



**LAHDEN AMMATTIKORKEAKOULU**  
*Lahti University of Applied Sciences*

# TOIMINNANOHJAUSJÄRJESTELMÄN TESTAUSPALVELUN TUOTTEISTUS

Case: Yritys X

LAHDEN  
AMMATTIKORKEAKOULU  
Liiketalouden ala  
Liiketalouden koulutusohjelma  
Opinnäytetyö  
Syksy 2014  
Harald Gangsö

Lahden ammattikorkeakoulu  
Liiketalouden koulutusohjelma

GANGSÖ, HARALD:

Toiminnanohjausjärjestelmän testauspalvelun tuotteistus  
Case: Yritys X

Liiketalouden opinnäytetyö

60 sivua, 1 liitesivua

Syksy 2014

TIIVISTELMÄ

---

Tämä opinnäytetyö käsittelee toiminnanohjausjärjestelmän testauspalveluiden tuotteistamista. Tutkimus sisältää ensin katsauksen toiminnanohjausjärjestelmiin, niiden laatutekijöihin sekä niiden käyttöönottoon liittyvään testaukseen, jonka jälkeen käsitellään asiantuntijapalvelun tuotteistamista. Tutkimus on tehty tapaus-tutkimuksena Yritys X:n osastolla A, joka toimittaa asiakkaille Dynamics AX-toiminnanohjausjärjestelmiä. Tietolähteinä on käytetty kirjallista ja sähköistä materiaalia.

Toiminnanohjausjärjestelmä, eli ERP-järjestelmä, on yritysten ja julkishallinnon käyttöön tarkoitettu tietojärjestelmä, joka yhdistää kaikki organisaation toiminnot yhdeksi kokonaisuudeksi ja tallettaa kaiken tiedon yhteen tietokantaan. ERP-järjestelmien tarkoituksena on tehostaa yrityksen toimintaa tehostamalla kaikkia yrityksen prosesseja ja siten tuottaa asiakkaille tuotteita ja palveluita kilpailijoita nopeammin ja edullisemmin.

Toiminnanohjausjärjestelmä on organisaation keskeisin tietojärjestelmä ja sen oikea ja luotettava toiminta on hyvin tärkeää. Järjestelmän laadun varmistamiseksi toimittajan tulee testata järjestelmää kattavasti järjestelmäkehityksen aikana, minkä lisäksi asiakkaan tulisi tehdä järjestelmän hyväksymistestaus ennen käyttöönottoa.

Tutkimuksen tarkoituksena on kartoittaa case-yksikön asiakaskunnassa tarvetta nykyistä laajemmalle tuotteistetulle testauspalvelulle toiminnanohjausjärjestelmien käyttöönotto- ja versionvaihdosprojekteissa. Lisäksi tarkoituksena on selvittää, kuinka yksikössä voidaan kehittää ohjelmistokehityksen aikaista testausta tehokkaammaksi. Tutkimuksen aineisto on kerätty haastatteleamalla järjestelmätoimittajan ja asiakkaan edustajia sekä tutkijan omana osallistuvana havainnointina.

Keskeisenä tuloksena havaittiin, että asiakkailla on usein huomattavia haasteita järjestelmän testauksessa muun muassa resurssien riittävyyden ja testauksen suunnittelun osalta. Lisäksi havaittiin, että asiakasyrityksissä ei osattu ennakoida testaukseen käytettävää aikaa. Toimittajalla oleva testauksen asiantuntemus on tuotteistettavissa ja sille on asiakaskunnassa kysyntää. Lisäksi toimittajan omia testausprosesseja voidaan tehostaa muun muassa kouluttamalla henkilökuntaa.

Asiasanat: Toiminnanohjausjärjestelmä, ERP, testaus, tuotteistus, asiantuntijapalvelu, Dynamics AX

Lahti University of Applied Sciences  
Degree Programme in Business Studies

GANGSÖ, HARALD:

Productization of Testing Services for  
ERP-Software Deliveries  
Case: Company X

Bachelor's Thesis in Business Studies 60 pages, 1 page of appendices

Autumn 2014

ABSTRACT

---

This thesis deals with the survey of productization of testing services for ERP-software (Enterprise Resource Planning). First, a general review about ERP-software, their quality attributes and testing prior deployment is presented as well as the theoretical background of the productization of expert service. The study was carried out as a case study for Company X's department A which delivers Dynamics AX -ERP-software for its customers. Sources for the theoretical section of this study include thematically related publications and the Internet.

ERP-software is a data system for corporations and the public sector. The software integrates all functions within an organization as a whole and all data is saved in one database. The purpose of ERP-software is to streamline the processes in the organization and thus enable the organization to deliver products and services to customers faster and cheaper than the competitors.

ERP-software is the most important software in an organization and therefore it is vital that it works correctly and reliably. In order to ensure high quality of the software the vendor has to test the software comprehensively during the software development. In addition to the testing done by the supplier, the customer should do acceptance testing before deployment.

The aim of the study was to examine customer needs for a productized testing service during software implementation and upgrade projects. Furthermore an attempt was made to find out how to enhance the internal testing processes to the supplier. Sources include interviews carried out with representatives of the customers as well as the supplier and the author's own observations.

The study results indicate that the customers had significant challenges with the testing of the software due to insufficient resources and testing plans. Moreover it was observed that the customers were not able to estimate the time needed for testing. There is a demand for testing services and the productization of a supplier's testing expertise can be developed. The internal testing processes to the supplier can be enhanced for instance by training the personnel in testing processes.

Key words: Enterprise Resource Planning, testing, productization, expert services, Dynamics AX

## SISÄLLYS

1	JOHDANTO	1
1.1	Tutkimuksen tausta ja hyödyt	3
1.2	Tutkimuksen tavoitteet, tutkimusongelmat ja rajaukset	4
1.3	Tutkimusmenetelmät	6
1.4	Tutkimuksen rakenne	7
2	ERP-JÄRJESTELMÄT JA NIIDEN TESTAUS	8
2.1	ERP-järjestelmät	8
2.2	ERP-järjestelmän toimitusprojekti	10
2.3	ERP-järjestelmän laatutekijät	12
2.4	Testaus	15
2.4.1	Testaus osana laadunvarmistusta	17
2.4.2	Testauksen vaiheet	18
2.5	Testauksen organisoiminen	23
2.5.1	Testausorganisaatio	26
2.5.2	Testauksen suunnittelu	27
2.5.3	Testauksen raportointi	28
3	ASiantuntijapalvelun tuotteistus	30
3.1	Asiantuntijapalvelu	30
3.2	Palvelun tuotteistus	33
3.3	Tuotteistamisen hyödyt yritykselle	36
4	CASE: YRITYS X	37
4.1	Tutkimuksen toteutus	37
4.2	Projektitoiminnot osastolla A	38
4.3	Microsoft Dynamics Sure Step	41
4.4	Tutkimuksen tulokset	43
4.4.1	ERP-järjestelmän laatu	43
4.4.2	ERP-järjestelmän testaus osana toimitusprojektia	45
4.4.3	Toimittajan tarjoamat testauksen lisäpalvelut	48
4.5	Johtopäätökset	49
5	YHTEENVETO	54
	LÄHTEET	58
	LIITTEET	

