipäällikkö.....	49
7.2.2.4	Hankkeen muut jäsenet.....	50
7.3	Valmistelut .....	50
7.3.1	Tietojärjestelmän infrastruktuuri .....	51
7.3.2	Yrityksen toimintaprosessit .....	51
7.3.3	Ohjelmistolisenssit.....	51
7.3.4	Yrityksen hoidettavat aktiviteetit.....	52
7.3.4.1	Varmuskopiointi .....	52
7.3.4.2	Dokumentointi.....	52
7.3.4.3	Häiriötilanteisiin varautuminen .....	53
7.3.4.4	Testiympäristö .....	53
7.3.5	Laitteisto ja ohjelmistovaatimukset .....	53

7.3.5.1	Palvelinvaatimukset.....	54
7.3.5.2	Työasemavaatimukset .....	54
7.3.5.3	Tietoliikennevaatimukset .....	55
7.3.5.4	Microsoft SQL Server .....	56
7.3.5.5	Microsoft SQL Server Express Edition -rajoitukset.....	57
7.4	Asennusvaihe .....	57
7.4.1	Asennuspaketit.....	58
7.4.2	Asennusprosessi.....	59
7.4.2.1	Lemonsoft ohjelmiston asennus työasemalle .....	59
7.4.2.2	Lemonsoft ohjelmiston asennus palvelimelle.....	60
7.4.3	Käyttäjät.....	60
7.4.3.1	Windows –autentikaatio .....	60
7.4.3.2	Tietokanta –autentikaatio .....	61
7.5	Tiedonsiirrot.....	61
7.5.1	Tiedonsiirtotavat .....	62
7.5.2	Tiedonsiirtorajapinnat .....	62
7.5.3	Lopullinen tiedonsiirto.....	63
7.6	Muutokset.....	63
7.6.1	Pilotointi.....	63
7.6.2	Ohjelmamuutokset .....	64
7.6.3	Muutostenhallinta .....	64
7.6.4	Ohjelmamuutosten lisääminen järjestelmään .....	64
7.6.5	Hyväksymistestaus.....	64
7.7	Koulutus .....	65
7.7.1	Esimerkki koulutusrungosta.....	65
7.7.2	Käyttöohjeet.....	67
7.8	Varsinainen käyttöönotto .....	67
7.8.1	Vanha järjestelmä.....	67

7.8.2	Uusi järjestelmä .....	68
8	YHTEENVETO .....	69
	LÄHTEET.....	70

# 1 JOHDANTO

Johdanto sisältää johdatuksen opinnäytetyöhön ja tavoitteiden esittelyn sekä rajauksen. Lisäksi luvussa 2. esitellään lyhyesti yritys ja sen tuotteet sekä millä toimialalla yrityksen asiakkaat toimivat. Lopuksi luvussa 3. kerrotaan hieman toiminnanohjausjärjestelmien historiasta ja sen kehittymisestä nykypäivään sekä mitä ohjelmistoja pk-sektorilla yleisesti käytetään tai tunnetaan.

## 1.1 Opinnäytetyön tavoite

Opinnäytetyön tavoitteena oli toteuttaa vaasalaiselle Lemonsoft Oy:lle opas käyttöönottosuunnitelmasta, joka koskee heidän ERP -tuotetta. Toteutettua opasta tutkittiin kirjallisen tiedon, Internetistä löydetyn tiedon ja tämän työn toimeksiantoyrityksen koulutusmateriaalien pohjalta. Oppaan tarkoituksena on varmistaa asiakkaan kannalta se, että ohjelmisto saadaan hallitusti ja tehokkaasti käyttöön yrityksessä. Lisäksi oppaalla halutaan helpottaa asiakasta ohjelmiston tai palvelun vaatimusmäärittelyn tarkastamisessa.

## 1.2 Opinnäytetyön rajaus

Tämä opinnäytetyö on rajattu koskemaan Lemonsoft -toiminnanohjausjärjestelmän käyttöönottoprojektia. Aihealueen kohdentamisen ja tarkentamisen vuoksi työssä esitellään perinteisesti tunnettua ohjelmistokehitysprosessia niin, että lukija saisi käsityksen ohjelmistokehityksen vaiheista sekä siitä, mihin vaiheeseen tämä opinnäytetyö keskittyy toteutusosassa.

## **2 LEMONSOFT OY**

### **2.1 Yritys**

Lemonsoft Oy on kotimainen kasvava ohjelmistotalo, joka on perustettu helmikuussa vuonna 2006. Yrityksen päätuotteena ovat Lemonsoft –ohjelmistot, jotka on suunnattu pääasiassa pk –sektorin yrityksille. Yritys keskittyy Lemonsoft ERP –ohjelmiston kehittämiseen ja siihen liittyvien palveluiden tuotteistamiseen. Tuotteiden myynnistä vastaa koko maan kattava jälleenmyyjäverkosto, jossa on lähes kolmekymmentä jäsentä.

Yrityksen tavoitteena on olla toiminnanohjausmarkkinoiden ykkönen kotimaan pk-yritysten keskuudessa, ja lisäksi tavoitteena on olla markkinoiden nopeimmin kasvava toimija toiminnanohjausmarkkinoilla. Yrityksen toimintastrategiaan kuuluu Lemonsoft –ohjelmistojen myynti ja käyttöönottoprojektit. (Lemonsoft Oy 2010 c.)

Yrityksen päätoimipiste sijaitsee Vaasassa, Runsorissa. Päätoimipisteen lisäksi yrityksellä on toimipiste Joensuussa. Yritys työllistää tällä hetkellä 14 henkilöä. Liikevaihto vuonna 2009 oli noin 700 000 euroa ja vuonna 2010 tavoitteena on saavuttaa 1,3 miljoonan euron liikevaihto.

Tulevaisuudessa yrityksen on tarkoitus laajentaa toimintaansa ulkomaille. Ensimmäisenä yritys aikoo aloittaa liiketoiminnan laajentamisen pohjoismaista, joista ensimmäisenä on Ruotsi. (Joki-Hollanti 2010.)

### **2.2 Tuotteet**

Lemonsoft ERP on kokonaisratkaisu, joka koostuu lukuisista käyttöönotettavista ja erilaisilla varioitavista moduuleista. Moduuleja ovat: taloushallinto, asiakkuudenhallinta, henkilöstöhallinto, logistiikka, tuotannonohjaus, johdon työkalut, projektinhallinta, rajapinnat, WEB-portaalit, mobiiliratkaisut sekä muut toimialakohtaiset ratkaisut. (Lemonsoft Oy 2010 c.)

### **2.3 Asiakkaat**

Lemonsoftin asiakkaita ovat tuotannolliset yritykset, tukkuliikkeet, maahantuojat, erikoistavara-kaupat, huoltopalveluyritykset, tilitoimistot sekä projektiliiketoimintaa tekevät yritykset. Lemonsoft –ohjelmisto soveltuu yrityksiin, joissa käyttäjien määrä on muutamasta henkilöstä noin sataan henkilöön. (Lemonsoft Oy 2010 d.)

### 3 TOIMINNANOHJAUSJÄRJESTELMÄT

ERP-järjestelmät eli toiminnanohjausjärjestelmät kehittyivät MRP II-järjestelmistä, jotka kehitettiin hallitsemaan organisaatioiden tuotannosuunnittelua, ohjausta ja varastoarvoja.

#### 3.1 Historia

Tiedetään, että ERP järjestelmän esiversioita oli olemassa vuonna 1880, jolloin tallennettiin tietoja reikäkortteihin. (Tieke 2008.) ERP-järjestelmät kehittyivät 1960- ja 1970-luvuilla varastonhallintaohjelmistoista edistyneempiin materiaalinhallintaohjelmistoihin (MRP, *Materials Requirements Planning*). MRP – ohjelmistojen avulla voitiin ennustaa materiaalien tarvetta tilausten ja myyntiennusteiden pohjalta. Ohjelmistot eivät olleet täydellisiä, joten jouduttiin selvittämään myyntiosastolta myyntiennusteet ja suunnittelemaan tuotantoa vastaamaan laskettuja ennusteita sekä laskemaan raaka-ainetarpeet ja tilaamaan ne toimittajilta. (Karessuo 2003.)

Oliver Wight esitteli 1980 luvun alussa MRP seuraajan, MRP II eli *Manufacturing Resource Planning* –järjestelmän, jonka avulla voitiin laskea tarkemmin edellä mainittuja ennusteita. Kertaalleen laskettu tieto oli heti jokaisen osaston saatavilla ja hyödynnettävissä jokaisen osaston omissa prosesseissa. Tätä informaation jakamista voidaankin pitää varsinaisena esiasteena toiminnanohjausjärjestelmille. (Karessuo 2003.)

1980-luvulla järjestelmiin tuotiin ominaisuuksia, jotka olivat etenkin autoteollisuudelle tärkeitä. Ominaisuutta kutsuttiin nimellä JIT (*Just In Time*), jonka nimestä voi päätellä, että järjestelmä tavoitteli tehokasta tuotannonohjausta mahdollisimman pienellä varastoarvolla ja eräkoolla.

1990-luvulla Toyota otti käyttöön Lean-toimintamallin, josta käytetään myös nimiä *Flow* ja *Demand-Pull*. Lean –malli tähtäsi parempaan tehokkuuteen, poistaen tuotannon tilauskuormia ja synkronoimaan tuotantoa vastaamaan asiakkaiden tarpeita kuin pidemmän aikavälin ennustuksille, jotka olivat usein epätarkkoja. (Allbusiness 2001.) JIT:n periaatteet olivat mukana Lean –toimintamallissa, mutta Lean sisälsi JIT –mallista



poiketen tuotannon lisäksi myös kaikki muut yrityksen toiminnot. (Karessuo 2003.) Lisäksi otettiin käyttöön automatisointia ja integrointia korostava CIM (*Computer Intergrated Manufacturing*) -menetelmä, joka ehti tuoda yrityksiin uusia ajatuksia ja toimintatapoja.

Varsinaiset ERP -järjestelmät syntyivät 1990-luvun alussa, kun Gartner Group arvioi resurssinhallintaohjelmistoja. Järjestelmien katsottiin kehittyneen niin paljon, että niistä voitiin käyttää nimeä ERP (*Enterprise Resource Planning*), josta suomenkieliseksi versioksi yleistyi termi toiminnanohjausjärjestelmä.

ERP-järjestelmien yleistyminen ja käyttöönotto jatkui 1990-luvun alussa, jolloin taloudellinen tilanne pakotti yrityksiä tehostamaan toimintaansa ja pysymään markkinoilla. Koveneva kilpailu pakotti yrityksiä ottamaan paremmin asiakkaat huomioon, lyhentämään toimitusaikoja ja tehostamaan toimintaa. Tänä päivänä ERP -järjestelmät eivät ole ainoastaan tuotantoyrityksille suunnattuja. ERP on nykyään hyödyllinen kaikille yrityksille ja yhteisöille, joilla on tarvetta integroida toimintoja yhdeksi ja helpoksi hallittavaksi kokonaisuudeksi. (Karessuo 2003.)

### **3.2 ERP II**

ERP II-määritelmä on saanut alkunsa, kun ERP -järjestelmien käyttäjät toivoivat sovelluksia toimitusketjun hallintaan, asiakkuuden hallintaan ja sähköisen kaupankäynnin toimintoihin. Aluksi markkinoille tuli yksittäisiä järjestelmäratkaisuja, mutta myöhemmin järjestelmätoimittajat ovat lisänneet tällaisia ominaisuuksia omiin tuotteisiinsa. ERP II on määritelty kaavalla  $ERP + SCM + CRM = ERP II$  eli kyseessä on vanha järjestelmä, johon on lisätty toimitusketjun- ja asiakkuuden hallintaan liittyviä näkökulmia. Yleistetysti, ERP II tarkoittaa ohjelmistoja, joiden avulla yritykset voivat jakaa informaatiota yhteistyökumppaneidensa kanssa Internetin välityksellä. Muita ERP II sisältämiä laajennuksia ovat APS (*Advanced Planning and Scheduling*), jonka avulla tarkkaillaan materiaalin-, työvoiman- ja tuotannon kuormitusta sekä TVM (*Total Value Management*), jonka tarkoituksena on optimoida yrityksen investointeja ja määrittää hinta kaikille sähköistä liiketoimintaa koskeville muutoksille. (Karessuo 2003.)

### **3.3 Toiminnanohjausjärjestelmän hyödyt**

Kaikilla nykyajan yrityksille on jonkinlaisia liiketoimintaa ohjaavia järjestelmiä. Toiminnanohjausjärjestelmän tehtävänä on yhtenäistää nämä yhteen järjestelmään ja nopeuttaa yrityksessä tapahtuvaa toimintaa. Tärkeimmäksi hyödyksi katsotaan eri toimintojen reaaliaikaiset seurantamahdollisuudet, joiden avulla tiedetään heti miten yrityksellä menee ja voidaan ennustaa esimerkiksi materiaalihankintoja tilausten perusteella. Informaatio on hyödynnettävissä heti kaikkialla, kun se on kertaalleen syötetty järjestelmään. Toimintoja voidaan automatisoida kuten esimerkiksi tarjouksen tekeminen asiakkaalle: asiakas hyväksyy tarjouksen, jonka jälkeen tarjous siirtyy avoimeksi tilaukseksi. Tilaus käsitellään ja tuotteet toimitetaan asiakkaalle, jonka jälkeen tilaus siirtyy laskutukseen sekä sieltä myyntireskontraan ja lopuksi kirjanpidon hoidettavaksi. Toimintoketjun aikana varastosaldo päivittyy siten, että tuotteita on vähentänyt kyseisen tilauksen verran varastosta. (Tieke 2008.)

### **3.4 Toiminnanohjausjärjestelmän käyttöönotto**

Toiminnanohjausjärjestelmän käyttöönottoaminen on vaativaa ja työlästä. Käyttöönottovaiheen tärkein vaihe on henkilöstön kouluttaminen, jonka osuus kustannuksissa on suurempi kuin varsinainen järjestelmä. Käyttöönoton ongelmana on saada yrityksen prosessit integroitua osaksi järjestelmää. Tästä syystä yrityksen on oltava valmis muokkaamaan omia prosessejaan, jotta toiminnanohjausjärjestelmästä saadaan kaikki hyöty irti. Valmisohjelman käyttöönottoaminen on yritykselle edullisempaa kuin lähteä räätälöimään ohjelmistoa vastaamaan yrityksen omia prosesseja, koska yrityksen on edullisempaa lähteä muuntamaan omia prosessejaan, jotta toiminnanohjaus toimisi yrityksen logiikan mukaan. Valmisohjelmien myötä erilaiset integraatiot ohjelmiin ovat mahdollisia, ja näin toiminnanohjausjärjestelmästä saadaan enemmän hyötyjä. Toiminnanohjausjärjestelmän käyttöönottoprosessi on jatkuva, ja sen tarkoituksena on kehittää yrityksen toimintaa tehokkaammaksi. (Tieke 2008.)

### 3.5 Toiminnanohjausjärjestelmän käyttöönoton ongelmat

Jokaisessa käyttöönottoprojektissa on ongelmia, jos etukäteen ei ole mietitty käyttöönottoon liittyviä riskejä tai siihen liittyviä ongelmia. Ongelmia aiheuttavat käyttöönotettava ohjelmisto, laitteisto ja järjestelmän tulevat käyttäjät.

Tietojärjestelmän suunnittelijoiden ongelmia ovat:

- organisaatio tunnetaan huonosti
- suunnittelijoiden ja johdon yhteys puuttuu
- käyttäjien toimenkuvat epäselviä
- järjestelmän käyttäjät eivät osallistu käyttöönottoon ja muutosvastarinta
- kustannusarviointi ei ole realistinen.

Tietojärjestelmän käyttäjien ongelmia ovat:

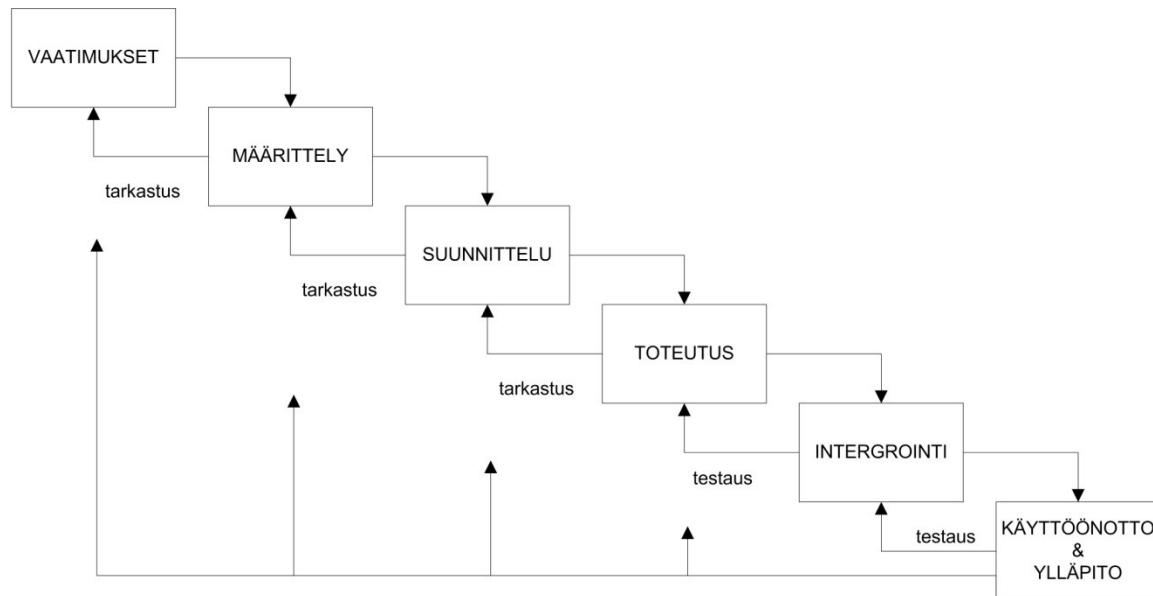
- tiedon syöttäminen
- käyttäjien tarpeet unohdettu rakennusvaiheessa
- tietojärjestelmä ei palvele toimintaa
- tietojärjestelmä aiheuttaa ongelmia organisaation toiminnassa
- puutteellinen koulutus. (Grotenfelt, Outi & Ilomäki, Liisa & Närvänen, Liisa 1989.)

## 4 PERINTEINEN OHJELMISTOKEHITYSPROSESSI

Tässä ja seuraavassa 5. luvussa esitellään ohjelmistokehitysprosessia. Tarkoituksena on antaa kokonaiskuva ohjelmistokehityksen vaiheista ja kertoa mihin vaiheeseen tämä opinnäytetyön aihe ajoittuu ohjelmistokehityksessä. Ensimmäiseksi kerrotaan lyhyesti ohjelmistokehityksen elinkaaresta ja sen jälkeen esitellään ohjelmistokehityksen yleisiä aktiviteetteja. Lopuksi esitellään erilaisia prosessimalleja. Lisäksi esitellään nykyajan nopean sovelluskehityksen metodologia eli minkälaisia ketteriä menetelmiä hyödynnetään ohjelmistokehityksessä.