## Työn keskeiset käsitteet

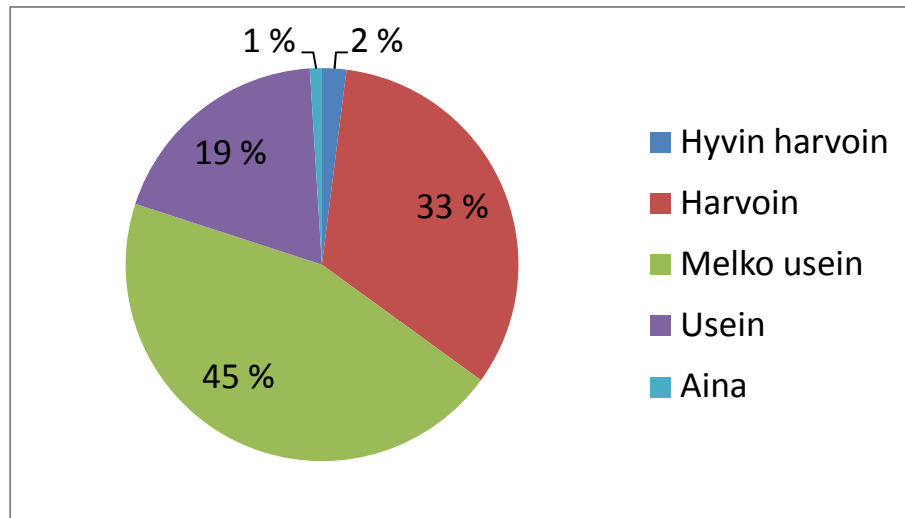
Tutkimuksessa on käytetty ISTQB:n (International Software Testing Qualifications Board 2007) testaussanastoa ja niiden suomenkielisiä käännöksiä. Testausvaiheisiin liittyvät käsitteet on selitetty kappaleessa 2.4 ja projektin vaiheet on selitetty kappaleessa 4.3.

ERP-järjestelmä	Toiminnanohjausjärjestelmä eli ERP-järjestelmä on yritysten käyttöön tarkoitettu järjestelmä, joka yhdistää organisaation eri toiminnot yhteen tietojärjestelmään
Toimitusprojekti	Projekti, jonka tavoitteena on toimittaa asiakasyritykselle ERP-järjestelmä. Kyseessä voi olla joko järjestelmän käyttöönotto tai versionvaihdos
Käyttöönotto	ERP-järjestelmien toimitusprojekteissa kriittinen vaihe, jossa valmistaudutaan ja otetaan käyttöön uusi järjestelmä
Perusjärjestelmä	Valmis ERP-järjestelmä, kuten Microsoft Dynamics AX, johon ei ole tehty asiakaskohtaisia sovituksia.
Vesiputousmenetelmä	Vaiheittainen ohjelmistokehityksen menetelmä
Agile-menetelmä	IT-projekteissa yleistynyt menetelmä, jossa yhdistetään analysointi-, suunnittelu- ja toteutusvaiheet. Tunnettu myös ketteränä menetelmänä.
Moduuli	ERP-järjestelmä koostuu moduuleista, eli osista, joita voidaan ottaa käyttöön

## 1 JOHDANTO

Opinnäytetyö käsittelee toiminnanohjausjärjestelmien (Enterprise Resource Planning, jatkossa käytetään lyhennettä ERP-järjestelmä) toimituksiin liittyvää testaus-  
ta. ERP-järjestelmät ovat yritysten ja julkishallinnon käyttöön suunniteltuja tietojärjestelmiä, jotka integroituvat yritysten eri toimintoihin ja siten pystyvät parantamaan yrityksen toimintaa eri tavoin. Nykyaikaiset ERP-järjestelmät kattavat valtaosan yrityksen tarpeista ja niiden oikea ja tehokas toiminta on yrityksille hyvin tärkeää. Tämän päivän liiketoimintakulttuuri yhdistettynä nykuteknologiaan mahdollistaa ERP-järjestelmän sovittamisen asiakkaan tarpeita vastaavaksi kokonaisuudeksi. Näin ollen järjestelmän toimittaja tekee useimmiten järjestelmään asiakaskohtaisia muutoksia, minkä seurauksena järjestelmän testauksesta tulee merkittävä osa toimitusprojektia.

ERP-järjestelmien kuten muidenkin IT-järjestelmien toimitusprojektit herättävät paljon huomiota epäonnistuessaan. Järjestelmien laajuudet ovat kasvaneet teknologian kehittyessä. Järjestelmien ollessa yhä monimutkaisempia, synnyttävät ne merkittäviä riskejä projekteissa, jotka realisoituessaan voivat tarkoittaa aikataulujen ja budjettien ylittymisiä. Tietojärjestelmien hankinta Suomessa –tutkimuksen mukaan 35 % tietojärjestelmähankintojen tilaajista kokee onnistuvansa hankinnoissa harvoin tai hyvin harvoin ja vain 20 % koki onnistuvansa usein tai aina. Tutkimukseen vastasi yhteensä 104 tilaaja- ja toimittajaorganisaatioita edustavaa henkilöä. (Tietojärjestelmien hankinta Suomessa 2013.) Hankinnoissa epäonnistuminen voi johtua budjetin tai aikataulun ylittymisestä tai järjestelmän epätyytyvästä toiminnasta. Kuviossa 1 on kuvattu onnistumisen jakautuminen tilaajien itsensä arvioimana.



KUVIO 1. Tietojärjestelmähankkeiden onnistumisen todennäköisyys tilaajanäkökulmasta (Tietojärjestelmien hankinta Suomessa 2013).

Tavoitteen eli liiketoiminnallisen hyödyn karkaaminen järjestelmän ulottumattomiin voi johtua ulkoisista muutoksista liiketoimintaympäristössä, mutta olettaen, että investoinnit ylipäätään tehdään vakaasti harkiten, voidaan sanoa, että valtaosa järjestelmistä ei vastaa niitä laatuodotuksia, jotka asiakas on järjestelmälle asettanut. Odotetun laatutason saavuttaminen edellyttää järjestelmän testausta eri valmistuksen vaiheissa. Samaisen tutkimuksen mukaan tilaajista 40 %:n ja toimittajista 51 %:n mukaan tilaajaorganisaation vastuuhenkilöiden resurssit ovat harvoin tai ei koskaan riittävät (Tietojärjestelmien hankinta Suomessa 2013). Ottaen huomioon testauksen merkityksen tavoitteen saavuttamisessa ja riittävän testauksen edellyttämän aktiivisen yhteistyön toimittajan ja tilaajan välillä, voidaan todeta tilaajaorganisaation vastuuhenkilöiden riittämättömien resurssien olevan suuri riski lopullisen laadun tuottamisessa, koska testauksella pyritään arvioimaan etukäteen järjestelmän laatua. (Burnstein 2002, 26-28.)

## 1.1 Tutkimuksen tausta ja hyödyt

Opinnäytetyö on tehty Yritys X:n osasto A:n toimeksiannosta. Kyseinen osasto toimittaa Microsoft Dynamics AX –ERP-järjestelmiä suomalaisille yrityksille ja yhteisöille. Microsoft Dynamics AX on globaalisti toimiville yrityksille tarkoitettu liiketoimintaratkaisu, joka kattaa ERP-järjestelmän toiminnot taloushallinnosta henkilöstöhallintoon (Microsoft 2014). Itselläni on useamman vuoden kokemus Dynamics AX:sta käyttäjänä järjestelmää käyttävän organisaation palveluksessa sekä konsultin roolissa järjestelmätoimittajan palveluksessa.