### 4.1 Ohjelmistokehitysprosessin elinkaari

Ohjelmistokehitysprosessin elinkaarella (*Software Development Life Cycle*) tarkoitetaan ajanjaksoa, joka kuvaa ohjelmiston kehittämisen aloittamisesta sen poistamiseen käytöstä. Ohjelmiston kehittämisessä olennaista on, että sitä kehitetään vaiheittain. Tavallisimmin ohjelmistohankkeet on vaiheistettu hyödyntäen vesiputousmallia (engl. *Waterfall Model*), joka on havainnollistettu kuvassa 1. Vesiputousmallista on olemassa erilaisia muunnelmia, mutta näillä malleilla on yhteistä se, että niistä voidaan erottaa helposti määrittely-, suunnittelu- ja toteutusvaiheet. Ennen määrittelyvaihetta, tehdään yleensä tarvekartoitukseksi (engl. *Requirements study*) kutsuttu vaihe. (Haikala & Märijärvi 2004.)



**Kuva 1. Esimerkki vesiputouksmallista (Haikala & Märijärvi 2004, 36).**

Kuvan 1. mukaisesti jokaiseen vaiheeseen liittyy toimenpiteitä, joiden tehtävänä on varmistaa, että ohjelmisto on laadukas. Näitä toimenpiteitä ovat tarkastukset, katselmukset ja testaaminen. Tarkastuksilla ja testauksella pyritään poistamaan virheet järjestelmästä mahdollisimman aikaisessa vaiheessa ja katselmuksia pidetään vaiheiden päätteeksi, joissa todetaan hankkeen tilanne ja se, että kaikki vaiheeseen liittyvät tavoitteet on saavutettu ja kaikki sovitut dokumentit on tuotettu. (Haikala & Märijärvi 2004.)

## 4.2 Ohjelmistokehitysprosessin yleiset aktiviteetit

Seuraavaksi esitellään ohjelmistokehitysprosessin yleisimpiä aktiviteettejä. Tarkoituksena on antaa kokonaiskuva ohjelmistokehitysprosessin kulusta ja vaiheista sekä osoittaa, mihin kohtaan opinnäytetyöaihe ajoittuu.

### 4.2.1 Vaatimukset

Ohjelmiston tai järjestelmän vaatimuksia lähdetään yleensä selvittämään tekemällä tarvekartoitus eli esitutkimus, jolla pyritään selvittämään järjestelmän vaatimukset. Vaatimukset asettaa ohjelmiston tilaaja eli asiakas, jolle ohjelmistoa ollaan kehittämässä. Tällaiset vaatimukset ovat asiakasvaatimuksia, joilla määritellään asiakkaan tarpeet.

Huomioitavaa on, että näillä vaatimuksilla ei ole mitään tekemistä sen kanssa millainen järjestelmä täyttää asiakkaan vaatimukset. Esitutkimuksen tehtävänä on etenkin vastata kysymykseen miksi ohjelmisto tai järjestelmä tulisi tehdä, mutta toisaalta myös miksi hanketta ei pitäisi toteuttaa. Esitutkimus on ohjelmiston lopputuloksen kannalta tärkein vaihe, koska harvoin väärillä asiakasvaatimuksilla päädytään tyydyttävään lopputulokseen. Esitutkimuksen tekeminen on myös osittain osa määrittelyvaihetta, koska asiakastarpeiden analysointi ja tarkentaminen jatkuu koko määrittelyvaiheen ajan. (Haikala & Märijärvi 2004.)

#### **4.2.2 Määrittely**

Määrittelyvaiheessa asiakasvaatimukset analysoidaan ja niistä johdetaan ohjelmistovaatimukset. Määrittelyn tuloksena on dokumentti, jota kutsutaan toiminnalliseksi määrittelyksi (engl. *Functional specification*). Toiminnalliseen määrittelyyn on kuvattuna ohjelmiston toiminnot, joita ovat käyttöliittymä ja kommunikointi ulkopuolisten järjestelmien kanssa. Lisäksi toiminnallinen määrittely sisältää ei-toiminnallisia vaatimuksia, joita ovat suoritusteho, vasteaika ja käytettävyys sekä lopuksi rajoitukset, joita ovat esimerkiksi käytettävissä oleva muistitila tai ohjelmointikieli. (Haikala & Märijärvi 2004.)

#### **4.2.3 Suunnittelu**

Suunnitteluvaiheella vastataan kysymykseen, miten määrittelyvaiheen pohjalta syntyneet järjestelmän toiminnot toteutetaan. Suunnitteluvaihe jaetaan tyypillisesti kahteen tai useampaan osaan. Suunnittelemisen aloitetaan arkkitehtuurisuunnittelusta, jossa järjestelmä jaetaan pienempiin itsenäisiin moduuleihin. Arkkitehtuurisuunnittelusta syntyvä dokumentti on tekninen määrittely. Suunnittelua jatketaan moduulisuunnitteluvaiheella, jossa jokaisen moduulin sisäinen rakenne suunnitellaan. Tyypillinen moduuli sisältää tietomäärittelyitä, tietoa käsitteleviä funktioita ja on kooltaan alle 1000 riviä ohjelmakoodia. (Haikala & Märijärvi 2004.)

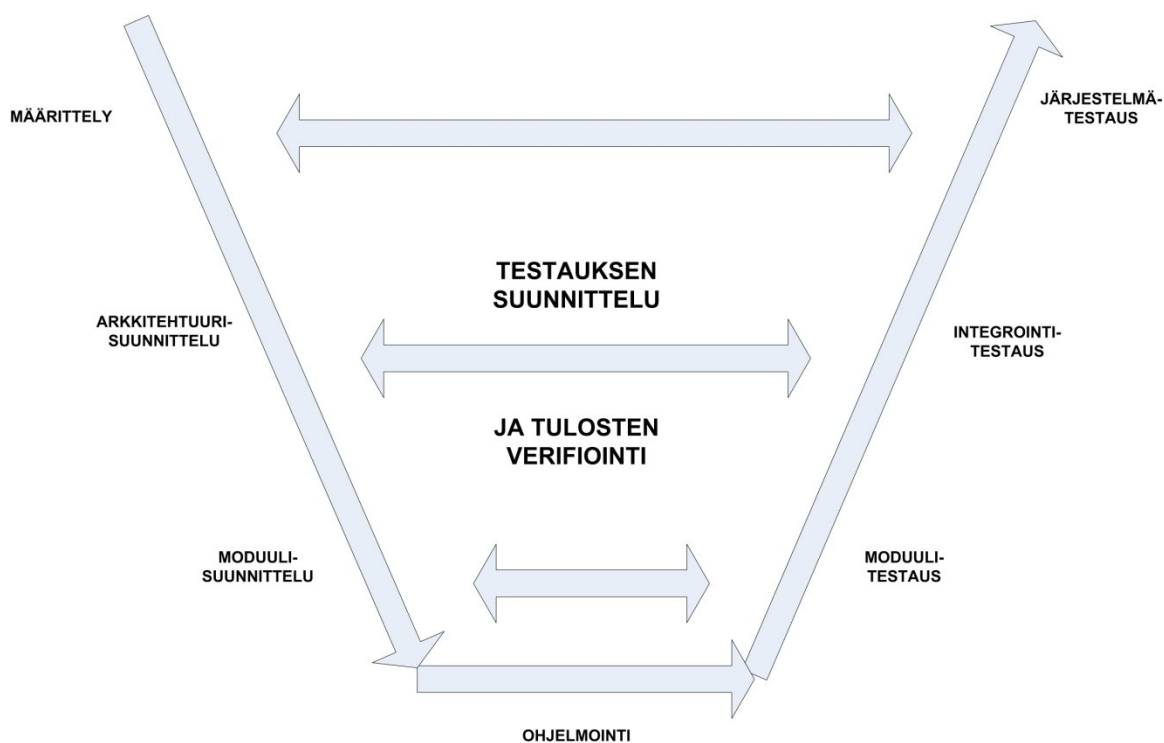
#### 4.2.4 Toteutus

Toteutusvaiheessa ohjelmoidaan ensimmäiseen virheettömään käännökseen saakka, kunnes suunnitellut toiminnot on kirjoitettu ohjelmakoodimuotoon ja kääntäjä ei ilmoita virheistä tai varoituksista. (Haikala & Märijärvi 2004.)

#### 4.2.5 Integrointi

Integrointivaiheessa yhdistetään toteutusvaiheessa tuotetut moduulit toisiinsa, joista muodostuu suurempia ja toiminnallisia kokonaisuuksia. (Haikala & Märijärvi 2004.)

#### 4.2.6 Testaus



Kuva 2. Testauksen V-malli (Haikala & Märijärvi 2004).

Testausvaiheessa pyritään löytämään virheitä ohjelmistosta. Testausta toteutetaan monella eri tasolla V –mallin mukaisesti. Kuvan 2 V –mallissa testaus on jaettu kolmeen tasoon: moduuli-, integrointi ja järjestelmätestaukseen. Moduulitestauksessa etsitään vikoja yksittäisistä moduuleista. Integrointitestauksessa etsitään vikoja moduulien

yhteistoiminnasta. Järjestelmätestauksessa etsitään vikoja koko järjestelmän toiminnoista ja suorituskyvyistä. (Haikala & Märijärvi 2004.)

#### **4.2.7 Käyttöönotto**

Asiakasprojekteissa käyttöönotto ja ylläpidon organisointi on ensisijaisen tärkeää, koska (Haikala & Märijärvi 2004.) käyttöönoton suunnittelu ja valmisteluvaiheen tarkoituksena on varmistaa, että itse käyttöönotto sujuu ongelmitta ja aikataulun mukaisesti. (Murch 2002.) Käyttöönottovaihe pitää sisällään useita samanaikaisestikin suoritettavia aktiviteettejä, kuten laitteiden ja ohjelmistojen asennusta, loppukäyttäjien ja tukihenkilöiden koulutusta, tiedottamista ja käyttöohjeiden laatimista. (TTL-julkaisusarja 2002.) (Lisää käyttöönotosta voit lukea kohdasta 6.) Tämän opinnäytetyön aihe on rajattu koskemaan pelkkää käyttöönottoa.

#### **4.2.8 Ylläpito**

Tutkimusten mukaan ylläpidon prosentuaalinen osuus koko järjestelmän elämänkaaresta vaihtelee 40-80 prosentin välillä, näin ollen ylläpito vaatii hyvin paljon resursseja yritykseltä, joka on ottamassa käyttöön uutta järjestelmää. Ylläpitotyön tarkoituksena on seurata palveleeko järjestelmä käyttäjää, ovatko asetetut vaatimukset saavutettu ja ajan tasalla. Ylläpitotyö käsitteenä on laaja käsite, koska se sisältää kaiken työn, jota järjestelmän eteen tehdään käyttöönoton jälkeen. Ylläpitotyö jaetaan kahteen osa-alueeseen: järjestelmän kunnossapito- ja kehittämistyöhön. Kehittämistyötä on kokonaan uusien järjestelmäratkaisujen tekeminen sekä olemassa olevan järjestelmäratkaisun muuntaminen. Laajahkojen muutosten tekeminen edellyttää tekijöiltä koko tietojärjestelmän tasolla tapahtuvaa suunnittelua. Kunnossapitotyötä ovat järjestelmässä esiintyvien vikojen syiden selvittelyä ja korjaamista sekä järjestelmän toimintakunnon ylläpitämistä. (Grotenfelt, Outi ym. 1989.)

### **4.3 Ohjelmistokehityksen prosessimalleja**

Vesiputousmallin lisäksi on olemassa muita elinkaarimalleja, joita voidaan hyödyntää ohjelmistokehityksessä. Näitä ovat muun muassa protoilumallit ja inkrementaaliset mallit, niin sanottu Evo-malli (*evolutionary delivery*). Kevyempiä menetelmiä ovat Agile

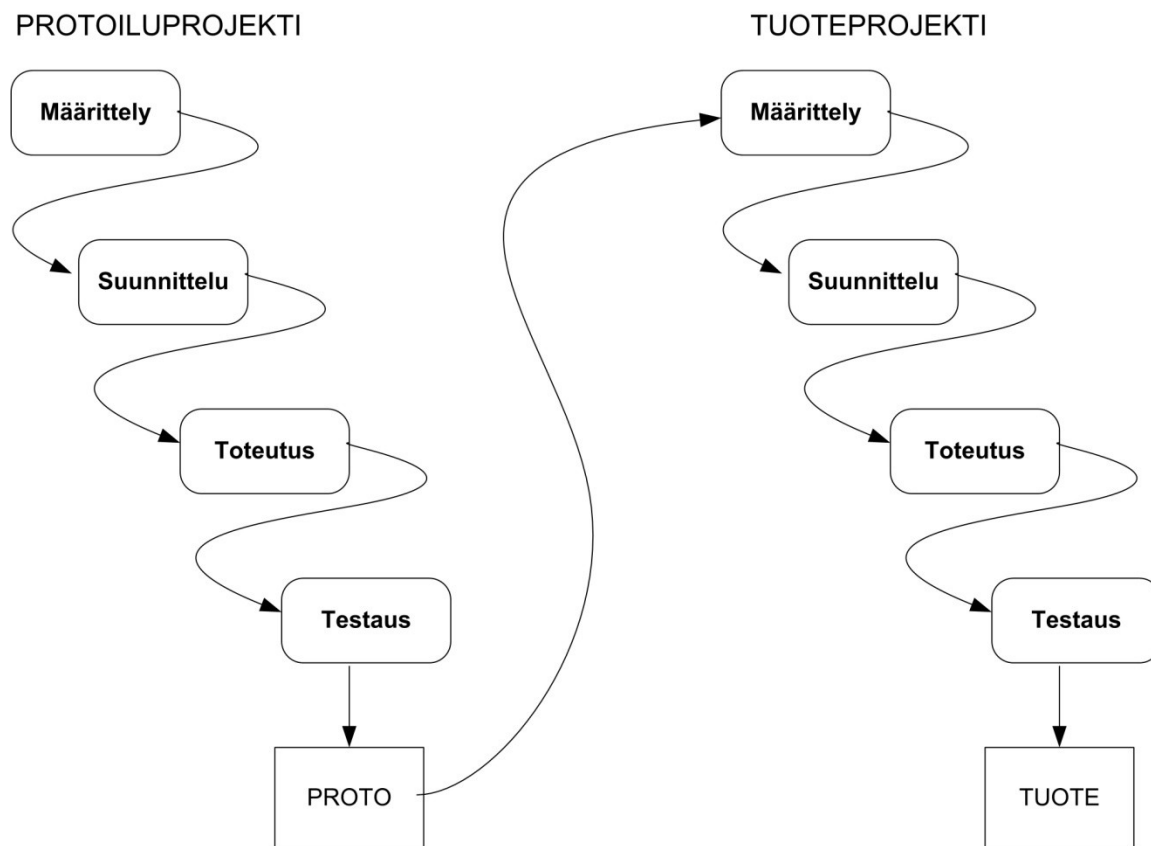


ja Scrum –menetelmät, joiden iteraatiot ovat lyhyitä ja näin kehittämisen katsotaan olevan jatkuvaa. (Haikala & Märijärvi 2004.)

### 4.3.1 Vesiputousmalli

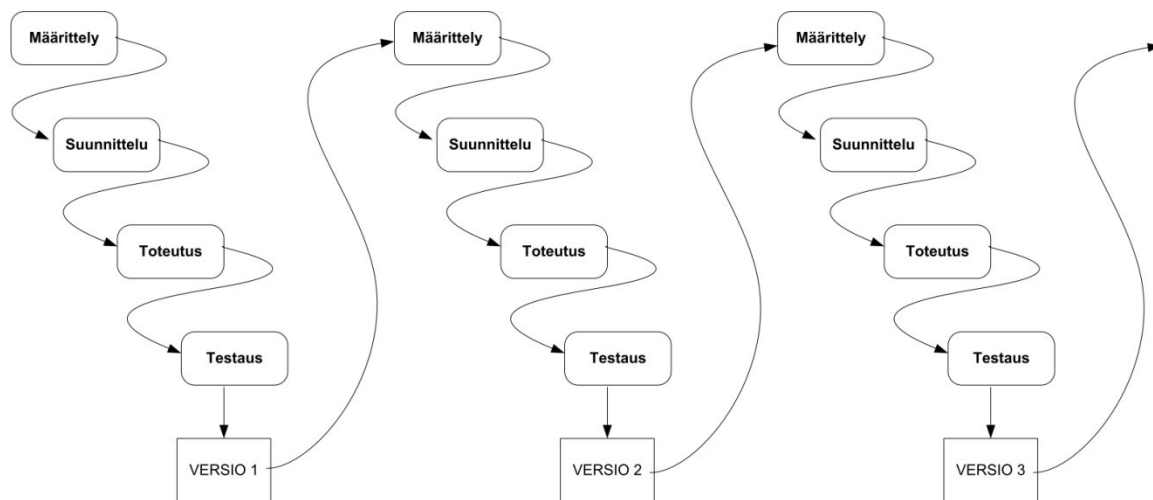
Vesiputousmalli (Kuva 1) on yleinen tapa kehittää ohjelmistoa. Malli soveltuu parhaiten projekteihin, joissa tavoitteet ovat selvillä ja kokonaisuus on monimutkainen (TAIK-koulutuskeskus 1998.). Käytännössä ohjelmistokehitys ei koskaan täysin voi edetä vesiputousmallin mukaisesti, koska usein osa vaatimuksista selviää hankkeen aikana ja näin ollen vaatimukset muuttuvat jatkuvasti. Mallia voidaan kuitenkin pitää toiminnallisena mallina, jonka mukaisesti pyritään toimimaan siinä määrin kuin on mahdollista (Haikala & Märijärvi 2004.)

### 4.3.2 Prototyypimalli



Kuva 3. Esimerkki protoilumallista (Haikala & Märijärvi 2004).

Kuvan 3 prototyypimalli soveltuu parhaiten uuden teknisen ratkaisun vaatiman kokeilun tekemiseen tai etsimään epäselviä asiakasvaatimuksia. Prototyypin käyttövaihtoehtoja ovat: 1. Prototyypin valmistuttua, sen avulla määritellään toteutettava järjestelmä, joka sitten toteutetaan uudelleen tai 2. Prototyyppi kehitetään valmiiksi järjestelmäksi. Prototyypimallin on huomattu olevan tehokas, kun määritellään käyttöliittymää ohjelmistolle. Tällä tavalla voidaan visuaalisesti osoittaa asiakkaalle, miltä ohjelmisto näyttää käyttäjän näkökulmasta. Vaarana on, että prototyypistä tehdään liian valmiin näköinen ja asiakkaan mielestä ohjelma näyttäisi olevan käytännössä valmis, vaikka näin ei ole. Lisäksi vaarana on joutua silmukkaan, jossa prototyyppiä kehitetään ja kehitetään vailla määränpäättä, mutta tämä ongelma on ratkaistavissa sopimusteknisesti. (Haikala & Märijärvi 2004.)



**Kuva 4. Tuotekehityssykli (Haikala & Märijärvi 2004).**

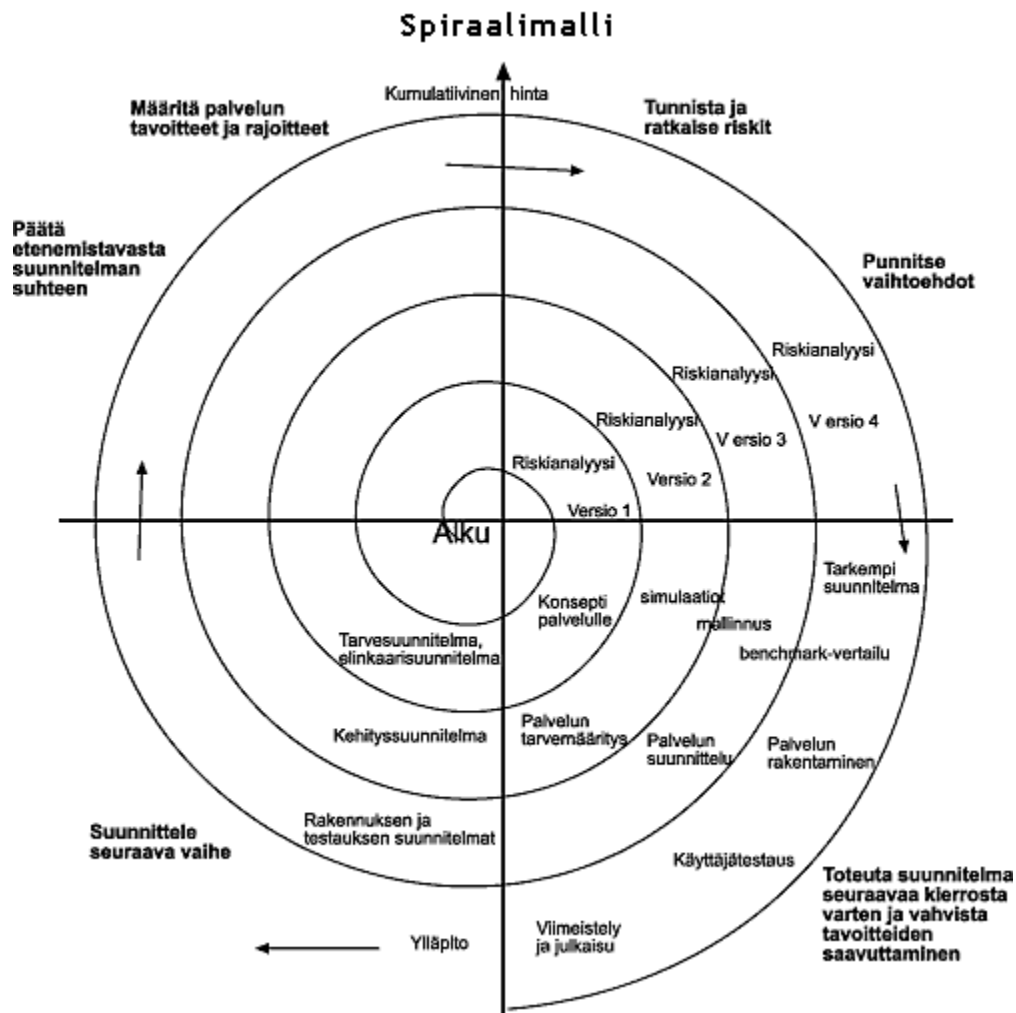
Tuotekehityshankkeet etenevät usein sykleissä eli järjestelmästä julkaistaan uusi versio tietyn ajanjakson päästä, esimerkiksi vuoden välein, joka sisältää uusia ominaisuuksia, korjauksia ja parannuksia. Uusi versio syntyy asiakkailta saatujen parannusehdotusten pohjalta. Kuvan 4 mallia käytettäessä tuotekehitys etenee liian pitkäkestoisesti ja loppukäyttäjältä saadaan palautetta vasta kehityssyklin myöhäisessä vaiheessa. Näin ollen tehtyjen määrittely- ja suunnitteluvirheiden korjaaminen tulee kalliiksi ja virheistä päästään oppimaan vasta seuraavassa syklissä. (Haikala & Märijärvi 2004.)

### 4.3.3 Inkrementaalinen malli

Inkrementaalinen eli EVO –mallin ideana on tehdä ensimmäiseksi ydinjärjestelmä, jota sitten kehitetään eteenpäin uusilla ominaisuuksilla. Kehitysprosessi jatkuu kuvan 4 tavoin, mutta yksittäiset iteraatiot ovat lyhyitä, viikosta muutamaan kuukauteen. Iteraation tuloksena syntyy uusia ominaisuuksia sisältävä järjestelmä. Näin ollen ohjelmistoa voidaan testata projektin alkuvaiheessa ja edellä mainittujen pitkäkestoisten mallien ongelmat on helpommin ratkaistavissa.

Evo –mallia kutsutaan inkrementaaliseksi malliksi, koska sillä tarkoitetaan ohjelmistokehitystä, jossa lopputuotetta kehitetään pieninä inkrementteinä projektin sisällä. (Haikala & Märijärvi 2004.)

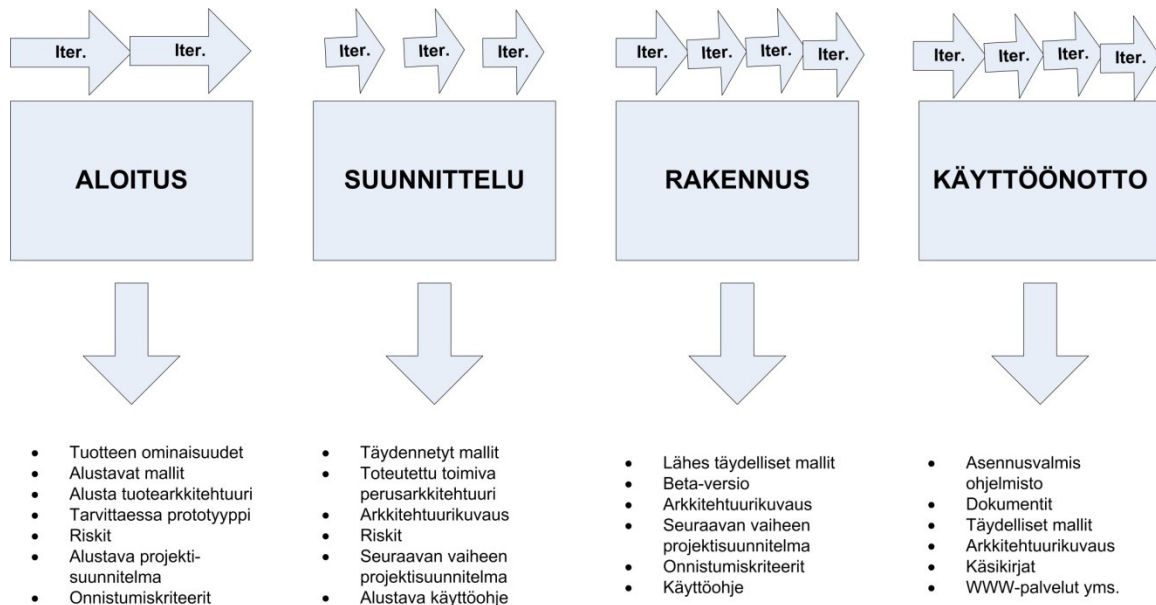
### 4.3.4 Spiraalimalli



Kuva 5. Esimerkki spiraalimallista (TAIK koulutuskeskus 1998).

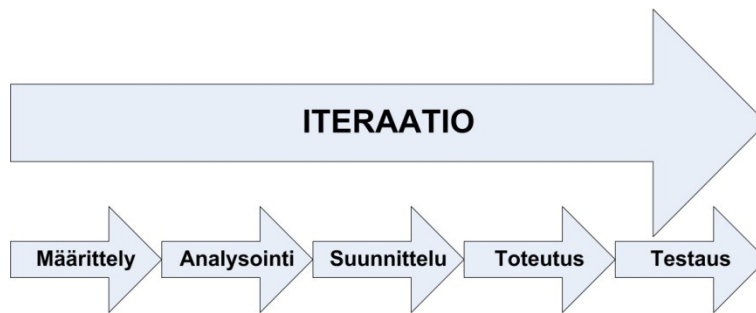
Spiraalimalli muistuttaa EVO –mallia. Malli koostuu useista samanlaisista kierroksista, joissa samat vaiheet toistetaan aina uudelleen. Malli on yhdistelmä vesiputous- ja prototyypimallia ja sisältää riskienhallinnan omana vaiheena. (Haikala & Märijärvi 2004.)

### 4.3.5 RUP –malli



Kuva 6. Esimerkki RUP -mallin päävaiheista (Haikala & Märijärvi 2004, 46).

RUP –malli (engl. *Rational Unified Process*) perustuu peräkkäisiin iteraatioihin. Jokainen iteraatio muodostaa oman vesiputouksensa, joka sisältää aktiviteetit: määrittely, analysointi, suunnittelu, toteutus ja testaus. Kuvassa 6 RUP –mallin kehitys jakaantuu neljään eri päävaiheeseen: aloitus (engl. *Inception*), suunnittelu (engl. *Elaboration*), rakennus (engl. *Construction*) ja käyttöönotto (engl. *Transition*). Jokainen päävaihe koostuu useammasta pienemmästä iteraatiosta, joka on esitetty kuvassa 7. Aloitus -vaiheen aktiviteetit painottuvat asiakasvaatimusten analysointiin eli kyseessä on esitutkimusvaihe. Suunnitteluvaiheessa toteutetaan arkkitehtuuri. Rakennusvaiheessa ohjelmistoa kehitetään lyhyin iteraatioin, joista syntyy uusi versio ohjelmasta niin sanottu beta –versio, joka jaetaan testikäyttäjille. Käyttöönottovaiheen tehtävänä on tuottaa käyttöohjeet ja lisenssisopimukset sekä miettiä ja julkaista tuotteelle ylläpitopalvelu. (Haikala & Märijärvi 2004.)



**Kuva 7. Esimerkki iteraatio –kehityksestä (Haikala & Märijärvi 2004).**

Tämäntyyppisten EVO –malleista johdettujen mallien ongelmaksi muodostuu se, että seuraavan inkrementin aloitus myöhästyy, koska projektiryhmän aika menee asiakkaan ongelmien selvittelyyn ja virheiden korjailuun. Toinen ongelma tulee vastaan, kun käytetään liian lyhyitä inkrementtejä, josta aiheutuu ohjelmiston sirpaloituminen ja kokonaisarkkitehtuurin rapautuminen. (Haikala & Märijärvi 2004.)

## **5 KETTERÄT MENETELMÄT**

### **5.1 Yhteiset piirteet**

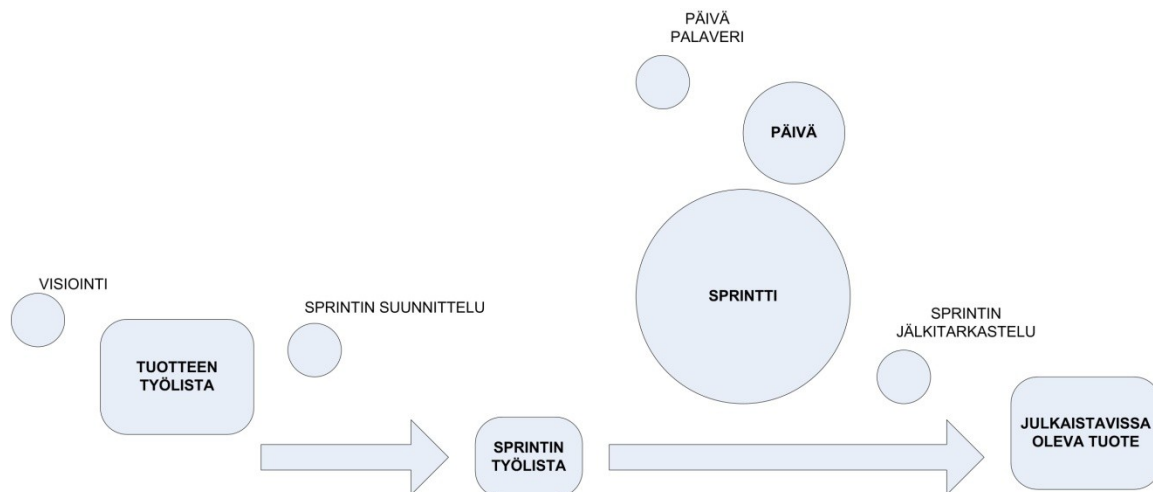
Agile eli ketterissä –menetelmissä iteraatiot ovat hyvin lyhyitä. Tämän vuoksi katsotaankin, että ketterillä –menetelmillä kehittäminen on ikään kuin jatkuvaa. Jatkuva kehittäminen mahdollistaa testauksen suorittamisen automatisoidusti. Kaikki ketterät –menetelmät rakentuvat erimittaisten syklien päälle. Tärkeimmät ovat sprintti ja päivä. Sprintillä tarkoitetaan yhtä kehitysjaksoa, jonka jälkeen ohjelmisto on periaatteessa julkaisuvalmis. (Haikala & Märijärvi 2004.) Päivällä tarkoitetaan päivittäistä palaveria, jossa käydään läpi asioita kunkin osallistuvan henkilön osalta: mitä teit eilen? mitä teet tänään? onko tekemiselle esteitä? (Mountain Goat Software 2010.)

Ketterät –menetelmät etenevät yksinkertaistettuna seuraavasti. Työ aloitetaan tekemällä ominaisuudelle testitapauksia. Testitapaukset ajetaan ja todetaan, että testit epäonnistuvat. Tämän jälkeen tehdään muutoksia niin kauan kunnes kaikki testitapaukset saadaan ajettua virheettömästi läpi. (Haikala & Märijärvi 2004.)

### **5.2 Ketteriä menetelmiä**

Seuraavaksi esitellään lyhyesti kaksi hyvin tunnettua ketterän –kehittämisen menetelmää: Scrum ja Agile Modeling, joita käytetään ohjelmistokehityksessä. Tarkoituksena on havainnollistaa erilaisia ketterän –kehittämisen arvoja ja periaatteita.

## 5.2.1 Scrum



Kuva 8. Scrum -kaavio (Sininen Meteori 2010 b).

Scrum tarjoaa ohjelmistokehityksen, joka keskittyy projektin ohjaukseen. Scrum keskittyy projektin vaiheistamiseen ja projektin kontrollointiin. Perinteisessä ohjelmistokehityksessä, joka noudattaa vesiputousmallin mukaista prosessimallia, on yleensä määrittelijä, suunnittelija, ohjelmoija, testaaja ja projektipäällikkö. Jälkimmäistä roolia eli projektipäällikköä lukuun ottamatta, rooleissa voi olla useampiakin henkilöitä. Vesiputousmallista poiketen, Scrum –projektissa esiintyy vain kolme roolia: tuotteen omistaja, Scrum –mestari ja tiimi. Omistajan tehtävänä on vastata tuotteen ominaisuuksista ja toiminnallisuudesta. Mestarin tehtävänä on huolehtia, että tiimi tekee työtä ja noudattaa projektisuunnitelmaa. Lisäksi mestarin tehtävänä on ratkoa ongelmia sekä johtaa päivittäisiä palavereita. Tiimiin kuuluvien tehtävänä on raportoida mestarille ongelmista ja työn etenemisestä. Tiimissä ei ole ketään nimetty erityisrooliin, esimerkiksi suunnittelija tai ohjelmoija vaan tiimi koostuu ammattilaisista joiden taidoilla tuotetta rakennetaan. (Sininen Meteori 2010 b) Kuvassa 8 on esitetty Scrum –prosessin vaiheet:

1. Suunnitteluvaihe, jossa määritellään tuotteen seuraavaan julkaisuun tulevat ominaisuudet. Ominaisuudet on listattuna työlistalle asiakkaiden tai tiimin toiveista. Näiden ominaisuuksien perusteella arvioidaan projektin aikataulu ja kustannukset.
2. Arkkitehtuurisuunnitteluvaihe, jossa luodaan vaatimusten täyttävä arkkitehtuuri.

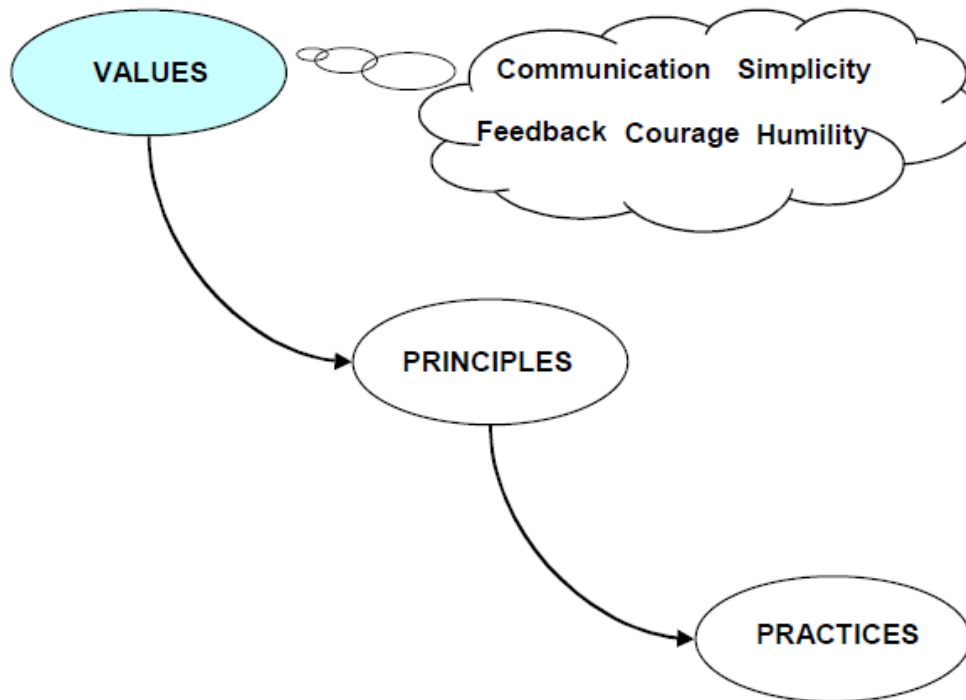


3. Toteutusvaihe, jossa toteutetaan kaikki sprinttiin työlistalta valitut ominaisuudet.
4. Päätösvaiheessa, jossa luodaan julkaisun dokumentaatiot. (Jyväskylän yliopisto 2010.)