Microsoft on testannut Dynamics AX-järjestelmän ennen sen markkinoille tuloa, joten perusjärjestelmälle ei tarvitse tehdä laajamittaista testausta asiakkaan toimesta. Asiakaskohtaiset sovitukset vaikuttavat puolestaan merkittävästi ERP-järjestelmän toimintaan ja halutun toiminnallisuuden käyttöönotto edellyttää huomattavan paljon testausta. Järjestelmän testaus tulisi tehdä järjestelmätoimittajan ja asiakkaan välisenä yhteistyönä oikean lopputuloksen varmistamiseksi. Testaus on pitkä prosessi ja se alkaa jo toiminnallisuuden määrittämisen yhteydessä, jolloin testauksen suunnittelu aloitetaan. Asiakkaille ERP-järjestelmään investoiminen on suuri investointi ja usein tarpeettomaksi koettu karsitaan pois. Karsitus toimitussopimuksen osaksi voi helposti joutua minimitason ylittävä testaus, mikäli ostajan johdossa ei ole riittävä ymmärrystä testauksen vaikutuksesta lopputulokseen. Minimitasolla tarkoitetaan välttämätöntä testausta, jolla todetaan järjestelmän toimivan normaalissa prosessissa tyydyttävällä tasolla.

Testauksen puute vaikuttaa ennen kaikkea järjestelmän oikeellisuuteen, käytettävyyteen ja luotettavuuteen. Vaikka järjestelmä itsessään normaalitilanteessa toimii tyydyttävällä tasolla, saattaa käyttäjä kokea, ettei järjestelmä vastaa odotettua laatua. Sopimusvaiheessa testauksen tärkeys hyvän lopputuloksen aikaansaamiseksi tulisi pystyä osoittamaan asiakkaalle. Varsinainen testaustyö edellyttää tekijältään ainakin jossakin määrin ymmärrystä tietotekniikasta ja toteutetuista ratkaisuista. Tilaajalla ei välttämättä ole riittäviä resursseja eikä riittävä osaamista tehokkaan ja kattavaan testaukseen. Vastaavasti toimittajalla on osaamista tietotekniikasta sekä kokemusta testauksesta. Kuitenkin osaamisen myyminen tilaajalle on toimeksiantajayksikössä koettu hankalaksi. Ratkaisuna voisi olla testauspalvelui-



den tuotteistaminen, jolloin toimittajan osaaminen pystytään myymään tehokkaammin tilaajalle (Tiensuu & Kari 2006, 27).

Yritys X:lle testauspalvelut luovat mahdollisuuden myydä testausosaamista ja kasvattaa näin liikevaihtoa ja tulosta. Lisäksi testauspalveluiden myyminen voisi parantaa asiakkaiden kokemaa laatua ja siten parantaa asiakastyytyvääsyyttä, mikä edesauttaa uusista kaupoista neuvoteltaessa. ERP-järjestelmien myynnissä asiakastyytyväisyys ja referenssit ovat tärkeässä osassa.

## 1.2 Tutkimuksen tavoitteet, tutkimusongelmat ja rajaukset

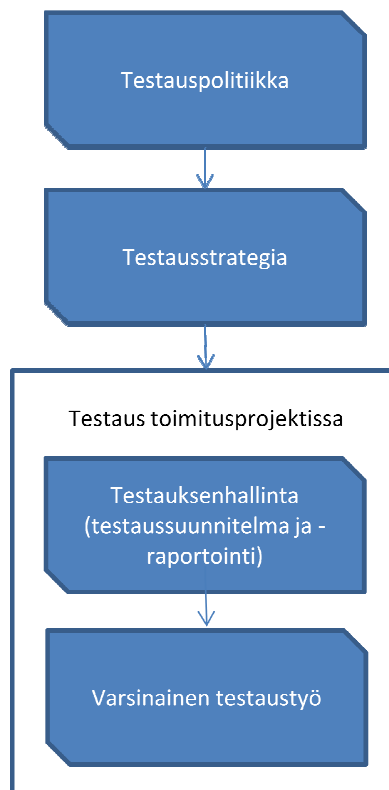
Tutkimuksen tavoitteena on tutkia ohjelmistokehitykseen liittyvää testausta kokonaisuutena sekä kartoittaa kokevatko Dynamics AX-toimitusprojektien asiakkaat tarvetta laajemmalla tuotteistetulle testauspalvelulle ja millä tavalla palvelua voitaisiin tuotteistaa. Lisäksi tutkimuksen tavoitteena on selvittää, kuinka testausprosesseja voidaan kehittää toimittajan näkökulmasta toimittajan omassa järjestelmän kehitystyössä toimitettujen järjestelmien laadun parantamiseksi.

Tavoitteena on toimittajalla parantaa asiakassovitettujen ERP-järjestelmien laatua, jotta ne vastaavat paremmin asiakkaan odottamaa laatua. Lisäksi tavoitteena on laajentaa osaamista testauksen saralla sekä laajentaa sitä kautta liiketoimintaa kattamaan laajemmat testauspalvelut kuin tähän asti asiakkaille on tarjottu. Perinteisesti testaus on suoritettu kehittäjien ja asiantuntijoiden toimesta. Testaus on kuitenkin vaativa prosessi, joka edellyttää erityistä osaamista, joten sen eriyttäminen omaan tiimiin on myös mahdollista.

Tässä opinnäytetyössä etsitään vastausta neljään tutkimusongelmaan.

- Onko yrityksen Dynamics AX-toimitusprojekteissa suoritettu testaus laadunvarmistuksen näkökulmasta riittävää?
- Onko asiakasyrityksillä riittävä osaaminen kattavaan ja tehokkaaseen hyväksymistestaukseen?
- Kokevatko asiakasyritykset tarvetta laajennetulle testauspalvelulle?
- Kuinka toimittaja voi kehittää Dynamics AX-toimitusprojektien testausprosesseja?

Opinnäytetyö on rajattu käsittämään Yritys X:n Dynamics AX –ERP-järjestelmän toimitusprojektien testausoiminnot. Tyypillinen toimitusprojekti on suuryritykselle myyty Dynamics AX-järjestelmä, johon on tehty merkittävässä määrin asiakas-kohtaisia muutoksia. Työssä keskitytään vesiputousmallilla, eli vaiheellisella ohjelmistotuotantoprosessilla, toimitettuihin projekteihin. Ketterien ohjelmistotuotantoprosessien mukaiset projektit on rajattu opinnäytetyön ulkopuolelle. Samaten organisaation testauspolitiikka ja –strategia sekä testausautomaatio on rajattu opinnäytetyön ulkopuolelle. Kuvio 2 havainnollistaa organisaation testauksen jakautumisen neljään eri tasoiseen osa-alueeseen. Ylemmällä tasolla oleva osa-alue vaikuttaa alempana olevan osa-alueen ratkaisuihin. Opinnäytetyössä käsitellään toimitusprojektiin sisältyviä osa-alueita, eli testauksenhallintaa ja varsinaista testaustyötä. Teorettisessa viitekehyksessä on käsitelty lyhyesti organisaatiotason osa-alueita, eli testauspolitiikkaa ja –strategiaa.