### 5.2.2 Agile Modeling

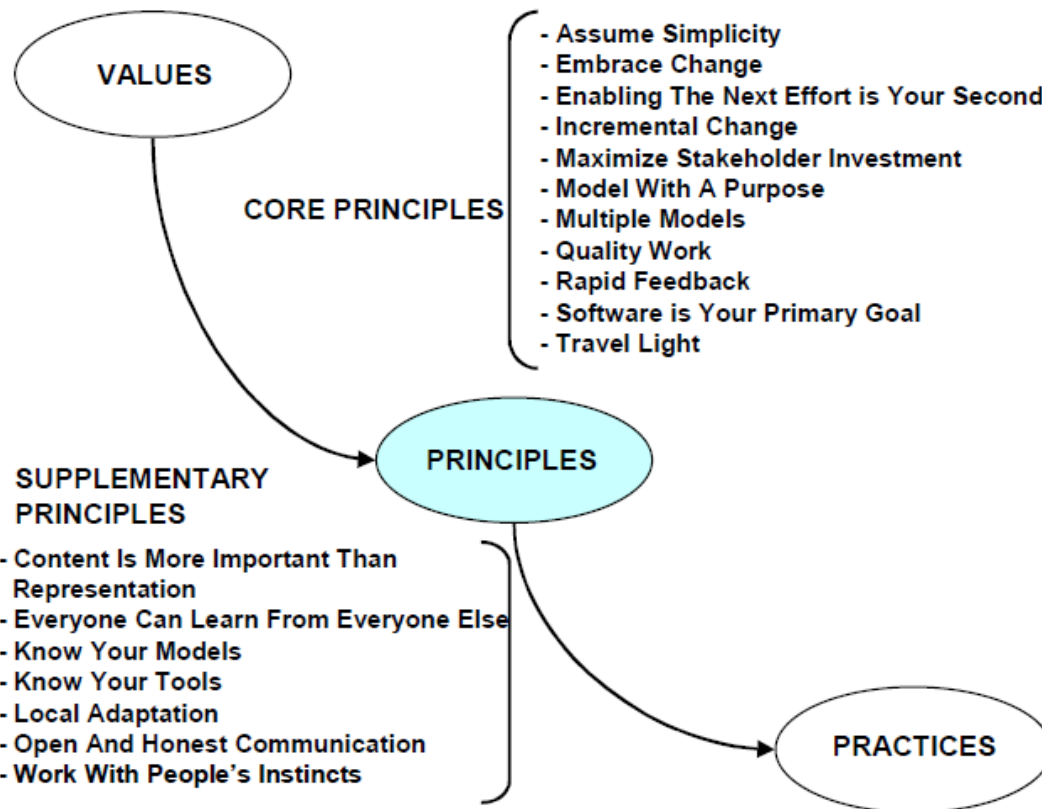
*Agile Modeling* tunnetaan yleisesti termillä *Agile*, jolla tarkoitetaan yleistetyksi kaikkia ketteriä –menetelmiä. Suomalaisia artikkeleita lukiessa voi huomata, että nämä termit ovat usein sekoittuneet. Kyseessä on siis kaksi aivan eri käsitettä. *Agile Modeling* eli ketterä mallinnus itsessään keskittyy suunnitteluun mallintamisen näkökulmasta. Menetelmä vastaa kysymykseen: Miten ketterää kehitystä voidaan mallintaa tehokkaasti ajautumatta suuriin suunnitteludokumentaatioihin, niin että koko projektin tavoite unohtuu. (Sininen Meteori 2010 a)

Ketterää mallintamista sovelletaan ohjelmistokehityksessä vaatimusmäärittelyn laatimisessa, analysointivaiheessa, suunnitteluvaiheessa ja arkkitehtuurinmäärittelyn laatimisessa. Mallinnuksen tavoitteena on saada aikaan tarpeeksi hyviä malleja, joiden avulla voidaan mallintaa tehokkaammin. Mallintamisesta saattaa syntyä vähemmän tuloksia, mutta projektissa tiedostetaan paremmin mitä ja miksi mallinnetaan. Usein luullaan, että ketterä mallintaminen on ohjelmistokehityksessä käytettävä prosessimalli, mutta näin asia ei ole. Ketterä mallintaminen täydentää omalta osaltaan, esimerkiksi XP (engl. *Extreme Programming*) tai RUP –prosessimallia.



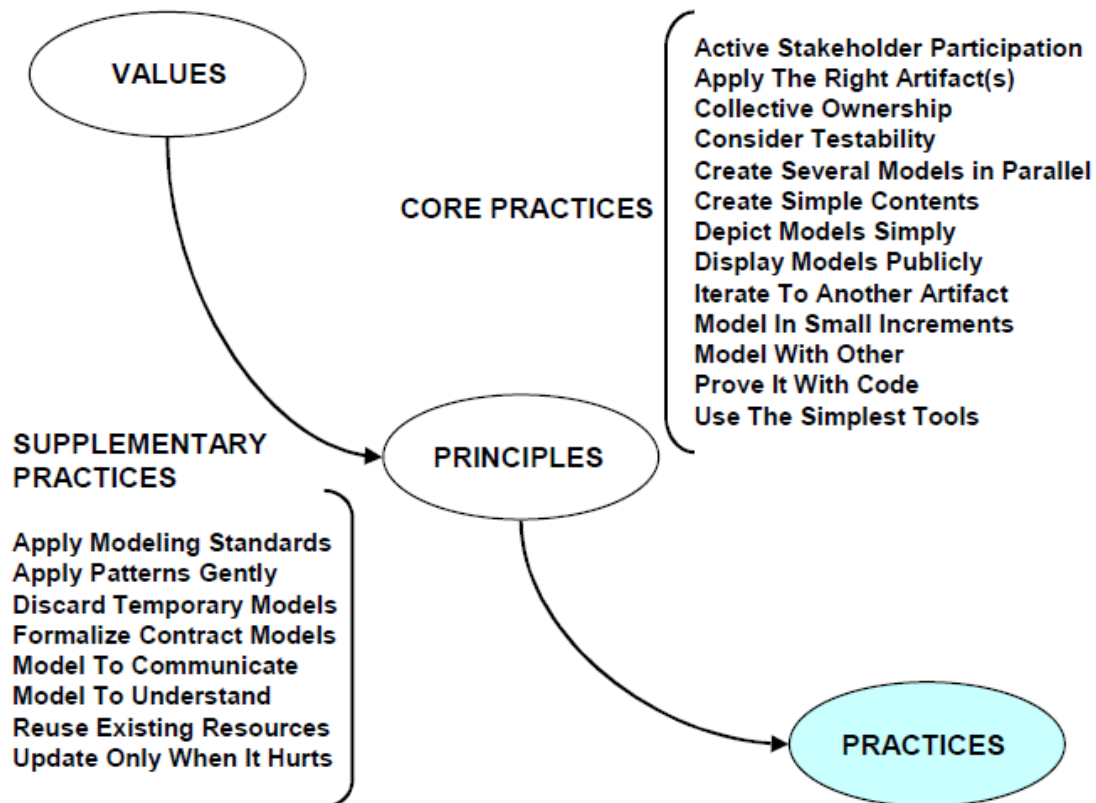
**Kuva 9. Ketterän mallintamisen arvot (Kajava & Nykopp 2001, 2).**

Ketterän mallintamisen arvoja (Kuva 9) ovat kommunikointi, helppotajuisuus, palaute, rohkeus ja nöyryys. Neljä ensimmäistä arvoa pohjautuu XP:ssä käytettyihin arvoihin, joista kommunikointia painotetaan eniten. Ketterä mallintaminen sopiikin parhaiten kun projektin henkilöt toimivat samassa toimipisteessä tai jopa samassa työtilassa. (Kajava & Nykopp 2001, 2-4.)



Kuva 10. Ketterän mallintamisen periaatteet (Kajava & Nykopp 2001, 3).

Ketterän mallintamisen periaatteet (Kuva 10), ovat peräisin myös XP:ssä toteutetuista periaatteista. Nämä periaatteet jaetaan ydinperiaatteisiin ja täydentäviin periaatteisiin. Periaatteilla pyritään pitämään dokumentointi riittävän kevyenä ja tuottamaan dokumentit riittävällä tarkkuudella. Projektin kuluessa dokumentteja ja malleja on ylläpidettävä jatkuvasti. Projektikohtaisesti on harkittava mitä dokumentteja ja malleja tarvitaan, sekä millä tarkkuudella ne tuotetaan, jotta dokumentointi olisi kustannustehokkainta. (Kajava & Nykopp 2001, 2-4.)



Kuva 11. Ketterän mallintamisen käytännöt (Kajava & Nykopp 2001, 3).

Ketterässä mallintamisessa käytännöt (Kuva 11) ohjaavat toimintaa. Käytännöt ovat peräisin myös XP:n käytännöistä ja ne jaetaan periaatteiden tavoin ydinkäytäntöihin ja täydentäviin käytäntöihin. Tärkeimmäksi katsotaan, että asianomistajan on osallistuttava projektiin päivittäin. Asianomistajalla tarkoitetaan asiakasta ja tilaajaa, mutta myös käyttäjiä, käyttäjien johtoa tai jopa help-deskin edustajia. Integrointi on jatkuvaa, jolla pyritään vastaamaan vaatimusten muuttumisesta aiheutuneisiin riskeihin. Mallinnuksen onnistumista voidaan varmentaa prototyypein, jolloin on mahdollista tarkastella konkreettisesti sovelluksen suorituskykyä ja käytettävyyttä. Ketterässä mallinnuksessa korostetaan mallin testattavuutta, sekä mietitään jo mallinnusvaiheessa miten kyseinen osuus voidaan testata.

Ketterä mallintaminen edellyttää hyvin toimivaa kommunikointia. Projektiin osallistuvien henkilöiden on ymmärrettävä toisiaan. Ketterä mallintaminen sopii projekteihin, joissa käytetään iteratiivista ja inkrementaalista prosessia, koska ketterän mallintamisen

tarkoituksena on täydentää muita keveitä menetelmiä, kuten XP:tä tai RUP –prosessia. Ketterää mallintamista voidaan myös hyödyntää perinteisessä ohjelmistokehityksessä, mutta tällöin menetelmästä ei saada kaikkea irti. Ketterää mallintamista ei kannata soveltaa, jos projektin asianomistaja ei voi aktiivisesti osallistua projektiin. (Kajava & Nykopp 2001, 2-4.)

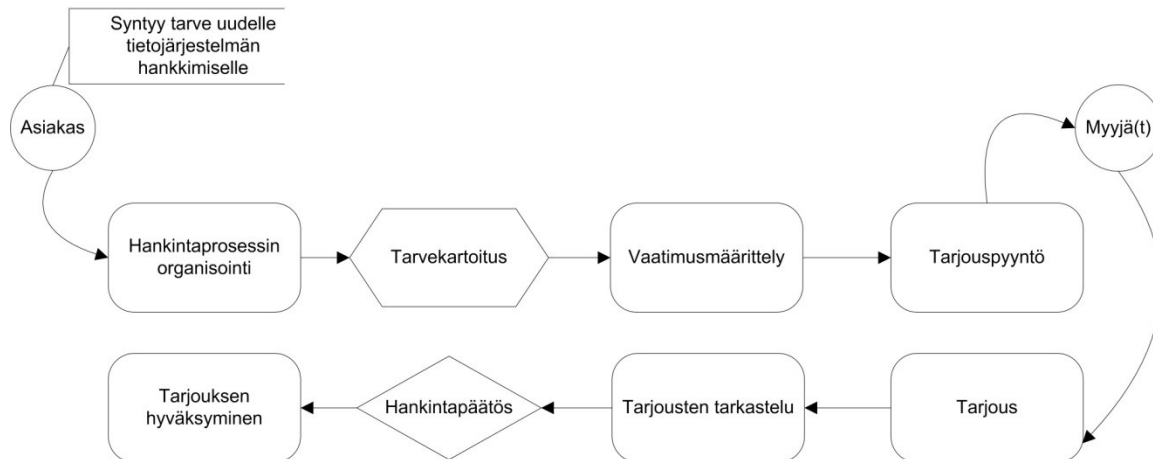
## 6 HANKINTAPROSESSI

Tässä ja seuraavassa 7. luvussa käsitellään opinnäytetyönä toteutettua opasta Lemonsoft toiminnanohjausjärjestelmän käyttöönotosta. Opas syntyi kirjallisen tiedon ja Lemonsoftin koulutusmateriaalien ja haastattelujen pohjalta. Tarkoituksena on antaa kokonaiskuva Lemonsoft -toiminnanohjausjärjestelmän käyttöönottoon liittyvistä seikoista. Ensimmäiseksi tässä luvussa esitellään lyhyesti toiminnanohjausjärjestelmän hankintaprosessia, jonka tarkoituksena on osoittaa käyttöönottovaiheen merkitys koko hankintaprosessin aikana. Lopuksi luvussa esitellään varsinaista Lemonsoft –ohjelmiston käyttöönottovaihetta, jonka toteutin opinnäytetyönä toimeksianto yritykselle. Käyttöönottovaihe sisältää ohjeita mm. käyttöönottosuunnitelman laatimiseen, tiedonsiirtojen suunnitteluun ja toteuttamiseen, asennusvaiheen suunnitteluun ja toteuttamiseen. Lisäksi luvussa käsitellään mahdollisten muutosten toteuttamista, ja kouluttamista sekä lopuksi esitellään mitä tapahtuu varsinaisena ohjelmiston käyttöönottopäivänä.

Toiminnanohjausjärjestelmän hankintaprosessi voidaan jaotella kolmeen eri päävaiheeseen: valintaprosessi, käyttöönottoprosessi ja lopullinen käyttö. (Vilpola & Kouri 2006, 75.) Nämä päävaiheet voidaan pilkkoa useisiin pienempiin kokonaisuuksiin. Tämä opas toiminnanohjausjärjestelmän käyttöönottosuunnitelmasta keskittyy laajennetusti Lemonsoft -toiminnanohjausjärjestelmän käyttöönottoprosessiin ja noudattaa <http://www.lemonsoft.fi/nethelp/> -sivustolta löytyvää käyttöönottoprosessin rakennetta.

## 6.1 Valintaprosessi

### TOIMINNAHOJAUSJÄRJESTELMÄN VALINTAPROSESSI



Kuva 12. Tyypillinen valintaprosessi (JHS-Suosituksset 2008).

Toiminnanohjausjärjestelmän valintaprosessia voidaan kuvata monella eri tapaa riippuen käytettävästä hankintamenettelystä. Hankintamenettelyjä ovat avoin menettely, rajoitettu menettely, neuvottelumenettely ja kilpailullinen neuvottelumenettely sekä suoramarkintamenettely. (Hankintatieto 2010.) Kuvan 12. prosessikaaviota voidaan käyttää avoimeen-, rajoitettuun-, ja suljettuun hankintamenettelyyn, kun valintaprosessi on saatettu onnistuneesti päätökseen. Käyttöönottohanketta jatketaan siirtymällä seuraavaan vaiheeseen eli itse käyttöönottoprosessiin, josta tämä opas on varsinaisesti kirjoitettu.

### 6.1.1 Hankintamenettelyt

Hankkiessa toiminnanohjausjärjestelmää, hankinnasta on järkevintä pyytää tarjoukset järjestelmän toimittajilta eli toteuttaa kilpailuttaminen, jotta saadaan selvitettyä edullisin vaihtoehto. Kilpailutuksen tekeminen ei ole pakollista, mutta kuitenkin suositeltavaa, oli sitten kyseessä toimittaja tai tuote. Tarjouskilpailun tekemisellä saadaan paras kuva markkinoilla olevista vaihtoehdoista. (TTL Julkaisusarja 2002, 29.)

Hankintamenettelyt voidaan jakaa kahteen kategoriaan, joita ovat joko suoraan toimittajien kanssa asiointi tai julkiset hankinnat, jota säätelee hankintalaki. (TTL Julkaisusarja 2002.) Hankintalaissa on säädetty 1.6.2010 lähtien seuraavasti: *”kesäkuusta lähtien kansallinen kynnysarvo tavara- ja palveluhankinnoille, käyttöoikeussopimuksille ja suunnittelukilpailuille on 30 000 euroa. Terveystenhoito ja sosiaalipalveluiden sekä koulutuspalveluiden yhteishankinnoille kynnysarvo on 100 000 euroa ja rakennus- ja käyttöoikeusurakoille 150 000 euroa.”* (Yrittäjät.fi 2010.)

### **6.1.2 Toimittajien kanssa asiointi**

Hankinnasta pyydetään tarjous suoraan toimittajilta. Hankinnasta laaditaan hankintasuunnitelma, johon kuvataan käytettävät menettely, joka määrää toimittajien kanssa asioimisen ja tiedottamisen prosessin, sekä yritysten yhteyshenkilöt ja menettelyt. Toimittajien tekemät tarkentavat kysymykset ja asiakkaan antamat vastaukset on ehdottomasti toimitettava tiedoksi kaikille tarjoustu valmisteleville toimittajille. Hankintatilanteessa vuorovaikutuksen on oltava avointa. Osapuolten näkemyksiä on kuunneltava ja asiakkaan on muistettava, että päätöksiä ei pidä lyödä lukkoon jo valmisteluvaiheessa. Ideaalista on, että valintaprosessi voidaan saattaa läpi joustavasti, jotta uusia ideoita voidaan hyödyntää valmisteluvaiheessa. (TTL Julkaisusarja 2002.)

### **6.1.3 Julkiset hankinnat**

Julkisia hankintoja säätelee Suomessa ja Euroopassa laki nimeltä hankintalaki. Laki määrää tekemään julkisen kilpailutuksen kaikista julkisista hankinnoista, jotka ylittävät tietyt kynnysarvot.