KUVIO 2. Tutkimuksen rajaukset (mukaillen Kasurinen 2011, 31)

### 1.3 Tutkimusmenetelmät

Tutkimusmenetelmä on laadullinen tapaustutkimus, koska se soveltuu menetelmänä parhaiten työhön. Organisaation tavoitteena on selvittää ja kartoittaa melko pienessä asiakaskunnassa tarvetta testauspalvelulle sekä selvittää, kuinka omaa toimintaa voidaan kehittää. Aikaisempaa tutkimustietoa aiheesta ei ole käytettävissä. (Saaranen-Kauppinen & Puusniekka 2006.)

Tutkimusaineisto kerätään teemahaastatteluina ja tutkijan omana osallistuvana havainnointina. Teemahaastattelussa haastatellaan sekä toimittajan että asiakkaan edustajia, joilla on kokemusta Yritys X:n toimittamista Microsoft Dynamics AX – ERP-järjestelmäprojekteista sekä mahdollisesti kokemusta myös muista toiminnanohjausjärjestelmäprojekteista. Teemahaastatteluun valitaan 5 – 7 sopivaa henkilöä ja haastattelut toteutetaan syksyllä 2014. Haastattelun teemat ovat tutkimuksen liitteenä (liite 1). Koska asiakaskunta on melko pieni ja ilmiö suhteellisen tuntematon, valittiin aineistonkeruutavaksi teemahaastattelut, koska siinä haastateltava henkilö voi melko vapaasti kertoa kokemuksistaan. Teemahaastattelun avulla voidaan selvittää toistaiseksi tuntematonta ilmiötä ja saada laaja kuva testaustarpeesta omassa asiakaskunnassa, jonka avulla testausta voidaan kehittää edelleen asiakkaiden tarpeita vastaavaksi. (Saaranen-Kauppinen & Puusniekka 2006.)

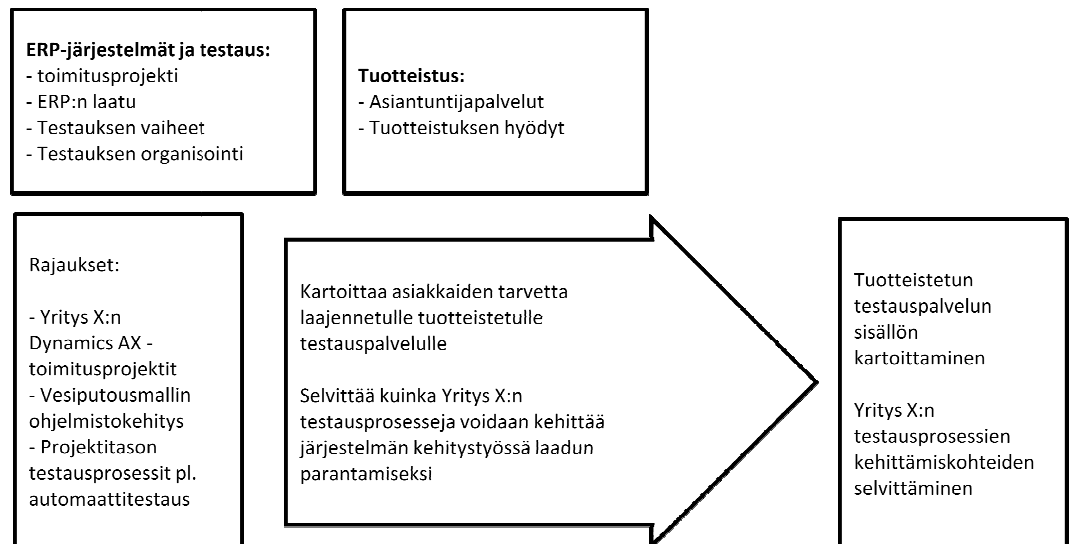
Tietojärjestelmien testauksesta on aikaisempaa tutkimusaineistoa, muun muassa Jussi Kasurisen väitöskirja sovellusten testausprosessien kehittämistä sekä Lyytinen & Hirschheimin empiirinen tutkimus tietojärjestelmäprojektien epäonnistumisten syistä. Aikaisempaa tutkimustietoa on käytetty hyväksi tutkimuksen teoriaosassa ja empirian teemojen valinnassa. On kuitenkin huomattava, että eri IT-järjestelmissä testauksen tarve ja kriittisyys sekä asiakkaan että toimittajan roolit testauksessa vaihtelevat riippuen onko kyseessä kokonaan räätälöity järjestelmä, valmisohjelmisto vai edellisten välimaastoon asettuva valmisohjelmisto, johon on tehty asiakaskohtaisia sovituksia. Suurin osa testaukseen liittyvistä teoksista keskittyvät hankkeisiin, jossa valmistetaan kokonaan uusi ohjelmisto.

## 1.4 Tutkimuksen rakenne

Tämä opinnäytetyö koostuu viidestä pääluvusta. Ensimmäisessä pääluvussa, johdannossa, on kerrottu tutkimuksen taustasta sekä esitetty tutkimuksen tavoitteet sekä tutkimusongelmat. Toisessa pääluvussa on kuvattu ERP-järjestelmän toimitamiseen ja sen testaukseen liittyvä teoreettinen viitekehys. Kappaleessa käsitellään myös tietojärjestelmien laatutekijöitä sekä niiden yhteyttä testaukseen. Seuraavassa pääluvussa esitetään asiantuntijapalveluiden tuotteistamiseen liittyvää teoreettista viitekehystä sekä asiantuntijaorganisaatiolle koituvia hyötyjä tuotteistamisesta.

Neljännessä pääluvussa esitellään kohdeyritys ja tutkimuksen tulokset sekä johtopäätökset. Viimeisessä luvussa, eli yhteenvedossa käsitellään vastaukset tutkimusongelmiin, pohditaan saatujen tulosten käyttökelpoisuutta sekä tutkimuksen reliabiliteettia ja validiteettia. Lisäksi esitetään jatkotutkimusaiheet.

Kuviossa 3 on esitetty tutkimuksen rakenne, pääkohdat, tavoitteet sekä rajaukset.