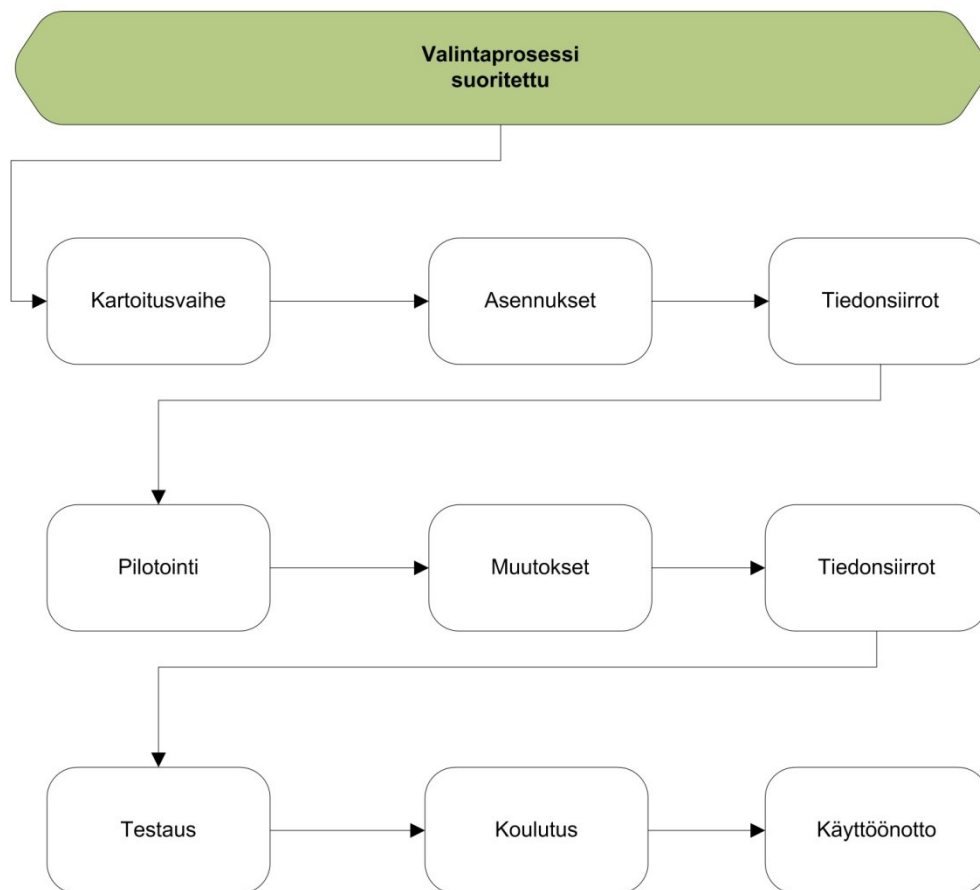
Julkisen hankinnan piirissä käytettäviä vaihtoehtoja ovat: avoin, rajoitettu ja neuvottelumenettely. Näistä vaihtoehtoista suositellaan käytettävän joko avointa tai rajoitettua menettelyä. Neuvottelumenettelyä käytetään usein palveluhankinnoissa, joissa kokonaishinnoittelu tai sopimusehtojen erittely valmisteluvaiheessa on erityisen vaikeaa. Neuvottelumenettelyä käytettäessä julkaistaan hankinnasta ensin hankintailmoitus, jonka jälkeen tarjoajilla on hankintalaissa määrätty aika jättää osallistumishakemus hankintaorganisaatiolle. Hakemuksen jättäneistä hankintaorganisaatio valitsee



neuvottelukumppanit, joille tarjouspyyntö esitetään. Hankintapäätös syntyy neuvottelujen jälkeen. (TTL Julkaisusarja 2002, 30.)

## 6.2 Käyttöönottoprosessi

# TOIMINNANOHJAUSJÄRJESTELMÄN KÄYTTÖÖNOTTOPROSESSI



Kuva 13. Tyypillinen käyttöönottoprosessi (Lemonsoft Oy 2010 a).

Käyttöönottoprosessin tarkoituksena on varmistaa se, että käyttöönotto sujuu ongelmitta ja hallitusti, sekä aikataulun mukaisesti. Kuvan 13 prosessikaavio kuvaa käyttöönottoprosessin korkean tason aktiviteetteja. Käyttöönottovaiheesta voi lukea lisää luvusta 7.

### **6.3 Varsinainen käyttö**

Kun käyttöönottoprosessi keskittyy järjestelmän käyttöönoton organisointiin, keskittyy varsinainen käyttö järjestelmän käyttöön, sekä ylläpito- ja kehittämistehtäviin. Ylläpidon tehtävänä on taata, että järjestelmä toimii oikein. Lisäksi tehtävänä on selvittää ja korjata mahdollisia vikoja sekä järjestää uusien ja vanhojen käyttäjien kouluttaminen. Kehittämistyön tehtävänä on taata ja tarkastella, että järjestelmä vastaa yrityksen tarpeita. (Grotenfelt ym. 1989.)

#### **6.3.1 Käyttöönottamattomat moduulit**

Käyttöönotetun toiminnanohjausjärjestelmän moduulivaihtoehdot saattavat jäädä riittämättömäksi yrityksen kasvaessa ja laajentuessa. Tähän on ratkaisuna se, että Lemonsoft -ohjelmistoa voidaan helposti päivittää ottamalla käyttöön uusia moduuleja ja toimintoja. (Lemonsoft Oy 2010 d.)

#### **6.3.2 Ohjelmistopäivitykset**

Ohjelmistopäivityksiä tehtäessä on mahdollista hankkia kunkin ohjelmistopäivityksen kohdalla koulutuspaketti koskien päivityksessä tulleita uusia ominaisuuksia. Vaihtoehtoisesti asiakas voi opiskella uudet ominaisuudet Lemonsoftin ohjeistuksen ja dokumentaation avulla. (Joki-Hollanti 2010.)

#### **6.3.3 Räätelöinnit**

Mahdollisuuksien mukaan voidaan järjestelmään toteuttaa yrityskohtaisia räätelöintejä esimerkiksi varaston tiedonkeruulaitteiden integraatioita yms. Räätelöinnit toteuttaa Lemonsoftin ulkoinen yhteistyökumppani. (Joki-Hollanti 2010.)

## **7 KÄYTTÖÖNOTTOVAIHE**

Käyttöönottovaihe käsittelee varsinaista Lemonsoft -toiminnanohjausjärjestelmän käyttöönottovaihetta. Tämä luku sisältää ohjeita mm. käyttöönottosuunnitelman laatimiseen, tiedonsiirtojen suunnitteluun ja toteuttamiseen sekä asennusvaiheen suunnitteluun ja toteuttamiseen. Lisäksi luvussa käsitellään mahdollisten muutosten tekemisistä järjestelmään, käyttäjien kouluttamista ja lopuksi esitellään mitä tapahtuu varsinaisena käyttöönottopäivänä.

### **7.1 Käyttöönottosuunnitelma**

Käyttöönottosuunnitelma pitää sisällään kaikki hankkeen käytössä olevat henkilöstöresurssit, tarkan aikataulun, hankkeen vaiheistuksen, käytössä olevan budjetin, tiedotussuunnitelman, riskien hallinnan ja koulutussuunnitelman. (Lemonsoft Oy 2010 d.)

#### **7.1.1 Henkilöstöresurssit ja tehtävät**

Hankkeeseen valitaan vastuhenkilö, joka tulee toimimaan projektipäällikkönä läpi hankkeen. Projektipäälliköllä täytyy olla aikaa johtaa hanketta ja hänen on voitava delegoida työtehtäviään myös muille työntekijöille. Hankkeeseen osallistuvien henkilöiden tulee olla hyvin sitoutuneita saattamaan hanke loppuun. (Lemonsoft Oy 2010 d; Pelin, 2009.)

#### **7.1.2 Aikataulu**

Käyttöönotto voi tapahtua milloin tahansa, mutta kiireaikoja tulee ja niitä kannattaa välttää. Käyttöönoton ajankohtaan kannattaa varata pelivaraa, jotta käyttöönottoa saadaan myöhennettyä, mikäli jotain yllättävää tulee tapahtumaan. Joustavalla aikataululla tavoitellaan onnistumista. (Lemonsoft Oy 2010 d; Murch, 2002.)

#### **7.1.3 Hankkeen vaiheistus ja päätöksentekopisteet**

Käyttöönottohanke kannattaa jakaa pienempiin peräkkäisiin aktiviteetteihin. Aktiviteetit tulee sijoittaa kalenteriin, sekä asettaa niiden välille päätöksentekopisteitä, joissa todetaan

edellytykset siirtyä seuraavaan vaiheeseen. Vaiheistamisella ja aktiviteettien valvonnalla on mahdollista saada parempi kustannusohjaus. (Lemonsoft Oy 2010 a.)

Päätöksentekopisteet ovat maaleja, joita tavoitellaan kunkin aktiviteetin kohdalla, jotta olisi mahdollista siirtyä suorittamaan seuraavaa vaihetta. (Pelin, 2009.)

#### **7.1.4 Budjetti**

Hankintaprojektin alussa on määritelty toiminnanohjausjärjestelmän käyttöönottohankkeelle budjetti, jonka rajoissa käyttöönotto on tarkoitus toteuttaa. Kustannusvalvonnan tehtävänä on pitää nämä kustannukset kurissa ja varmistaa, että käyttöönotto pysyy taloudellisesti kannattavana. (Ruuska, 2007.) Käyttöönottohankkeen budjettiin on arvioitu kustannukset tuntimääräisesti. Poikkeukset tulee päivittää budjettiin yrityksen omia toimintatapoja noudattaen. (Pelin, 2009.)

Käyttöönottohankkeen yleisiä kuluja ja budjetoitavia kohteita ovat:

- koulutukset
- ohjelmistoon tehtävät muutokset
- integrointi ja testaus
- tiedon analysoiminen ja muuntaminen
- konsultointi
- henkilöstön sitouttaminen
- ylläpito. (The Financial Express 2008.)

#### **7.1.4.1 Kustannusarviointi**

Kustannusarviointi on osa projektin kannattavuuslaskelmaa, joka on apuna päätöksen teossa, kun hanketta ollaan käynnistämässä. Kustannusarviointi toimii kustannusvalvonnan vertailukohteenä. Kustannuksia arvioitaessa on muistettava, että arviointi tehdään riittävällä tarkkuudella ja niiden on sovelluttava kustannusvalvonnan kohteiksi.

Kustannusarviointi tarkentuu vaiheittain:

1. Alustava kustannusarviointi, jonka tarkkuus on suuntaa antava.
2. Peruskustannusarviointi, joka tarkentuu, kun selvillä ovat toteutettavat aktiviteetit, laiteluettelot ja tarjoukset.
3. Lopullinen kustannusarviointi tehdään, kun suunnitelmat ovat valmiina, jolloin suurin osa hankinnoista ja sopimuksista on tehty. (Pelin, 2009, 174-179.)

#### **7.1.4.2 Projektibudjetti**

Projektibudjetti on aikaan sidottu projektin taloudellinen toimintasuunnitelma, jota laadittaessa on tiedettävä missä järjestyksessä projektin aktiviteetit aiotaan suorittaa. Tämä edellyttää, että projektiaikataulu on valmis. Projektibudjettia ei ole sidottu kalenteriaikaan, vaan projektin aikatauluun. Aikataulun venyessä on korjattava projektibudjetti vastaamaan nykytilannetta. Projektibudjettiin kirjataan projektin menot. Projektibudjetti laaditaan kalenterivuosi- ja vuosipohjalle, johon eritellään kustannukset kuukausittain. Budjettia voidaan hyödyntää yrityksen vuosibudjetin perustana.

Projektin työtehtävät saadaan budjetoitua aikataulun avulla oikeaan ajankohtaan. Vaikeammin kohdistettavissa ovat erilaiset hankinnat, jotka erilaisten toimitusaikojen ja maksusopimusten vuoksi saattavat vääristää ja hankaloittaa budjetin seurantaa. Tästä voi aiheutua kustannusten siirtymisiä, mutta näillä ei ole merkitystä, kunhan siirtymät muistetaan kustannusvalvonnassa. Kustannusten siirtymiset johtuvat esimerkiksi laskutussiirtymistä, eivätkä siis kustannusylityksestä tai –alituksesta.

Projektin budjettia laadittaessa on otettava huomioon sen soveltuminen kustannusvalvonnan perustaksi. Tämän vuoksi raportoinnissa on käytettävä samoja nimikkeitä, kuin projektibudjetissa on määritelty. (Pelin, 2009, 179-281.)

#### **7.1.4.3 Kustannusvalvonta**

Kustannusvalvonnan tarkoituksena on varmistaa projektin taloudellinen toteutus. Raportoinnin ja valvonnan tulee olla mahdollisimman säännöllistä ja ajantasaista tietoa sisältävää, sekä kattaa kaikki kustannukset ja ohjata projektia. Käyttöönottohankeessa kustannusvalvonta kytkeytyy koko hankintaorganisaation kustannusseurantaan, laskutoimituksiin ja kirjanpitoon. (Pelin, 2009, 182-187.)

#### **7.1.5 Tiedottaminen ja tiedotussuunnitelma**

Projektipäällikön keskeinen tehtävä hankeessa on toimia tiedottajana. Tehtävänä on välittää tietoa hankeessa muille osallistujille sekä hankkeen ulkopuolelle (Pelin, 2009, 294-295.) Projektipäällikkö laatii tiedottamisesta tiedotussuunnitelman, joka määrää hankeessa käytettävän raportointikäytännön, dokumentoinnin ja kirjallisentiedon välittämisen. Tiedotussuunnitelma on osa projektisuunnitelmaa, jonka laatiminen ajoittuu projektin suunnittelu- ja organisointivaiheeseen. (Ruuska, 2007, 211-213.)

Tiedottamisessa on etenkin harkittava seuraavia seikkoja:

- Mikä on viestin tavoite?
- Kuka on viestin vastaanottaja?
- Mikä on oikea viestintätapa?
- Milloin on paras ajankohta? (Pelin, 2009, 297.)

Tiedottamisen painopistealueet toiminnanohjausjärjestelmän käyttöönotossa ajoittuvat etenkin hankkeen alkuun. Tiedottaminen on hoidettava avoimesti ja henkilökunnalle on kerrottava, miksi toiminnanohjausjärjestelmä otetaan käyttöön, mitä etuja siitä on ja mitä muutoksia järjestelmä tuo yrityksen toimintatapoihin. Toteutusvaiheessa on tärkeää tiedottaa miten käyttöönottohanke etenee niin osastoittain kuin yrityksen sisäisessä viestinnässäkin. Käyttöönottohanke loputtua projektipäällikkö laatii loppuraportin

hankkeesta, jossa kerrotaan käyttöönottohankkeen onnistumisesta ja toiminnanohjausjärjestelmän nykytilasta, sekä mitä jatkossa on kehitettävä. Tämän loppuraportin pohjalta voidaan käynnistää uusia hankkeita, kuten esimerkiksi koulutus- ja kehittämishankkeita. (Lemonsoft Oy 2010 d; Joki-Hollanti, 2010.)

### **7.1.6 Riskien hallinta**

Käyttöönottohankkeen onnistumisen kannalta vakavia riskejä ovat:

1. Yrityksen henkilöstö tai käyttöönottohankkeen henkilöt eivät sitoudu syystä tai toisesta. Yleisin syy on aikaresurssien puute.
2. Vanhan järjestelmän informaatio on puutteellista tai sen oikeellisuuteen luetaan liikaa. Toisin sanoen informaation oikeellisuutta ei tarkasteta.
3. Avain henkilö joutuu jättämään hankkeen, ja tämä johtaa väistämättä hankkeessa ongelmiin, ellei olemassa ole varasuunnitelmaa. (Joki-Hollanti, 2010.)

### **7.1.7 Koulutussuunnitelma**

Koulutussuunnitelma on osa käyttöönottosuunnitelmaa, joka voidaan tehdä myös erikseen. Suunnitelmaan määritetään ne henkilöt, jotka tarvitsevat koulutusta uuden järjestelmän käytössä, sekä se mitä heille opetetaan ja miten heidät koulutetaan. (Murch, 2002, 119.) Koulutuksessa ei ole kyseessä vain työntekijöiden kouluttaminen, vaan *”miten käyttäjien ja lopputuotteen ylläpidosta sekä käyttäjätuesta vastaavan henkilöstön koulutus ja perehdyttäminen järjestetään”* (Ruuska, 2007, 183.). Koulutuksesta laaditaan koulutusaikataulu, joka sisältää koulutussuunnitelman kehittämisen ja koulutuksen toteuttamisen aika- ja kustannusarviot. Kustannusarvioihin sisällytetään koulutuksen suunnitteluun ja toteuttamiseen käytetty aika ja siitä syntyvien kustannusten lisäksi myös hankkeen aiheuttamat epäsuorat kustannukset, jotka syntyvät tuottavuuden laskusta, kun työntekijät ovat poissa työpaikaltaan saamassa koulutusta. (Murch, 2002, 119.) Kouluttamisesta voit lukea lisää kohdasta 7.7.

## **7.2 Projektioorganisaatio**

Hankintapäätöksen synnyttyä perustetaan yrityksen ja toimittajan välille hanketyöryhmä, jossa työskentelee hankkeeseen osallistuvat henkilöt.

### **7.2.1 Työryhmä ja hanke**

Hanketyöryhmän tehtävänä on johtaa käyttöönottohanke yrityksen päässä. Hanketyöryhmän tehtävänä on laatia käyttöönotosta suunnitelma, jota noudatetaan käyttöönoton aikana. (TTL Julkaisusarja 2002.)

### **7.2.2 Hanke**

Käyttöönotto on hanke, jota on suunniteltava. Hankkeesta on laadittava tarkka käyttöönottosuunnitelma. Lisäksi käyttöönottohankeessa noudatetaan hyvää projektikulttuuria:

- projekti vaiheistetaan ja päätöksentekopisteet tunnistetaan
- projektipäälliköllä on riittävästi valtaa ja hänen panosta arvostetaan
- projektia johdetaan yhteisesti hyväksytyjen sääntöjen mukaisesti. (Lemonsoft Oy 2010 d.)

#### **7.2.2.1 Hankkeen omistaja**

Hankkeen omistaa yleensä yrityksen toimitusjohtaja tai talouspäällikkö. Omistaja toimii hankkeen ohjausryhmässä puheenjohtajana. (Pelin, 2009.)

Hankkeen omistajan tehtävänä on:

- päättää hankkeen aloittamisesta, keskeyttämisestä ja päättämisestä
- nimetä hankkeelle ohjausryhmä
- vastata viimekädessä siitä, että hankkeella on käytettävissä riittävät resurssit. (Lemonsoft Oy 2010 a; Pelin, 2009.)



### 7.2.2.2 Hankkeen ohjausryhmä

Ohjausryhmässä on vähintään asiakkaan edustaja, ja lisäksi isoimmissa hankkeissa on suositeltavaa, että mukana on Lemonsoftin edustaja. Myyjä voi olla mukana ohjausryhmässä, mutta tämä on hyvin harvinaista käytönotoissa. (Joki-Hollanti, 2010.) Ohjausryhmä on käyttöönottohankkeen korkein päättävä elin, jonka toimintaa ohjaavat hankkeen omistajan määräykset, sekä yhteisesti päätetyt hanketta koskevat yleisohjeet. (Pelin, 2009.)

Ohjausryhmän tehtävänä on:

- määrittää hankkeelle aikataulu, tavoite ja rajata sen budjetti
- nimetä projektipäällikkö ja hyväksyä projektipäällikön laatima käyttöönottosuunnitelma
- mahdollistaa hankkeelle sen tarvitsemat henkilö- ja muut resurssit
- hyväksyä hankkeen tulos. (Lemonsoft Oy 2010 d; Pelin, 2009.)

### 7.2.2.3 Projektipäällikkö

Projektipäälliköksi valitaan vastuuhenkilö asiakasyrityksestä, koska hänen on oltava työyhteisöstä tuttu henkilö, joka kykenee johtamaan hanketta ja yrityksen henkilöstöä. (Joki-Hollanti, 2010.) Projektipäälliköllä on kokonaisvastuu hankkeesta ja suunnittelusta sekä lisäksi hankkeen toimeenpanosta ja aktiviteettien valvonnasta. Projektipäällikön tehtävänä on raportoida työryhmälle hankkeen etenemisestä ja tapahtumista. (Pelin, 2009.)

Projektipäällikön keskeiset tehtävät ovat:

- laatia projektisuunnitelma tai olla mukana sen laatimisessa
- käynnistää työryhmän työskentely ja ohjata ryhmää
- huolehtia hankkeen dokumentoinnista ja arkistoinnista
- laatia hankkeesta loppuraportti. (Lemonsoft Oy 2010 d; Pelin, 2009.)

#### 7.2.2.4 Hankkeen muut jäsenet

Hankkeessa työskentelee merkittäviä henkilöitä, jotka edustavat oman ammattialansa osaamista. Heiltä edellytetään oman vastualueen ammattitaidon hallintaa ja yhteistyökykyä.

Projektisihteeri tai projektiassistentti, jonka tehtävänä ovat:

- vastata hankkeen asiakirjojen luokittelusta, arkistoinnin suunnittelusta, dokumentoinnin ohjauksesta ja pitää hankkeen viralliset dokumentit ajan tasalla
- laatia aikatauluja ja seurata aikatauluja sekä koordinoita eri vaiheiden aikatauluja
- huolehtia tapaamisjärjestelyistä ja raportoinnista
- sekä olla mukana, kun budjettia laaditaan.

Muita käyttöönottohankkeeseen osallistuvia jäseniä voivat olla konsultit, jotka toimivat hankkeessa asiantuntijaroolissa. Konsulttien lisäksi hankkeeseen voi osallistua IT-päällikkö, IT-tukityöntekijöitä, tehtaan työnjohtajia tai työntekijöitä ja heidän lisäksi myös yrityksen ulkopuolisia asiantuntijoita. Hankkeeseen osallistuvien tehtävänä on:

- osallistua projektisuunnitelman laatimiseen oman vastualueen osalta (tehtävän sisältö, työmäärä ja aikataulu)
- huolehtia asetettujen työtehtävien suorittamisesta
- dokumentoida työn tulokset
- raportoida projektipäällikölle (Lemonsoft Oy 2010 d; Pelin, 2009.)

### 7.3 Valmistelut

Tarkastusvaiheessa käydään läpi yrityksen nykyisen tietojärjestelmän infrastruktuurin nykytila ja toimintaprosessien kypsyys. Tarkoituksena on varmistaa ja etsiä asioita, jotka eivät ole tulleet ilmi aiemmin, esimerkiksi ennen valintaprosessia tai laadittaessa vaatimusmäärittelyä uudelle järjestelmälle. (Joki-Hollanti, 2010.)

Laitteiston nykytason kartoittamisella saadaan selville, mitä muutoksia tai uudistuksia on tehtävä uuden järjestelmän vaatimusten pohjalta. Lemonsoft –ohjelmiston vaatimuksista kerrotaan tarkemmin kohdassa 7.3.5.

### **7.3.1 Tietojärjestelmän infrastruktuuri**

Yrityksen tietojärjestelmän infrastruktuuria tarkastellaan kartoittamalla sitä ympäristöä, jossa Lemonsoft –ohjelmisto tulee toimimaan laitteineen, ohjelmistojen rajapintaliitoksineen, sekä ohjelmiston tulevine käyttäjineen ja alihankkijoineen. (Joki-Hollanti 2010.) Tässä vaiheessa voi syntyä muutostarpeita uuteen toiminnanohjausjärjestelmään tai uusia toimintatapoja tai menetelmiä. Samalla selviää laitteiston nykytaso, mikäli se ei vielä ole selvillä. (Vilpola & Kouri, 2007.)

### **7.3.2 Yrityksen toimintaprosessit**

Yrityksen toimintaprosesseista laaditaan prosessikaavioita, joiden avulla selvitetään mahdolliset muutokset muun muassa tehtävistä toimintatapojen muutoksista. Muutokset saattavat koskea työtehtävissä ilmeneviä päällekkäisyyksiä, joita voi olla kahden tai useamman työntekijän välillä. Toimintaprosessien tutkiminen auttaa toimintatapojen integroimisessa uuteen toiminnanohjausjärjestelmään, jonka tarkoituksena on tehostaa yrityksen toimintaa.

Tutkimalla ja mallintamalla yrityksen pääprosessit ja tukijärjestelmät, saadaan selville yrityksen toimintamalli, joka sisältää yrityksen ydin- ja tukiprosessit. (Vilpola & Kouri, 2007.)

### **7.3.3 Ohjelmistolisenssit**

Lemonsoft ohjelmistolisenssit ovat ohjelmisto- ja käyttäjäkohtaisia. (Lemonsoft Oy 2010 a.) Ohjelmistokohtaisella lisenssillä tarkoitetaan, että lisenssi on sidottu tietokantapalvelimen instanssin-nimeen. Tämä tarkoittaa sitä, että yrityksellä on oikeus käyttää Lemonsoft-ohjelmistoa liiketoiminnassaan, kun yritys hankkii ohjelmistolisenssin (engl. *Device CAL*). (Microsoft Oy 2010 b.) Käyttäjäkohtaisella lisenssillä tarkoitetaan, että yrityksen työntekijällä on oikeus käyttää tiettyä ohjelmamoduulia tai ohjelmistoa,

kun yritys hankkii käyttäjälisenssin tietylle käyttäjämäärälle (engl. *User CAL*). (Microsoft Oy 2010 b.)

Ohjelmistolisenssit voidaan hankkia ostamalla tai vuokraamalla ns. SaaS –menetelmällä sekä hakemalla ohjelmistohankinnalle rahoitusta luottoyhtiöltä. (Joki-Hollanti, 2010; Lemonsoft Oy 2010 a.)

### **7.3.4 Yrityksen hoidettavat aktiviteetit**

Yrityksen vastuulla olevat aktiviteetit ovat: varmuuskopiointi, käyttöönoton aikana tapahtuva dokumentointi (muun muassa käyttöohjeiden laadinta), varautuminen mahdollisiin häiriötilanteisiin sekä testiympäristön perustaminen. Näiden aktiviteettien lisäksi voi olla myös muita hoidettavia aktiviteettejä riippuen muun muassa organisaation tietojärjestelmän rakenteesta, henkilöstönmäärästä ja sijainnista. (Lemonsoft Oy 2010 d.)

#### **7.3.4.1 Varmuuskopiointi**

Tietokannan varmuuskopiointi on yrityksen vastuulla. Tehtävänä on testata varmuuskopioinnin toimivuus ja se, että varmuuskopioitu tiedosto pystytään palauttamaan vikatilanteen tapahtuessa. Tämä on tehtävä säännöllisesti, koska säännöllisesti tehty varmuuskopiointi minimoi tiedon menettämisiä. (Lemonsoft Oy 2010 d.)

#### **7.3.4.2 Dokumentointi**

Yrityksen on tehtävä oma ohjeistus ohjelmiston käytöstä, koska Lemonsoftin ohjeistus on laadittu yleisellä tasolla. Ohjeita syntyy käyttöönoton eri vaiheissa: asennuksessa, tiedonsiirrossa, pilotoinnissa ja testauksessa sekä koulutuksessa. Koulutukseen osallistuvan tulisi vähintään tehdä dokumentti, joka liittyy työntekijän käyttämään työkaluun tai toimintoon Lemonsoft –ohjelmistossa. Tuotettuja dokumentteja voidaan käyttää ohjeistuksena uusille käyttäjille. Dokumentoinnin päävastuu on järjestelmän pääkäyttäjällä. (Lemonsoft Oy 2010 d.)

### 7.3.4.3 Häiriötilanteisiin varautuminen

Mahdollisia vika- ja häiriötilanteita varten on mietittävä varasuunnitelmia etenkin palvelimen tai työasemien hajoamisten varalle. On selvitettävä kauanko uuden laitteen saaminen kestää ja voidaanko mahdollisesti jotakin työasemaa käyttää palvelimena, kunnes uusi laite saadaan. (Lemonsoft Oy 2010 d.)

### 7.3.4.4 Testiympäristö

Testiympäristö on perustettava etenkin isommissa ympäristöissä, koska uutta järjestelmää ei kannata asentaa tuotantokäytössä olevalle palvelimelle. Vaarana voi olla tuotannon pysähtyminen tai häiriintyminen. Ennen tuotantokäyttöön siirtymistä on varmistuttava ohjelmiston toimivuudesta.

Hyvä ja suositeltava tapa perehtyä Lemonsoft -toiminnanohjausjärjestelmään on virtualisoida työasemalle Microsoft Server palvelinohjelmisto. Palvelinohjelmisto voidaan virtualisoida työasemalle esimerkiksi Microsoft Virtual PC -ohjelmiston avulla sekä asentamalla Lemonsoft -ohjelmisto virtuaaliseen ympäristöön. Näin Lemonsoft -ohjelmistoa voidaan ajaa turvallisesti ja paikallisesti, sekä lisäksi tutkia ominaisuuksia ja antaa järjestelmästä palautetta toimittajalle.

Testaus on tehtävä etenkin, jos yritys aikoo käyttää Lemonsoft ERP:n palvelurajapintoja (engl. *Web Service*) hyödyntääkseen kolmannen osapuolen ohjelmistoa. (Lemonsoft Oy 2010 d.)

### 7.3.5 Laitteisto ja ohjelmistovaatimukset

Lemonsoft -ohjelmistot tahdotaan pitää raikkaina ja nykyaikaisina, sekä ohjelmistotekniikaltaan ajan hermolla olevina. Tämä tarkoittaa, että ohjelmistot noudattavat tarkasti mietittyjä ohjelmistostandardeja. Lemonsoft on Microsoftin sertifioitu ISV/Software Solutions -kumppani. Sertifiointi takaa, että Lemonsoftin ohjelmisto on ja säilyy nykyaikaisena noudattaen uusimpia teknologioita. (Lemonsoft Oy 2010 c; Microsoft Oy 2010 c.)

### 7.3.5.1 Palvelinvaatimukset

Palvelimen laitteistojen on oltava vähintään vähimmäisvaatimusten mukaisia, jotta Lemonsoft ERP toimii luotettavasti ja reaaliajassa. Laitteistovaatimusten määrittelyssä kannattaa miettiä sitä, minkälaiselle kuormitukselle palvelin tulee altistumaan. Käyttäjämäärän kartoittamisella voidaan helposti selvittää tarvittava palvelin ja tietoliikenneverkonkapasiteetti. Lisäksi käyttäjämäärän kartoittamisella saadaan selville tarvittava palvelimen käyttöjärjestelmän lisensointitapa.

Palvelimen vähimmäisvaatimukset Lemonsoft ERP:lle on esitetty taulukossa 1.

**Taulukko 1. Vähimmäis palvelinvaatimukset (Lemonsoft Oy 2010 f).**

Käyttöjärjestelmä:	Microsoft Windows Server 2003, SP1
Suoritin:	Yli 1GHz Pentium
Keskusmuisti:	min. 2 GB
Muuta:	.Net Framework 3.5 tai uudempi
	SQL Server 2005 tai Server 2005 Express
	IIS –palvelinohjelmisto (selainliittymille ja raportoinnille)

Palvelimelle on ominaista, että mitä enemmän keskusmuistia, sen paremmin palvelin suoriutuu raskaista operaatioista, kuten esimerkiksi lyhyin väliajoin tehtävistä varmuuskopioinneista.

### 7.3.5.2 Työasemavaatimukset

Työasemavaatimukset poikkeavat palvelinvaatimuksista siten, että työasema ei altistu samanlaiselle käyttäjäkuormalle kuin palvelin. Näin ollen esimerkiksi tietokoneen keskusmuistin määrä ja prosessorin kellotaajuus voi olla matalampi kuin palvelimella. Työaseman ohjelmistotasolla riittää, että työasemalle on asennettu viimeisin Microsoft

.NET Framework ja oikea versio Lemonsoft ERP –asiakasohjelmasta, sekä Microsoft Crystal Reports –raportointiohjelmisto. (Lemonsoft Oy 2010 a.)

Työaseman vähimmäisvaatimukset Lemonsoft ERP:lle on esitetty taulukossa 2.

**Taulukko 2. Vähimmäis työasemavaatimukset (Lemonsoft Oy 2010 f).**

Käyttöjärjestelmä:	Microsoft XP SP2, Windows Vista
Suoritin:	1GHz Pentium
Keskusmuisti:	min. 512 MB (suositus 1 GB)
Muuta:	.Net Framework 3.5 tai uudempi
	Internet -yhteys
	Tulostin
	Skanneri ostolaskujen skannausta varten
	Viivakoodinlukija ostolaskujen ja aikaleimauksia sekä tuotannon leimauksia varten
	Outlook 2003 tai uudempi CRM –integraatioota varten

Palvelinvaatimuksista poiketen käyttäjän työasemaksi sopii hieman kevyempikin laitteisto, jonka avulla Lemonsoft –ohjelmistoa käytetään.

### **7.3.5.3 Tietoliikennevaatimukset**

Tietoliikennevaatimukset määräytyvät käytettävien palvelujen mukaan. Esimerkiksi SaaS –palvelujen käyttö edellyttää nopeaa tietoliikenneyhteyttä, että palvelu toimii reaaliajassa nopeasti ja luotettavasti.

Internet-yhteyttä tarvitaan myös varmentamaan, että yrityksellä on ohjelmistolisenssit ajan tasalla. (Joki-Hollanti 2010.)

#### 7.3.5.4 Microsoft SQL Server

Microsoft –ohjelmistot tukevat tuoreimpia versioita Microsoftin relaatiotietokantapalvelin ohjelmistosta. Tuettuina versioina ovat kaikki SQL Server 2005 ja SQL Server 2008 eri versiot. Ohjelmiston mukana toimitetaan ilmainen SQL Server Express –versio, jolla on omat rajoituksensa. (Microsoft Oy 2010 a.) SQL Server versioiden erot ovat listattuna taulukossa 3. Jossa on SQL Server 2005 versioiden erot ja seuraavalla sivulla taulukossa 4. on SQL Server 2008 versioiden erot. SQL Server versioissa on käytännöllisiä eroja, joten sen takia on hyvä kartoittaa oikea versio ennen käyttöönottosuunnitelman laatimista.

**Taulukko 3. SQL Server 2005 versioiden pääasialliset erot (Microsoft Oy 2010 d).**

Toiminto	Express	Workgroup	Standard	Enterprise
Proessorien määrä	1	2	4	OS MAX
Keskusmuistin määrä	1GB	3GB	OS MAX	OS MAX
64-bit tuki	WOW	WOW	WOW	WOW
Tietokannan max. koko	4GB	ei rajaa	ei rajaa	ei rajaa

WOW on lyhenne, joka tulee nimestä Windows on Windows. Tällä mahdollistetaan 32 - bittisten sovellusten suorittaminen 64 -bittisessä tietojärjestelmässä. (Microsoft Oy 2010 f.)



Taulukko 4. SQL Server 2008 versioiden pääasialliset erot (Microsoft Oy 2010 e).

Toiminto	Ilmaisversiot		Pääversiot		Erikoisversiot		
	Express	Compact	Enterprise	Standard	Workgroup	WEB	Developer
Proessorien määrä	1	OS MAX	OS MAX	4	2	4	OS MAX
Keskusmuistin määrä	1GB	OS MAX	OS MAX	OS MAX	4GB	OS MAX	OS MAX
64-bit tuki	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä	Kyllä
Tietokannan max. koko	4GB	4GB	Rajoittamaton	Rajoittamaton	Rajoittamaton	Rajoittamaton	Rajoittamaton

### 7.3.5.5 Microsoft SQL Server Express Edition -rajoitukset

Microsoft SQL Server Express Edition on Microsoftin kaupallisesta versiosta kevyempi ilmaisversio, joka toimii vain Microsoftin omissa Windows -ympäristöissä. Ilmaisversion rajoituksena on yhden prosessorin tuki, rajoitettu yhden gigatavun keskusmuistin määrä ja tietokannan suurin koko voi olla enintään neljä gigatavua. (Microsoft Oy 2010 d; Microsoft Oy 2010 e.)

Lisäksi puutteina on tietokannan ylläpidon kannalta seuraavia merkittäviä toimintoja:

- ei töiden ajastettuja toimintoja, jonka myötä esimerkiksi tietokannan varmuuskopiointia ei voida tehdä ajastetusti. Tämän rajoituksen vuoksi asiakkaan on huolehdittava manuaalisesti tehtävästä varmuuskopiointista
- ei tiedon tuonti- tai vienti -toimintoja
- tietokannan peilaaminen on estetty. (Microsoft Oy 2005.)

## 7.4 Asennusvaihe

Asennusta on suunniteltava ennen sen toteuttamista.

- Mitä on asennettava?
- Kuka asentaa?
- Minne asennetaan?
- Missä järjestyksessä asennetaan?

Nämä aktiviteetit on sisällytettävä käyttöönottosuunnitelmaan ja jakaa tehtävät hankkeeseen osallistuville henkilöille.

### 7.4.1 Asennuspaketit

Asennuspaketit saa ladattua helposti Lemonsoftin FTP –palvelimelta, jos käyttäjä on saanut tunnuksen ja salasanan palvelimelle.

Palvelimella olevat paketit ovat ZIP –paketteja, jotka on merkitty aikaleimalla, joka kertoo sen, kuinka tuore ohjelmistoversio on kyseessä. Paketin nimen perässä on myös merkintä siitä, onko kyseessä virallinen julkaisu (r), esiversio (CTP) tai esijulkaisuversio (RC).

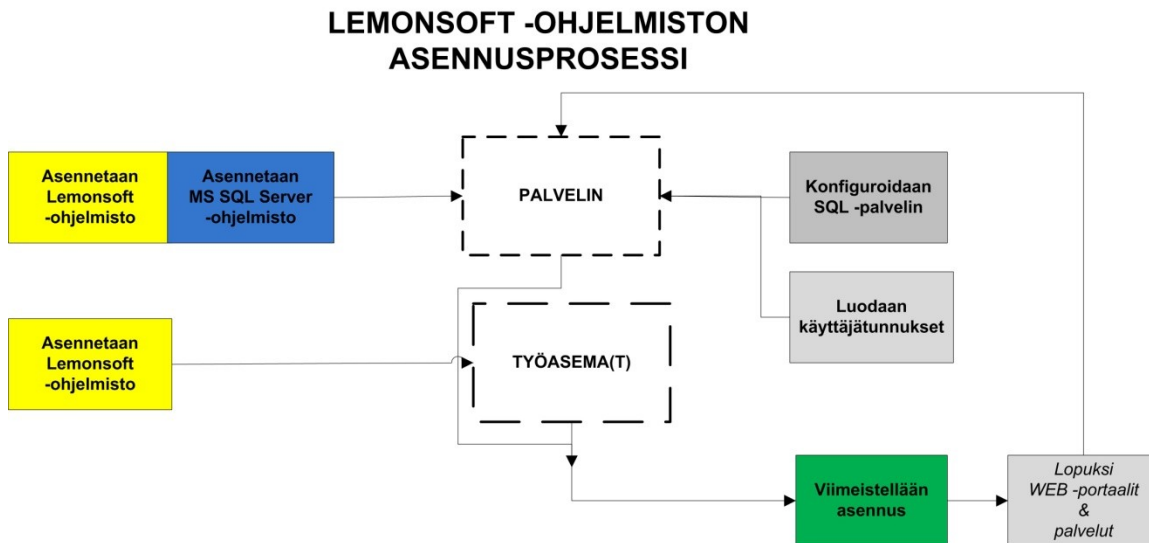
Paketit on nimetty seuraavasti:

- LemonsoftFull paketti sisältää kaiken tarvittavan tyhjän koneeseen
- LemonsoftUpdate paketti sisältää vain päivityksen ohjelmistosta
- LemonsoftMobile paketti sisältää ohjelmiston PDA-laitteita varten
- LemonsoftWebServices paketti sisältää ohjelmiston palvelurajapinnasta
- LemonsoftWebPortal paketti sisältää ohjelmiston portaalia varten
- LemonsoftWinServices paketti sisältää Windows palvelut.