KUVIO 3. Tutkimuksen pääkohdat, tavoitteet ja rajaukset

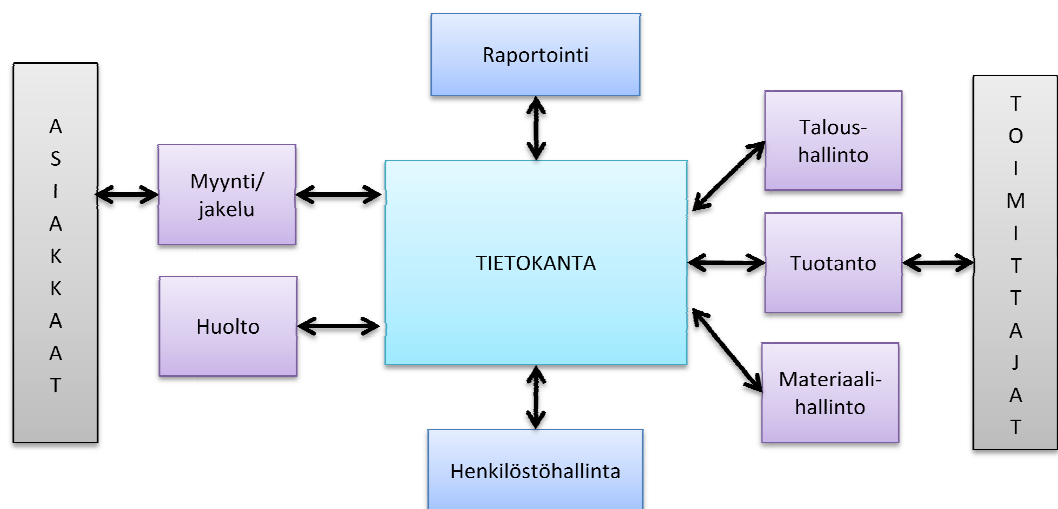
## 2 ERP-JÄRJESTELMÄT JA NIIDEN TESTAUS

Seuraavissa luvuissa käsitellään ensin ERP-järjestelmien käsite, toimitusprojektien sisältö sekä tietojärjestelmien laatutekijät. Sen jälkeen käsitellään tietojärjestelmien testaukseen liittyvää teoriaa.

### 2.1 ERP-järjestelmät

ERP-järjestelmä on tietojärjestelmä, joka integroi saumattomasti yrityksen eri osastot ja niiden tietojärjestelmät yhdeksi kokonaisuudeksi ja yhteen tietokantaan. Kokonaisuus hyödyttää kaikkia osastoja ja kaikkia yrityksen tarpeita, koska sen avulla koko yrityksen tieto pystytään tarjoamaan kaikkien osastojen käytettäväksi samanaikaisesti ja joustavasti. Eri osastot ja toiminnot käyttävät hyväkseen tietojärjestelmään tuotettua tietoa prosesseissa, ja näin pystytään vähentämään manuaalisia työvaiheita ja virhemahdollisuuksia. (Wailgum 2008; Thorley 2014.)

Kuviossa 4 on kuvattu ERP-järjestelmälle tyypillinen rakenne. Yrityksen eri toiminnot ovat linkitettyinä yhteiseen tietokantaan, joka yhdistää eri moduulit toisiinsa. Asiakkaan tekemä tilaus on heti kaikkien tietoa tarvitsevien toimintojen tiedossa, ja se voi esimerkiksi aiheuttaa ostotilauksen syntymisen järjestelmässä automaattisesti.



KUVIO 4. ERP-järjestelmän rakenne (Thorley 2014)

ERP-järjestelmä on yrityksen liiketoiminnan kannalta merkittävin tietojärjestelmä. Yksittäisten eri tarkoituksiin soveltuvien järjestelmien sijaan yrityksissä on yksi järjestelmä, johon kaikki erilliset järjestelmät integroidaan ja johon kaikki yrityksen merkittävät toiminnot ovat tavalla tai toisella integroitu. Nykyisessä yhteiskunnassa tietomäärän yhä kasvaessa yrityksissä tarvitaan järjestelmiä, joihin voidaan tallettaa tietoa ja josta sitä voidaan reaaliaikaisesti tehokkaasti hakea analysointia varten. Tähän tarpeeseen ERP-järjestelmät vastaavat tallentamalla kaiken tiedon yhteen tietokantaan, josta tieto on haettavissa eri prosesseihin. (Monk & Wagner 2007, 17.)

ERP-järjestelmät ovat kehittyneet 1970-luvun MRP-järjestelmistä (Material Requirements Planning), jotka keskittyivät materiaalitarpeen laskentaan, ja 1980-luvun MRP II-järjestelmistä (Manufacturing Resources Planning), joissa materiaalitarpeen lisäksi laskettiin tuotteen optimaalista reittiä tuotannossa sekä kapasiteettia. MRP II-järjestelmiin lisättiin myös integraatioita taloushallintoon, jakeluun sekä henkilöstöhallintaan. 1990-luvulla kehitettiin ERP-järjestelmät, joihin voitiin integroida kaikki yrityksen toiminnot. Luonnollisesti kehitys on jatkunut ja järjestelmiin on lisätty 2000-luvulla esimerkiksi verkkokauppa- ja CRM-toimintoja. (Thorley 2014.)

ERP-järjestelmien tarkoituksena on tehostaa yrityksen toimintaa tehostamalla kaikkia yrityksen prosesseja ja siten tuottaa asiakkaille tuotteita ja palveluita kilpailijoita nopeammin ja edullisemmin. ERP-järjestelmän ylläpitämisen yhden tietokannan ansiosta eri yrityksen toiminnot kykenevät käyttämään hyväkseen reaaliaikaisesti tietokantaan muiden toimintojen tallentamaa tietoa ja siten yrityksen sisäistä tiedonkulkua parannetaan. Esimerkiksi ostotoiminnoissa pystytään reagoimaan nopeasti tilanteeseen, jossa todellinen myynti ylittää tai alittaa aikaisemmat myyntiennusteet ja näin ollen ostomääriä pystytään oikaisemaan. (Monk & Wagner 2007, 1, 17.)

Tässä opinnäytetyössä keskitytään ERP-järjestelmistä Microsoft Dynamics AX:an. Järjestelmä kuuluu Microsoftin ERP- ja CRM-järjestelmät kattavaan Dynamics-tuoteperheeseen. AX on suurille yrityksille tarkoitettu liiketoimintaratkaisu, mikä tarkoittaa, että järjestelmä täyttää paitsi ERP-järjestelmälle asetetut vaa-

timukset, myös sisältää muuta liiketoiminnan edellyttämää toiminnallisuutta, kuten tehokkaat business intelligence- ja raportointitoiminnallisuudet. Se soveltuu niin vähittäiskaupalle, rahoitus- ja palvelualalle, valmistukseen kuin julkiselle sektorille. (Microsoft 2014.)

Dynamics AX:n historia alkaa tanskalaisen Damgaard A/S:n vuonna 1998 Tanskassa ja Yhdysvalloissa julkaisemasta ERP-järjestelmästä, jonka nimi oli Axapta 1.0. Aluksi ohjelma tarjosi ratkaisut taloushallinnolle, varaston ja tuotannonhallinnolle sekä logistiikalle ja myynnille. Microsoft osti yhtiön vuonna 2002 ja on sen jälkeen jatkanut järjestelmän kehitystyötä. Uusia versioita julkaistaan muutama vuoden välein, mutta sen lisäksi on useita pienempiä julkaisuja, joissa on pienempiä lisäyksiä aiempaan toiminnallisuuteen sekä havaittujen virheiden korjauksia. Microsoftin aikana järjestelmän käyttöliittymää on muokattu siten, että se muistuttaa enemmän Microsoftin Office –ohjelmia. Järjestelmää on myös kehitetty siten, että se on yhteensopiva Office-ohjelmien kanssa. (Sycor GmbH 2014.)