Esimerkki nimeämiskäytännöstä:

LemonsoftFull\_20100406\_r.zip eli kyseessä on asennuspaketti, joka on päivätty 6.huhtikuuta 2010. Kirjain r kertoo sen, että ohjelmisto on virallinen julkaisu. (Lemonsoft Oy 2010 a.)

## 7.4.2 Asennusprosessi



Kuva 14. Yksinkertainen asennusprosessi (Lemonssoft Oy 2010 a).

Lemonssoft –ohjelmiston asentaminen yksinkertaiseen verkkoympäristöön tapahtuu kuvan 16. mukaisesti:

1. asennetaan Lemonssoft- ja MS SQL Server –ohjelmisto samalle palvelimelle
2. asennetaan Lemonssoft –ohjelmisto jokaiselle työasemalle
3. luodaan kaikille käyttäjätunnukset SQL –palvelimelle
4. konfiguroidaan SQL –palvelinta esimerkiksi kytketään etäyhteystoiminto päälle
5. viimeistellään asennusvaihe käynnistämällä ja aktivoimalla Lemonssoft –ohjelmisto
6. lopuksi voidaan asentaa mahdolliset WEB -portaalit ja palvelut. (Lemonssoft Oy 2010 a.)

### 7.4.2.1 Lemonssoft ohjelmiston asennus työasemalle

Kun käynnistetään haluttu asennuspaketti, niin asennusohjelma tarkistaa onko koneelle jo asennettu Microsoft .NET Framework v3.5. Jos ei ole, niin asennusohjelma lataa paketin Microsoftin palvelimelta ja asentaa sen. (Lemonssoft Oy 2010 a.)

Työasemalle pakolliset ohjelmistot ovat Microsoft Crystal Reports ja Lemonsoft -ohjelmisto, mutta Microsoft SQL Server ei ole pakollinen työasemille. (Lemonsoft Oy 2010 a.)

Asennusohjelma kopioi tiedostoja työasemalle aikansa ja ilmoittaa lopuksi onnistuneesta asennuksesta. (Lemonsoft Oy 2010 a.) Mikäli asennusohjelma ilmoittaa ongelmista, niin virhesanoma tulee kirjoittaa ylös paperille ja ottaa yhteyttä myyjään tai tukeen. Vaihtoehtoisesti ongelma voidaan yrittää ratkaista niin, että etsitään vastausta <http://www.lemonsoft.fi/nethelp/> -sivustolta.

#### **7.4.2.2 Lemonsoft ohjelmiston asennus palvelimelle**

Toimivaan ympäristöön riittää, että asennetaan Lemonsoft –ohjelmisto työasemalle. Suositeltavaa olisi myös saada käynnistää Lemonsoft –asiakasohjelmisto palvelimelta mahdollisen vikatilanteen sattuessa tai kiireellisen etätyön tarpeessa. Palvelimeen saadaan helpommin yhteys kuin työasemaan, koska yleensä työasemat sammutetaan työpäivän päätteeksi. (Lemonsoft Oy 2010 a.)

Kun ohjelmistoa asennetaan palvelimelle, niin noudatetaan samoja ohjeita kuin työasemallekin ohjelmistoa asentaessa.

#### **7.4.3 Käyttäjät**

Ennen käyttöönottovaihetta on syytä miettiä, ketkä yrityksessä ovat oikeutettuja saamaan pääkäyttäjän roolin Lemonsoft ERP –ympäristössä. Luonnollista on, että roolin saa vähintään yrityksen talouspäällikkö tai toimitusjohtaja. (Lemonsoft Oy 2010 d.) Lisäksi on päätettävä, miten käyttäjät tunnistetaan ja miten käyttöoikeudet jaetaan käyttäjille. Vaihtoehtoina käyttäjän tunnistautumiselle ovat joko Windows- tai SQL –autentikaatio. (Joki-Hollanti 2010.)

##### **7.4.3.1 Windows –autentikaatio**

Windows –autentikaatiossa käyttäjätunnukset ovat tallennettuna yrityksen AD –palvelimelle. Palvelimelle säilytetään käyttäjätunnusten lisäksi erilaisia objekteja, kuten esimerkiksi verkon tietokoneiden tietoja, käyttäjäryhmiä jne. AD tarjoaa yrityksen verkon

tietokoneille kirjautumispalvelun eli lyhyesti AD on keskitetty käyttäjätunnusten hallintajärjestelmä. (Microsoft Oy 2009 a.)

Käytettäessä Windows –autentikaatiota yhdessä Lemonsoft -ohjelmistojen kanssa. Käyttäjienhallinta on mahdollista hoitaa keskitetysti. Käyttäjille voidaan jakaa käyttöoikeudet lisäämällä käyttäjät tiettyyn käyttäjäryhmään, jolla on oikeudet käyttää Lemonsoft –ohjelmistoa tai ohjelmiston tiettyä moduulia. Itse Lemonsoft –ohjelmistossa on vielä erikseen yksityiskohtaisempi käyttöoikeuksien jakomahdollisuus, jossa voidaan jakaa oikeudet käyttää tiettyjä moduuleja ja toimintoja. Käyttäjien tiedot sijaitsevat kahdessa eri paikassa:

- AD –palvelimella on käyttäjätunnus ja salasana sekä mahdollisesti käyttäjän sähköpostiosoite.
- Lemonsoft –ohjelmistossa on käyttäjän henkilökohtaiset tiedot koskien osoitetta, työsuhdetta, palkkatietoja, projekteja.

Käyttäjän tunnistaminen Lemonsoft –järjestelmää varten tapahtuu, kun käyttäjä kirjautuu käyttämään tietokoneensa Windows -käyttöjärjestelmää, joka on rekisteröity käyttämään yrityksen AD –palvelinta käyttäjätunnistamisessa. (Joki-Hollanti, 2010.)

#### **7.4.3.2 Tietokanta –autentikaatio**

Tietokanta –autentikaatiolla tarkoitetaan, että käyttäjien tunnus, salasana ja käyttöoikeudet on luotu suoraan SQL –palvelimelle. Tunnus voi olla sama tai erillinen, joka on käyttäjän tietokoneen Windows –käyttöjärjestelmässä, mutta tunnus on silti perustettava erikseen SQL -palvelimelle. Kun käyttäjä aikoo käyttää Lemonsoft -ohjelmistoa, täytyy hänen antaa käyttäjätunnus ja salasana joka kerta uudelleen. (Microsoft Oy 2009 b; IT Tool Box 2002.)

### **7.5 Tiedonsiirrot**

Ennen tietojen siirron toteuttamista, toimenpidettä on syytä suunnitella ja siihen on varattava aikaa. Huomioon täytyy ottaa mitä tietoja siirretään uuteen järjestelmään, voidaanko tietoja siirtää automaattisesti ja mitkä niistä täytyy syöttää käsin järjestelmään.

Koesiirrolla varmistetaan, että siirto on mahdollista toteuttaa ja tieto on mahdollisen ehyttä. Tiedon siirtäminen on mahdollista toteuttaa määrämuotoisesta tiedostosta tai suoraan tietokannasta, jos siihen päästään kiinni. Yleisemmin vanhasta järjestelmästä siirretään tuotenimikkeet, tuotteiden rakenteet, asiakkaat ja toimittajat. Näiden lisäksi on mahdollista tuoda tilikartat, työvaiheet, kustannuspaikat, projektit, palkkalajit, hyllysoitteet, ehdot, henkilöt, tarjoukset, myynti- ja ostotilaukset, myyntilaskut ja –reskontra, ostoreskontra, palkat, tuotannon avoimet työt sekä hinnastot. (Lemonsoft Oy 2010 d.)

Tiedon eheyttämisellä vältetään turhalta työltä, ja etenkin tiedonsiirron uudelleen toteuttamiselta. Ennen tietojen siirtämistä voidaan miettiä, onko järkevää siirtää esimerkiksi kaikkia asiakkaita, koska joukossa saattaa olla sellaisiakin asiakkaita, joilla ei ole enää toimintaa tai yritykset ovat fuusioituneet. Lisäksi on tärkeää huomioida CRM:n tarpeet ajantasaisten asiakastietojen osalta. (Lemonsoft Oy 2010 d.)

### **7.5.1 Tiedonsiirtotavat**

Lemonsoft mahdollistaa ohjelmallisen siirron, jolla tarkoitetaan sitä, että Lemonsoft tekee räätälöidyn sovelluksen tiedonsiirtoa varten tai asiakas hyödyntää palvelurajapintoja toteuttaakseen WWW – sovelluksen tiedonsiirtoa varten. Palvelurajapintojen lisäksi asiakkaalla on mahdollisuus siirtää informaatiota Lemonsoft –ohjelmiin CSV tai XML muodossa, mutta vaihtoehto vaatii paljon manuaalista ja ohjelmallista muokkausta. Kätevin ja helpoin tapa lienee tuoda ensin siirrettävä data Excel –työkirjaan, eheyttää data ja siirtää eheytetty työkirjan sisältö tietokantaan. (Lemonsoft Oy 2010 d.)

### **7.5.2 Tiedonsiirtorajapinnat**

Lemonsoft -yritysohjelmistot on avoin ohjelmisto. Asiakkaalla on laajat mahdollisuudet tehdä omia Lemonsoft yhteensopivia sovelluksia, käyttäen hyväksi samaa ohjelmistorajapintaa mitä Lemonsoft -ohjelmistokin. Sovellukset hyödyntävät palvelurajapintoja, joiden läpi luetaan ja kirjoitetaan informaatiota Lemonsoftiin. (Lemonsoft Oy 2010 a.)

### **7.5.3 Lopullinen tiedonsiirto**

Lopullinen tiedonsiirto tehdään käyttöönottohetkellä vain siinä tapauksessa, jos vanhaan järjestelmään on lisätty sellaista informaatiota, jota ei ole vielä uudessa järjestelmässä. Tiedonsiirtoa ei tarvitse kuitenkaan tehdä alusta saakka, sillä uudet tiedot on mahdollista tuoda vanhasta järjestelmästä, joten tiedonsiirtoa ei tarvitse tehdä alusta saakka. (Lemonsoft Oy 2010 d; Joki-Hollanti, 2010.)

## **7.6 Muutokset**

Mahdolliset muutokset syntyvät, kun pilotointivaiheessa tai hankkeen toteutusvaiheen aikana havaitaan, että ohjelmisto ei ole määriteltyjen ominaisuuksien mukainen. Pilotointivaiheessa kirjataan kaikki poikkeavaa toimintaa oleva informaatio, kuten ohjelmistovirheet, puutteet tulosteissa, sekä yrityksen kannalta merkittävien toiminnallisuuksien toimimattomuus, kuten esimerkiksi kolmannen osapuolen laitteet tai ohjelmistot. Pilotointivaiheessa havaitut ongelmat, virheet ja poikkeavuudet listataan muutosvaatimusmäärittelyyn ja pisteytetään tärkeysjärjestykseen. Luetteloidut tiedot kirjataan Lemonsoftin omaan muutostenhallintatyökaluun ja muutokset pyritään korjaamaan ennen varsinaista ohjelmiston käyttöönottoa. Tehdyt muutokset lisätään ohjelmakoodiin, josta muodostetaan uusi versio ohjelmistosta. (Joki-Hollanti, 2010.)

### **7.6.1 Pilotointi**

Pilotointi on rajattu kokeiluhanke, joka on tärkeä osa järjestelmätestausta ja osa pääkäyttäjäkoulutusta. Pilotointi toteutetaan asiakkaan tuotantoympäristössä, ennalta suunnitellulla osastolla, ennen järjestelmän siirtämistä tuotantokäyttöön. Pilotoinnin avulla asiakas testaa ohjelmistoa omassa ympäristössään käyttäen omia resursseja, kuten esimerkiksi työntekijöitä, yrityksen tietojärjestelmän infrastruktuuria ja toimintaprosesseja. Pilotoinnilla on määrä varmistaa, että määritellyt ominaisuudet toimivat, ja lisäksi sillä varmistetaan koko järjestelmän toimivuus. Pilotoinnin avulla voidaan minimoida käyttöönoton riskejä ja näin käyttöönotto sujuu aikataulullisesti hallitummin. Pilotointi vaiheessa syntyy dokumentaatio, joka auttaa henkilökunnan koulutuksessa ja käyttöönotossa. Pilotointivaihe ei yksinään riitä järjestelmän

hyväksymiseksi, vaan pilotointia seuraa yleensä varsinainen hyväksyntä testaus. (Märijärvi & Haikala, 2004; Halonen.)

### **7.6.2 Ohjelmamuutokset**

Lemonsoft –ohjelmistoon tehtävät muutokset voivat olla nopeita raporttiräätälöintejä, jotka voidaan jopa toteuttaa paikanpäällä muokkaamalla ohjelmiston konfigurointitiedostoja. Tällaisia raporttiräätälöintejä ovat esimerkiksi logojen, nimien tai rivitietojen asemointi tulosteisiin. Hieman suurempia ja hitaammin toteutettavissa olevia muutoksia ovat Lemonsoft –ohjelmistoon tehtävät muutokset, jotka sovitetaan 4 kk julkaisuaikatauluun. Kriittisten muutosten toteuttaminen on mahdollista, mutta niiden toteuttamisesta on sovittava erikseen ja ne toteutetaan kauppasopimuksen kirjoittamisen jälkeen. (Lemonsoft Oy 2010 d; Joki-Hollanti, 2010.)

### **7.6.3 Muutostenhallinta**

Lemonsoftilla muutostenhallinta perustuu ketterien ohjelmistokehitysten menetelmille. Pyydetty muutokset kirjataan asiakkaan toimesta muutostenhallintajärjestelmään, josta Lemonsoftin ohjelmistokehittäjät voivat muutoksia tarkastella ja ottaa työlistalle. Tehdyt muutokset lisätään järjestelmään ja aika-ajoin syntyy uusi ohjelmistoversio Lemonsoft ERP:stä (noin kahden viikon välein). Syntynyt ohjelmistoversio on tarkoitettu vain testikäyttöön, joten sitä ei suositella käytettävän tuotantokäytössä. (Joki-Hollanti, 2010.)

### **7.6.4 Ohjelmamuutosten lisääminen järjestelmään**

Lisätty ohjelmamuutos on testattava heti, kun muutos on asennettu tai tehty järjestelmään. Tehdyn muutoksen toiminnasta on raportoitava Lemonsoftille, jos toiminnallisuudessa on vielä korjattavaa tai muutettavaa. (Joki-Hollanti, 2010.)

### **7.6.5 Hyväksymistestaus**

Hyväksymistestauksella asiakas simuloi uuden järjestelmän toimintaympäristöä, jonka testaamisen tukena käytetään tuotettuja käyttöoppaita ja menettelyjä. (Murch, 2002.) Hyväksymistestaus on suoritettava ennen varsinaista käyttöönottoa.



## 7.7 Koulutus

Jokainen henkilöstöryhmä koulutetaan käyttämään Lemonsoft –toiminnanohjausjärjestelmää. Henkilökunnan kouluttaminen on tärkeä osa käyttöönottoa. Osa pidettävistä koulutuksista voidaan pitää käyttöönoton aikana tai jättää käyttöönoton jälkeiseen aikaan. Koulutuksella sitoutetaan henkilökuntaa ja vähennetään muutosvastarintaa jota syntyy, kun uusia toimintatapoja otetaan käyttöön. Käyttöönoton aikana pidetyt koulutustilaisuudet varmistavat käyttöönoton sujumista. (Lemonsoft Oy 2010 d.)

Koulutus noudattaa ennalta laadittua koulutussuunnitelmaa. Koulutuksissa hyödynnetään Lemonsoftin laatimia koulutusrunkoja ja koulutusmateriaaleja. Asiakas on laatinut koulutussuunnitelman arvioiden henkilökunnan tarvitsemaa koulutusta tarvekartoittamisella, esimerkiksi tarvitseeko henkilökunta koulutusta tuotannonohjausjärjestelmän perustietojen osalta vai töiden lisäämisessä tuotantoon. Tarkemmin voit lukea koulutusrunгон sisällöstä kohdasta 7.7.1.

Ennen käyttöönottohetkeä pidetyt koulutukset mahdollistavat, että työntekijät voivat olla mukana osallistumassa tarkastuksiin ja järjestelmän muuntamiseen sekä antamassa palautetta koulutuksen onnistumisesta. (Murch, 2002, 120.)

### 7.7.1 Esimerkki koulutusrungosta

Tässä esimerkissä käydään läpi tuotannonohjaukseen liittyvät asiat tuotantopäällikön työtehtävien osalta. Koulutuksen yhteydessä perustetaan tuoterakenteita ja -vaiheita sekä näiden lisäksi mietitään vakiotyövaiheita ja perustetaan ne järjestelmään. Tuotantoväen kouluttamisella pyritään antamaan edellytykset, että tuotannonohjausjärjestelmä voidaan ottaa käyttöön tuotannossa.