AX-järjestelmää hankittaessa ostetaan kerralla koko järjestelmä kaikkine moduuleineen. Kaikkia moduuleja ei kuitenkaan yleensä oteta käyttöön, vaan käyttöönottohetkellä valitaan kokoonpano, jolla käyttö aloitetaan. Myöhemmin moduuleja voidaan tarpeen mukaan ottaa käyttöön. Näin ollen järjestelmä on joustava ja se soveltuukin hyvin dynaamisille ja kasvaville yrityksille. Käyttöönottokustannukset riippuvat siitä, kuinka laajassa käytössä järjestelmä on sekä kuinka paljon järjestelmää räätälöidään vastaamaan asiakkaan liiketoiminnan tarpeita. Microsoftin tuotteena järjestelmä on helppo integroida muiden Microsoftin sovellusten kanssa. (Microsoft 2014)

## 2.2 ERP-järjestelmän toimitusprojekti

ERP-järjestelmän toimitusprojektit ovat haastavia kokonaisuuksia. Pelkästään oikean teknologian valinta on monivaiheinen projekti, jossa edellytetään sekä liiketoiminnan nykytarpeiden tuntemista, tulevaisuuden strategian tuntemusta sekä tuntemusta erilaisista ERP-järjestelmistä. Hankkeet ovat kalliita, joten yritysten tulee panostaa projektin jokaiseen vaiheeseen, jotta järjestelmästä saataisiin mahdollisimman paljon hyötyä. Mikäli yrityksissä ei panosteta riittävästi eri vaihei-

siin, tulee järjestelmä kuluttamaan organisaation resursseja lisäämisen sijaan. Teknologiaa ja toimittajaa valitessaan monet yritykset tekevät yhteistyötä ulkopuolisten konsulttien kanssa, jotka yhdessä yrityksen kanssa löytävät yritykselle parhaan ratkaisun. (Monk & Wagner 2007, 33.)

ERP-järjestelmien toimitusprojekteissa yleisimmin tilaaja tilaa järjestelmän toimittajalta, joka joko markkinoi itse valmistamaansa tai kolmannen osapuolen valmistamaa järjestelmää. Projektin aluksi käydään läpi asiakkaan tarpeet ja vaatimukset ja niitä verrataan valitun teknologian toiminnallisuuteen. Tätä esisuunnittelun vaihetta on luonnollisesti aloitettu jo ennen teknologian valintaa. Mikäli järjestelmä ei pysty vastaamaan asiakkaan tarpeeseen, on joko asiakkaan muutettava omia prosessejaan tai järjestelmän ohjelmakoodiin on tehtävä muutos. Suuremmissa hankkeissa ohjelmakoodiin tehdään lähes poikkeuksetta muutoksia. Ohjelmakoodin muutokset mahdollistavat olemassa olevien prosessien käytön ja siten vähentävät prosessien muuttamisesta johtuvia kustannuksia yrityksissä. Vastavasti muutokset saattavat aiheuttaa järjestelmän epävakautta sekä ylläpidon vaatimien resurssien kasvamista. Joka tapauksessa muutokset kasvattavat merkittävästi testauksen tarvetta. (Wailgum 2008.)

Järjestelmien käyttöönotto jaetaan usein useampaan vaiheeseen ja on myös tavanomaista, että aluksi käyttöön otetaan vain tietyt moduulit ja myöhemmässä vaiheessa moduulien määrää kasvatetaan liiketoiminnan tarpeiden mukaisesti (Monk & Wagner 2007, 198). Konserneissa on melko tavanomaista, että järjestelmä otetaan vaiheittain käyttöön eri konserniyrityksissä.

Tietojärjestelmäprojektit jakautuvat useen eri vaiheeseen, eikä ERP-järjestelmien toimitusprojektit eroa siinä mielessä muista tietojärjestelmäprojekteista. ERP-järjestelmäprojekteissa yleisimmin käytetyt menetelmät ovat perinteinen vesiputousmenetelmä ja ketterän menetelmän variaatiot. Vesiputousmenetelmässä edetään vaiheittain analysoimalla ensin kaikki tarpeet, joihin seuraavassa vaiheessa suunnitellaan ratkaisut, jotka viimeisessä vaiheessa toteutetaan eli koodataan. Vesiputousmenetelmässä on selkeä eteneminen, mutta sen heikkoutena pidetään muun muassa jäykkyyttä, koska aikaisempiin vaiheisiin palaaminen voi olla hankalaa. Ketterässä menetelmässä puolestaan projekti jaetaan useisiin osa-alueisiin,



jotka käsitellään erikseen. Kunkin osa-alueen osalta lyhyen ajan sisällä analysoidaan samanaikaisesti tarpeita, jotka toteutetaan suunnitellun ratkaisun mukaisesti. Ketterän menetelmän etuna pidetään sitä, että yksittäiset osa-alueet valmistuvat nopeammin ja ne ovat aikaisemmin toimitettavissa asiakkaalle. (Palmquist, Lapham, Miller, Chick & Ozkaya 2013, 1-9.) Valitulla menetelmällä on huomattava vaikutus projektin läpivientiin. ERP-järjestelmän toiminnallisten tarpeiden analysoinnin sekä ratkaisujen suunnittelun ja toteutuksen aikana rakennetaan tekniset ympäristöt ja suoritetaan tarpeelliset ohjelmien asennukset.

### 2.3 ERP-järjestelmän laatutekijät

Kansainvälinen ISO-standardi 25010 käsittelee tietojärjestelmien laatuvaatimuksia ja sen arviointia. Standardin esittelyn mukaan tietojärjestelmien sidosryhmään kuuluvat muun muassa kehittäjät, järjestelmien ostajat, käyttäjät sekä ostajaorganisaation asiakkaat. Tietojärjestelmien avulla on tarkoitus tuottaa arvoa sidosryhmille. Jotta arvoa voidaan tuottaa, tulee tietojärjestelmän laatu varmistaa mitaamalla ja arvioimalla sitä järjestelmän eri vaiheissa. (ISO/IEC 25010 2011.)

Sovelluskehityksessä on monta kriittistä vaihetta. Kaikki eri vaiheet tähtäävät siihen, että lopputulos, eli valmis ohjelmisto on asiakkaan odotusten mukainen. Kuten aiemmin on todettu, tehdään usein ERP-järjestelmiin merkittävää sovelluskehitystä, jotta se pystyisi vastaamaan asiakkaan liiketoiminnan tarpeisiin. Ohjelman tulee teknisesti pystyä suoriutumaan sille kuuluvista tehtävistä. Yleensä asiakkaalla ja toimittajalla onkin hyvä yhteisymmärrys siitä mitä sovelluksen tulee tuottaa. On kuitenkin huomattava, että sovelluksen laatua ei määritä se tuottaako järjestelmä halutun lopputuloksen, vaan asiakas määrittää laadun useamman määrällisen ja laadullisen kriteerin mukaan. Burnsteinin (2002) mukaan tietojärjestelmien laadun kriteerit on listattu seuraavaksi:

- Oikeellisuus
  - o Järjestelmä toimii oikein määrittelyn ja tarkoituksensa mukaisesti
- Luotettavuus
  - o Järjestelmä suoriutuu tarkoituksensa mukaisista tehtävistä määrättyissä tilanteissa ja ajan kuluessa
- Käytettävyys
  - o Järjestelmän opetteluun, käyttämiseen, lähdetietojen valmisteluun ja tuotosten tulkitsemiseen vaadittavat resurssit
- Rikkomattomuus
  - o Järjestelmä kestää tarkoitettuja ja tarkoittamattomia hyökkäyksiä sitä kohtaan
- Siirrettävyys
  - o Järjestelmän siirrettävyys ympäristöstä toiseen
- Ylläpidettävyys
  - o Sovelluskehityksen ja ylläpidon edellyttämät resurssit
- Yhteentoimivuus
  - o Järjestelmäintegraation aikaansaamisen edellyttämät resurssit

(Burnstein 2002. 24; ISO/IEC 25010 2011)

Edellä kuvatusta listauksesta voidaan todeta, että sovelluksen käyttöönottoa ennen tulee suorittaa mittavaa testausta, jotta voidaan varmistaa sovelluksen olevan sopiva tarkoitettuun käyttöön sekä varsinaisen toiminnan että asiakkaan odotusten osalta. Testauksen tarkoituksena on paitsi varmistaa, että järjestelmä toimii määrittelyn mukaisesti ja tuottaa oikeita tuloksia. Lisäksi on varmistettava, että sovelluksen toiminta ja asiakkaan odottama käyttökokemus ovat tasapainossa.

Sovelluksen laatu tuotetaan toimitusprojektin aikana. Projektin alkuvaiheessa tehdään paljon päätöksiä, jolloin tietoa ja asiantuntemusta projektitiimissä on vielä rajallisesti. Nämä päätökset ovat kuitenkin lopputuloksen kannalta tärkeimpiä. Projektin epäonnistuminen johtuu Lyytinen & Hirschheimin (1987, 264-265) mukaan neljästä tekijästä:

1. Epäonnistuminen vastaavuudessa, jolloin valmiin järjestelmän ja määritettyjen tavoitteiden välillä ei ole sopivuutta
2. Epäonnistuminen prosessissa, jolloin projekti ei valmistu aikataulussa ja/tai budjetissa
3. Epäonnistuminen vuorovaikutuksessa, jolloin käyttäjien asenteet järjestelmää kohtaan ovat negatiiviset
4. Epäonnistuminen odotuksissa, jolloin tietojärjestelmä ei vastaa käyttäjien tai sidosryhmien odotuksia

(Lyytinen & Hirschheim 1987, 264-265)

Vastaavuus voidaan todeta vertaamalla valmista järjestelmää asiakkaan ja toimittajan tuottamiin vaatimusmäärittelyihin. Käytännössä vaatimus voi kuitenkin olla epäselvä ja kuvattu ainoastaan karkealla tasolla. Myöhemmissä vaiheissa saataan todeta kahden tai useamman vaatimuksen olevan ristiriidassa. Edellä mainituista syistä johtuen vastaavuuden toteaminen voi olla työlästä. (Lyytinen & Hirschheim 1987, 265.)

Aikataulun tai budjetin pettäminen on usein sekä toimittajalle että asiakkaalle näkyvä epäonnistuminen ja sen vuoksi epäonnistumista ei juurikaan voida kiistää. Sen syistä voidaan silti keskustella ja olla eri mieltä. Usein projektin onnistumista arvioidaan sen mukaan, onko järjestelmä pystytty toimittamaan budjetissa ja aikataulussa. Vuorovaikutus on toimitusprojektin aikana kriittisessä osassa ja siinä epäonnistuminen voi johtaa aikataulun ja budjetin pettämiseen. Käyttäjien negatiiviset asenteet järjestelmää kohtaan aiheuttavat järjestelmän käyttöönottavassa organisaatiossa paljon kustannuksia ja järjestelmälle asetettuja odotuksia ei sen vuoksi välttämättä saavuteta.

Odotuksissa epäonnistumisen toteaminen on hankalaa, koska käyttäjän odotukset voivat olla epärealistiset tai odotuksia ei ole kuvattu eri syistä johtuen tai ne eivät ilmene vaatimusmäärittelyistä. Käyttäjien odotukset voivat perustua aikaisempien järjestelmien ratkaisuihin ja arkkitehtuuriin. Vaatimusmäärittelyvaiheessa voi olla myös hankalaa tietää kuinka järjestelmän käyttöönotto tulee vaikuttamaan organisaation toimintaan ja asiakas saattaa sivuuttaa vaatimusmäärittelyssä omasta mielestään epäolennaiset itsestäänselvytykset, joiden olemassaolosta toimittaja ei ole

sen vuoksi lainkaan tietoinen. Odotuksissa epäonnistuminen voidaan havaita ja kokea eri tavoin. On kuitenkin syytä huomata, että odotuksissa epäonnistuminen on yhden tai useamman sidosryhmän kokemus järjestelmästä. (Lyytinen & Hirschheim 1987, 264.)

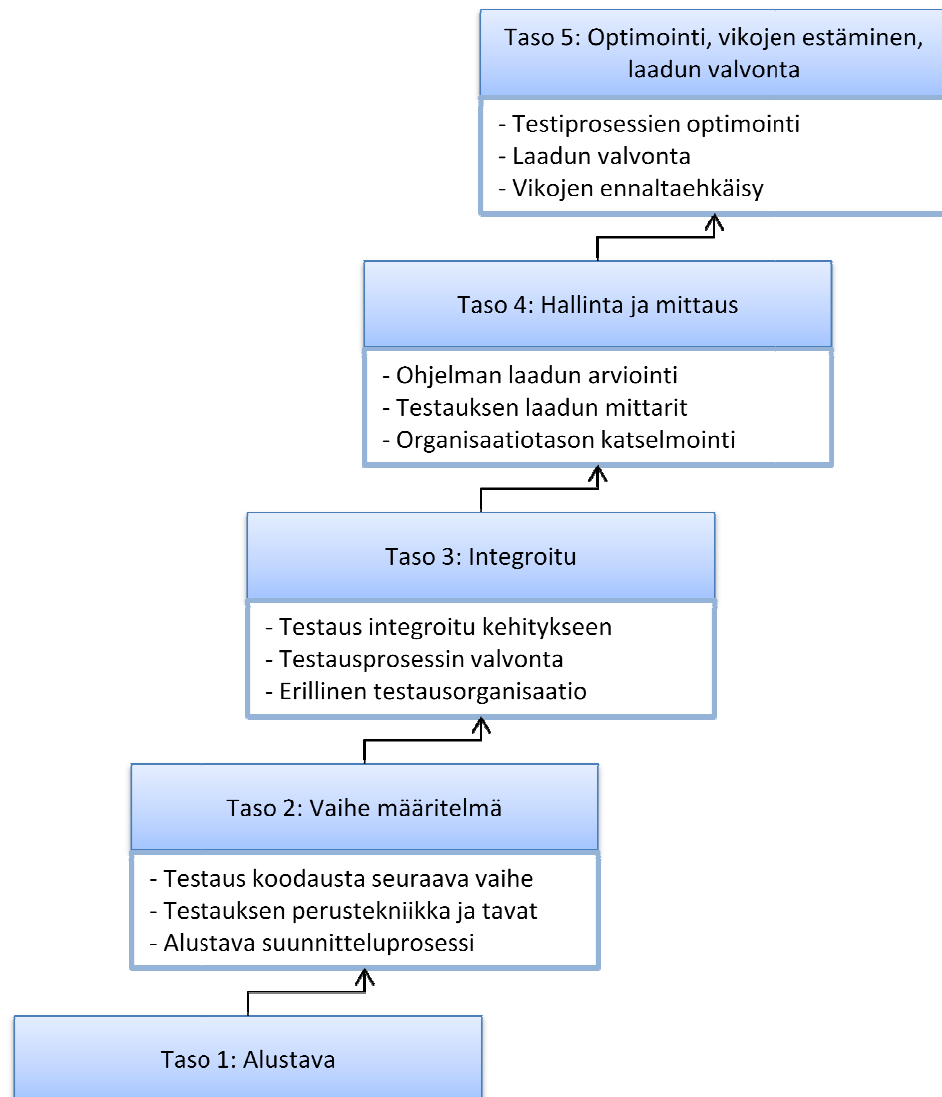
Järjestelmän huolellisella kehityksen aikaisella testauksella sekä asiakkaan suorittamalla hyväksymistestauksella pyritään varmistamaan valmiin järjestelmän laatu. Seuraavassa luvussa käydään läpi testausta sovelluskehityksessä.