Tuotantopäällikön koulutusrunko pitää sisällään Lemonsoft –tuotannonohjausjärjestelmästä seuraavat asiat:

1. Perustiedot, jotka voidaan kouluttaa ennen varsinaista käyttöönottohetkeä.  
Koulutus sisältää: tuote tietojen ja tuote rakenteiden perustamisen,

- varastopaikkojen ja hyllyosoitteiden perustamisen, työvaiheiden-, työvuorojen ja töihin liittyvien asioiden kouluttamisen.
2. Kuinka lisätään töitä tuotantoon. Koulutus sisältää: kuinka töitä lisätään tuotantoon myyntitilauksista, varastosta, ostotyökalusta.
  3. Varastosaldot. Koulutus sisältää: materiaalitarpeiden ja varausten tarkastelun tuotannosta ja varastosta.
  4. Tuotannon kuormituksen tarkastelun koulutus sisältää: kuormitusasteiden tarkastelun sekä koneittain että ryhmittäin, ja lisäksi toimitusaikojen tarkastelun.
  5. Tuotannossa olevien töiden tarkastelun koulutus sisältää: työjonon ja tuotantokalenterin tarkastelun ja muuttamisen.
  6. Työaikojen raportoimisen koulutus sisältää: manuaalisesti tehtävät raportoinnit, kirjauksien optiot, leimauspäätte (rfid ja viivakoodi), tietojen korjaaminen ja tarkistaminen, töiden keskeyttäminen, viallisten tuotteiden kirjaaminen, töiden paloittelu sekä palkanlaskenta.
  7. Alihankinta. Koulutus sisältää: alihankinta töiden määrittelyn ja käsittelyn.
  8. Työnumerot. Koulutus sisältää: tuotantolistan ja työjonon hallinnan.
  9. Tuotantotilanne. Koulutus sisältää: koontitöiden ja yksittäisten töiden hallinnan.
  10. Koontityöt. Koulutus sisältää: tuotantolistojen ja työjonojen hallinnan.
  11. Jälkilaskenta. Koulutus sisältää: jälkilaskennan raportoinnin (standardi aika vs. toteuma), ostojen kohdistamisen työnumerolle.
  12. Muu raportointi. Koulutus sisältää: keskeneräisen tuotannon ja tuotantopäiväkirjan tarkastelu.
  13. Sarjanumerot. Koulutus sisältää: valmistettavan tuotteen, komponentin jäljittäminen.
  14. Eränumerot: Koulutus sisältää: valmistettavan tuotteen, komponentin jäljitettävyyden.
  15. Tiedon tuonti, vienti ja integrointi. Koulutus sisältää: tiedon käsittelyn ohjelmistojen välillä. Esimerkiksi tiedon vienti Lemonsoftista Exceliin tai päinvastoin. (Lemonsoft Oy 2010 e.)

### **7.7.2 Käyttöohjeet**

Varsinaisten käyttöohjeiden tuottaminen on ehdottomasti koulutukseen osallistuvien henkilöiden vastuulla, koska Lemonsoftin tuottamat ohjeet ovat hyvin yleisellä tasolla. Koulutukseen osallistuvat tuottavat käyttöohjeita, joita voidaan hyödyntää myöhemmin esimerkiksi uuden henkilön kouluttamisessa tai loman jälkeen muistin virkistämiseen. (Joki-Hollanti, 2010.)

### **7.8 Varsinainen käyttöönotto**

Uuden toiminnanohjausjärjestelmän käyttöönottopäivästä on tiedotettava etukäteen henkilökunnalle ja heidät on koulutettava ennen käyttöönottohetkeä vähintään käyttämään heidän työssään tarvitsemiaan toimintoja. Käyttöönottohetki ei mielellään saisi ajoittua esimerkiksi kiireaikoihin eikä myöskään ennen kesälomia, koska henkilökunta saattaa unohtaa loman aikana koulutuksissa opitut asiat.

#### **7.8.1 Vanha järjestelmä**

Vanha järjestelmä syrjäytetään tuotantokäytöstä, mutta jätetään taustalle esimerkiksi tietojen tarkastelua varten, jos jotakin tietoa ei ole saatu siirrettyä uuteen järjestelmään. (Joki-Hollanti, 2010.)

Käyttöönottopäivänä olisi suositeltavaa estää kirjausten tekeminen vanhaan järjestelmään esimerkiksi siten, että järjestelmästä voidaan vain lukea vanhoja tietoja. Toinen mahdollisuus on estää käyttäjiltä uusien tietojen kirjaaminen vanhaan järjestelmään. Vanhan järjestelmän pääkäyttäjän on suotavaa silti päästä täysin oikeuksin kiinni vanhaankin järjestelmään. Lisäksi mahdollisesti on poistettava työntekijöiden työpisteiltä vanhan järjestelmän käyttöohjeet ja manuaalit, jos sellaisia on. Työpisteille tulee tuoda uuden järjestelmän käyttöohjeet, koska kaikki kirjaukset on tehtävä nyt uuteen järjestelmään, jotta yrityksen- ja tuotannon toiminta olisi ajantasaista ja häiriötöntä. (Joki-Hollanti, 2010.)

### **7.8.2 Uusi järjestelmä**

Uusi järjestelmä otetaan käyttöön pääjärjestelmäksi koko yrityksessä. Järjestelmän käyttöönotto on mahdollista toteuttaa vaiheittain, esimerkiksi ottamalla käyttöön kriittiset yritystoiminnot ensimmäisenä ja tuotanto viimeisenä. Käyttöönottopäivänä paikalle on mahdollista saada asiantuntijoita ongelmatilanteiden ratkaisemiseksi, mutta tästä menettelystä on sovittava erikseen. (Joki-hollanti, 2010.)

## 8 YHTEENVETO

Projekti selkiytti käsitystä toiminnanohjausjärjestelmistä ja siitä, minkälainen projekti toiminnanohjausjärjestelmän käyttöönotto kokonaisuutena on. Alan kirjallisuudesta ja palaverissa saadun tiedon pohjalta, toiminnanohjausjärjestelmän käyttöönottoprosessin vaiheet selkenivät. Ilman toimeksiantoyrityksen antamaa panosta, tätä työtä ei olisi ollut mahdollista toteuttaa. Kirjallisuudessa ja käytännössä faktat eivät täysin tue toiminnanohjausjärjestelmän käyttöönottoa, koska ohjelmistot, yritykset ja käyttöönotto-organisaatiot ovat erilaisia. Lyhyesti sanottuna ei ole olemassa yhteneväistä opasta, jonka avulla saisi toiminnanohjausjärjestelmän lähes ongelmitta käyttöön yrityksessä. Järjestelmän käyttöönotto vaatii asianomaisilta asiaan perehtymistä ja resursseja, jotta hanke onnistuu täysin ja jotta saadaan luotua edellytykset yrityksen paremmalle toiminnalle.

Työn tavoitteena ollut opas pätee vain Lemonsoft –toiminnanohjausjärjestelmään, joten sitä ei voi suoraan soveltaa muiden järjestelmien käyttöönottoon. Työn tavoitteet saavutettiin lähtökohtiin nähden hyvin, koska aihealueen tuntemus oli vähäistä. Tuloksena syntyi opas, jota voidaan kehittää eteenpäin esimerkiksi luomalla johonkin projektinhallintaohjelmistoon vastaavanlainen käyttöönottorunko, jonka avulla asiakas voi hallita käyttöönottohanketta tehokkaammin ja visuaalisemmin.

Toiminnanohjausjärjestelmän käyttöönotto vähänkään isommassa organisaatiossa on erittäin haasteellista. Yrityksen on tiedettävä toiminnanohjausjärjestelmältä vaadittavat ominaisuudet ennen käyttöönoton tapahtumista, koska tällöin on vaarana, että ohjelmisto ei sovellu lainkaan yrityksen käyttöön. Valittua ohjelmistoa joudutaan mahdollisesti räätälöimään, jotta ohjelmisto istuisi paremmin yrityksen toimintamalleihin. Käyttöönottoprojektin onnistumisen kannalta haasteita esiintyy asiakkaan ja toimittajan resurssien riittävydessä ja henkilöstön sitouttamisessa. Yrityksen on tunnettavat omat prosessit, jotta toiminnanohjausjärjestelmää voidaan edes harkita otettavan käyttöön yrityksessä, koska ilman prosessien tuntemusta järjestelmä ei sulaudu yrityksen toimintaan.

## LÄHTEET

Allbusiness 2001. Does ERP Fit in a LEAN World. [online] [viitattu: 26.4.2010] Saatavilla [www-muodossa:<URL:http://www.allbusiness.com/management/785112-1.html>](http://www.allbusiness.com/management/785112-1.html)

Fonecta 2008. Taloustiedot [viitattu: 21.04.2010] Saatavilla [www-muodossa <URL:http://profindexb2b.fonecta.com/companysingle.aspx?cID=7089415>](http://profindexb2b.fonecta.com/companysingle.aspx?cID=7089415)

Grotenfelt, Outi & Ilomäki, Liisa & Närvänen, Liisa. 1989. Tietojärjestelmän toteutus ja käyttöönotto. Helsinki, Valtion painatuskeskus.

Haikala, Ilkka & Märijärvi, Jukka. 2004. Ohjelmistotuotanto. 10p. Hämeenlinna, Karisto Oy

Halonen, Raija. Tietojärjestelmän vaihtaminen. Oulun yliopisto. Tietojenkäsittelytieteiden käsittelylaitos. Tapaustutkimus.

Hankintatieto 2010. Hankintamenettelyt [online] [viitattu 26.4.2010] Saatavilla [www-muodossa:<URL:http://www.hankintatieto.fi/index.php/hankintatieto/Tietoa-hankinnoista/Hankintamenettelyt>](http://www.hankintatieto.fi/index.php/hankintatieto/Tietoa-hankinnoista/Hankintamenettelyt)

IT Tool Box 2002. Windows Authentication vs. SQL Server Authentication. [online] [viitattu: 10.05.2010] Saatavilla [www-muodossa:<URL:http://database.ittoolbox.com/documents/windows-authentication-vs-sql-server-authentication-18609>](http://database.ittoolbox.com/documents/windows-authentication-vs-sql-server-authentication-18609)

JHS-Suosituksset 2008. JHS 167 Neuvottelumenettelyjen käyttö ICT -hankinnoissa. [online] [viitattu 22.04.2010] [www-muodossa:<URL:http://docs.jhs-suositukset.fi/jhs-suositukset/JHS167/JHS167.html>](http://docs.jhs-suositukset.fi/jhs-suositukset/JHS167/JHS167.html)

Joki-Hollanti, Kari, toimitusjohtaja 6.5.2010. Lemonsoft Oy, Vaasa. Haastattelu.

Jyväskylän yliopisto 2010. Ketterien menetelmien ja trac koulutuksen sisältö [viitattu: 18.05.2010] Saatavilla [www-muodossa:<URL:https://webapps.jyu.fi/koppa/avoimet/thk/agile-ja-trac/agile/scrum-esittely>](http://www.muodossa:<URL:https://webapps.jyu.fi/koppa/avoimet/thk/agile-ja-trac/agile/scrum-esittely>)

Kajava, Merja & Nykopp, Sebastian 2001. Ketterä mallinnus. Systemityö. Nro 4, 2-4.

Karessuo, Anna-Kaisa 2003. Toiminnanohjausjärjestelmät – ERP ja ERP II. Aalto yliopisto. Automaation tietotekniikka. Seminaarityö.

Kettunen, Sami 2002. Tietojärjestelmän ostaminen – käytännön opas yrityksille. Porvoo, WS Bookwell Oy.

Lemonsoft Oy 2009. Ratkaisukuvaus. 2009.

Lemonsoft Oy 2010 a. Ohjeistus. [online] [viitattu 16.4.2010] Saatavilla [www-muodossa:<URL:http://www.lemonsoft.fi/nethelp/>](http://www.muodossa:<URL:http://www.lemonsoft.fi/nethelp/>)

Lemonsoft Oy 2010 b. Tuotteet. [viitattu: 17.4.2010] Saatavilla [www-muodossa: <URL: http://www.lemonsoft.fi/tmp\\_lemon08\\_site\\_5.asp?lang=1&sua=3&q=y&s=212>](http://www.muodossa:<URL:http://www.lemonsoft.fi/tmp_lemon08_site_5.asp?lang=1&sua=3&q=y&s=212>)

Lemonsoft Oy 2010 c. Yritys. [viitattu: 17.4.2010] Saatavilla [www-muodossa: <URL: http://www.lemonsoft.fi/tmp\\_lemon08\\_site\\_0.asp?lang=1&sua=3&q=y&s=173>](http://www.muodossa:<URL:http://www.lemonsoft.fi/tmp_lemon08_site_0.asp?lang=1&sua=3&q=y&s=173>)

Lemonsoft Oy 2010 d. Kohderyhmä. [viitattu: 25.05.2010] Saatavilla [www-muodossa: <URL: http://www.lemonsoft.fi/tmp\\_lemon08\\_site\\_0.asp?lang=1&sua=3&q=y&s=174>](http://www.muodossa:<URL:http://www.lemonsoft.fi/tmp_lemon08_site_0.asp?lang=1&sua=3&q=y&s=174>)

Lemonsoft Oy 2010 d. Koulutusmateriaali.

Lemonsoft Oy 2010 e. Tuotannon koulutusrunko.

Lemonsoft Oy 2010 f. Vähimmäisvaatimukset.

Microsoft Oy 2005. SQL Server Express Edition Overview [online] [viitattu: 27.4.2010] Saatavilla [www-muodossa:<URL:http://msdn.microsoft.com/en-us/library/ms345154\(SQL.90\).aspx>](http://msdn.microsoft.com/en-us/library/ms345154(SQL.90).aspx)

Microsoft Oy 2009 a. Active Directory Domain Services. [online] [viitattu: 10.5.2010] Saatavilla [www-muodossa:<URL:http://msdn.microsoft.com/en-us/library/aa362244\(VS.85\).aspx>](http://msdn.microsoft.com/en-us/library/aa362244(VS.85).aspx)

Microsoft Oy 2009 b. Choosing an Authentication Modes. [online] [viitattu: 10.05.2010] Saatavilla [www-muodossa:<URL:http://msdn.microsoft.com/en-us/library/ms144284.aspx>](http://msdn.microsoft.com/en-us/library/ms144284.aspx)

Microsoft Oy 2010 a. Microsoft Dynamics NAV. [viitattu: 19.05.2010] Saatavilla [www-muodossa:<URL:http://www.microsoft.com/finland/dynamics/nav/default.mspix>](http://www.microsoft.com/finland/dynamics/nav/default.mspix)

Microsoft Oy 2010 b. Lisenssisanasto. [online] [viitattu: 28.4.2010] Saatavilla [www-muodossa:<URL:http://www.microsoft.com/finland/license/lisenssisanasto.mspix>](http://www.microsoft.com/finland/license/lisenssisanasto.mspix)

Microsoft Oy 2010 c. Sertifiointi. [online] [viitattu: 07.05.2010] Saatavilla [www-muodossa:<URL:https://partner.microsoft.com/finland/partner>](https://partner.microsoft.com/finland/partner)

Microsoft Oy 2010 d. SQL Server 2005. [Online] [viitattu: 29.04.2010] Saatavilla [www-muodossa:<URL:http://www.microsoft.com/SqlServer/2005/en/us/compare-features.aspx>](http://www.microsoft.com/SqlServer/2005/en/us/compare-features.aspx)

Microsoft Oy 2010 e. SQL Server 2008 [Online] [viitattu 29.04.2010] Saatavilla [www-muodossa:<URL:http://www.microsoft.com/sqlserver/2008/en/us/editions.aspx>](http://www.microsoft.com/sqlserver/2008/en/us/editions.aspx)

Microsoft Oy 2010 f. WOW64 Implementation Guide. [online] [viitattu: 27.4.2010] [<URL:http://msdn.microsoft.com/en-us/library/aa384274\(v=VS.85\).aspx>](http://msdn.microsoft.com/en-us/library/aa384274(v=VS.85).aspx)

Mountain Goat Software 2010. Scrum. [viitattu: 19.05.2010] Saatavilla [www-muodossa:<URL:http://www.mountaingoatsoftware.com/scrum/daily-scrum>](http://www.mountaingoatsoftware.com/scrum/daily-scrum)

Much, Richard 2002. IT-Projektinhallinta. Helsinki, Edita Prima Oy.

Pelin, Risto 2009. Projektinhallinnan käsikirja. 6p. Jyväskylä, Gummerus Kirjapaino Oy.

Silfverberg, Paul. [viitattu: 20.4.2010] Saatavilla [www-muodossa: <URL:http://www.mol.fi/esf/ennakointi/raportit/pvopas.pdf>](http://www.mol.fi/esf/ennakointi/raportit/pvopas.pdf)



Sininen Meteori 2010 a. Ketterät käytännöt – Agile [viitattu 18.05.2010] Saatavilla www-muodossa:<URL:<http://www.ketteratkaytannot.fi/fi-FI/Menetelmat/AM/>>

Sininen Meteori 2010 b. Ketterät käytännöt – Scrum [viitattu: 18.05.2010] Saatavilla www-muodossa:<URL:<http://www.ketteratkaytannot.fi/fi-FI/Menetelmat/Scrum/>>

Sysoptima 2005. History and Evolution of ERP [viitattu: 22.04.2010] Saatavilla www-muodossa <URL:[http://www.sysoptima.com/erp/history\\_of\\_erp.php](http://www.sysoptima.com/erp/history_of_erp.php)>

TAIK koulutuskeskus 1998. WWW-palveluiden käytettävyys ja tuotanto. [viitattu: 18.05.2010] Saatavilla www-muodossa:<URL:<http://www.uiah.fi/mediastudio/survey4/24.html>>

Talentum & Ruuska, Kai. 2007. Pidä projekti hallinnassa – suunnittelu, menetelmät ja vuorovaikutus. 6.painos. Talentum Media Oy

Tampereen yliopisto. Projektisuunnitelma [online] [viitattu 16.4.2010] Saatavilla www-muodossa:<URL:<http://www.uta.fi/tvema/projektit/projektisuunnitelma.html>>

The Financial Express 2008. ERP Implementation - The Hidden Cost. [online] [viitattu: 11.05.2010] Saatavilla www-muodossa:<URL:<http://www.financialexpress.com/news/erp-implementationthe-hidden-cost/277216/>>

Tieke 2008. ERP luultua tärkeämpi pk-yrityksille. [online] [viitattu: 20.4.2010] Saatavilla www-muodossa:<URL:[http://www.tieke.fi/tieke/tieken\\_tiedotteet\\_2008/erp\\_luultua\\_tarkeampi\\_pk-yrityks/](http://www.tieke.fi/tieke/tieken_tiedotteet_2008/erp_luultua_tarkeampi_pk-yrityks/)>

TTL-julkaisusarja 2002. Tietojärjestelmän hankinta – Ohjelmistotoimittajan ja ratkaisun valinta. Vantaa, Tummavuoren kirjapaino Oy.

Tuotantotalouden perusopinnot 2010. Kuopion yliopiston Avoimessa yliopistossa. [viitattu 9.11.2009] Saatavilla www-muodossa:<URL:[http://www.uku.fi/avoin/tuta/j4\\_sisallys.htm](http://www.uku.fi/avoin/tuta/j4_sisallys.htm)>

Vilpola, Inka & Kouri, Ilkka 2006. Toiminnanohjausjärjestelmän hankinta C-CEI-menetelmän avulla. Vantaa, Dark Oy.

Visma 2010. Visma Nova. [viitattu 19.05.2010] Saatavilla www-muodossa:<URL: <http://www.visma.fi/Ohjelmistoratkaisut/Nova/>>

Yrittäjät.fi 2010. Yrittäjän ABC. [online] [viitattu 6.5.2010] Saatavilla www-muodossa:<URL: [http://www.yrittajat.fi/fi-FI/yritystoiminnanabc/julkiset\\_hankinnat/](http://www.yrittajat.fi/fi-FI/yritystoiminnanabc/julkiset_hankinnat/)>