## 2.4 Testaus

Edellä on kuvattu toiminnanohjausjärjestelmien laajuus ja kriittisyys yrityksille sekä järjestelmän toimitusprojektin vaiheet. Lisäksi tietojärjestelmien laatutekijät sekä järjestelmähankkeiden mahdollisia epäonnistumisen syitä on käsitelty. ERP-järjestelmän testaus on avainasemassa, jotta epäonnistumisen riski saataisiin minimoitua ja ostajalle voitaisiin toimittaa laadukas ja käyttökelpoinen järjestelmä.

TMMi Foundation on voittoa tavoittelematon organisaatio, joka pyrkii parantamaan tietojärjestelmien testausprosesseja ja –käytäntöjä. TMMi Foundation ylläpitää Test maturity model (TMM) –nimistä viisitasoista portaittaista mallia testausprosessin kehittämiseksi (ISTQB:n testaussanasto 2007). Kuviossa 5 on kuvattu viisitason malli, jota pitkin organisaatio voi edetä ja tavoitella seuraavaa tasoa. Alimmalla tasolla testaus on kaotoinen prosessi, jossa testaus ja debuggaus eivät eroa toisistaan ja toimintaa kuvaavat määritykset puuttuvat. Testauksen tavoitteena on osoittaa, että sovellus toimii. Toiseksi alimmalla tasolla testaus suunnitelmalla ei yleensä tehdä ennen koodin valmistumista, mutta varsinkin testaus on kuitenkin suunnitelmallista. Testauksen tavoitteena on osoittaa, että sovellus toimii määrittelyn mukaisesti. Keskimmaisella tasolla testaus on integroitu sovelluskehityksen ajalle ja testauksen suunnittelu aloitetaan jo vaatimusmäärittelyvaiheessa ja testaus suoritetaan vaatimusmäärittelyjä vasten. Testaus on eriytetty organisaatiossa ja testausta pidetään omana ammattialana. Testauksen etenemistä seurataan sille soveltuvien työkalujen avulla. Toiseksi ylimmällä tasolla ohjelman laatua arvioidaan testauksen aikana katselmointien avulla. Testauksessa huomioidaan myös sovelluksen luotettavuus, käytettävyys sekä ylläpidettävyys ja kaikki testi-

tapaukset tallennetaan tietokantaan uudelleenkäyttöä ja regressiotestausta varten. Ylimmällä tasolla testauksen prosessit ovat jo hallinnassa, koska yritys on jo saavuttanut tason 4. Näin ollen testauksessa keskitytään prosessien optimointiin, laadunvalvontaan sekä vikojen ennaltaehkäisyyn. (Burnstein 2002, 10-16)



KUVIO 5. TMM:n mukainen viisiportainen testauksen tason määrittely (Burnstein 2002, 13)

#### 2.4.1 Testaus osana laadunvarmistusta

Testauksen tarkoitus on varmistaa sovelluksen laatu ja löytää sovelluksessa olevia vikoja. Sovelluskehityksen aikana vikoja voi aiheutua eri vaiheissa, vika saattaa johtua määrittelyvirheistä, suunnitteluvirheistä, huolimattomasta tai virheellisestä koodauksesta tai testauksen virheestä. Lisäksi eri vaiheissa voi syntyä väärinkäsityksiä, jotka aiheuttavat vian järjestelmässä. (Burnstein 2002, 45-51.)

Testauksen yhteydessä käytetään sanoja validointi ja verifiointi kuvaamaan laadun varmistuksen osa-alueita. Edellä kuvattujen kaltaisten vikojen löytäminen liittyy ohjelman validointiin. Vastaavasti verifiointilla varmistetaan, että ohjelma vastaa sitä miksi se on tarkoitettu, eli sen oikeellisuus, luotettavuus ja käytettävyys sekä muut laadun kriteerit vastaavat tilaajan odottamaa laatua. Taloudellisesta näkökulmasta katsottuna ei ole tarkoitus löytää kaikkia mahdollisia vikoja, koska sen toteuttaminen ei olisi liiketaloudellisesti kannattavaa. Testauksen tarkoitus on varmistaa, että ohjelma vastaa riittävällä tasolla asiakkaan odottamaa laatua. (Burnstein 2002, 6-8.)

Mikäli käyttöönotettavaan ERP-järjestelmään ei ole lainkaan tehty muutoksia ohjelmakoodiin, ei varsinaista järjestelmän testausta tarvitse suorittaa, koska voidaan olettaa järjestelmän valmistajan tehneen jo kattavat testaukset. Tällöin testaus rajoittuu siihen, että testataan järjestelmän prosessien ja yrityksen liiketoimintaprosessien yhteensopivuus sekä varmistetaan siitä, ettei järjestelmässä ole järjestelmän valmistajan aiheuttamia virheitä. Usein kuitenkin ERP-järjestelmiin on tehty asiakaskohtaisia sovituksia asiakkaan vaatimusten ja tarpeiden perusteella tai järjestelmässä on liittymäraja- ja muihin järjestelmiin. Asiakaskohtaiset sovitukset ja liittymät edellyttävät järjestelmän huolellista testausta, jotta voidaan varmistua niiden toimivan oikein ja vastaavan yrityksen tarpeisiin. Käyttöönotettavaan ERP-järjestelmiin konvertoidaan lähes poikkeuksetta ainakin jossakin määrin aikaisemmista järjestelmistä historiatietoja ja käyttöönottohetken avoimia tapahtumia, kuten laskukanta ja varastosaldot. Konvertoinnilla tarkoitetaan vanhassa järjestelmässä olevien tapahtumien ja saldojen siirtämistä soveltuvin osin uuteen järjestelmään. Myös konversioiden testaus on tärkeää. Leggen mukaan testauksen tarkoitus on varmistaa, että liiketoimintaprosesseista suoriudutaan onnistuneesti,

kun järjestelmä otetaan tuotantokäyttöön, joten tämä laadunvarmistus on kattavampaa kuin testaus, jossa varmistetaan järjestelmän toimivan määrittelyn mukaisesti. (Legge 2011, 3-4.)

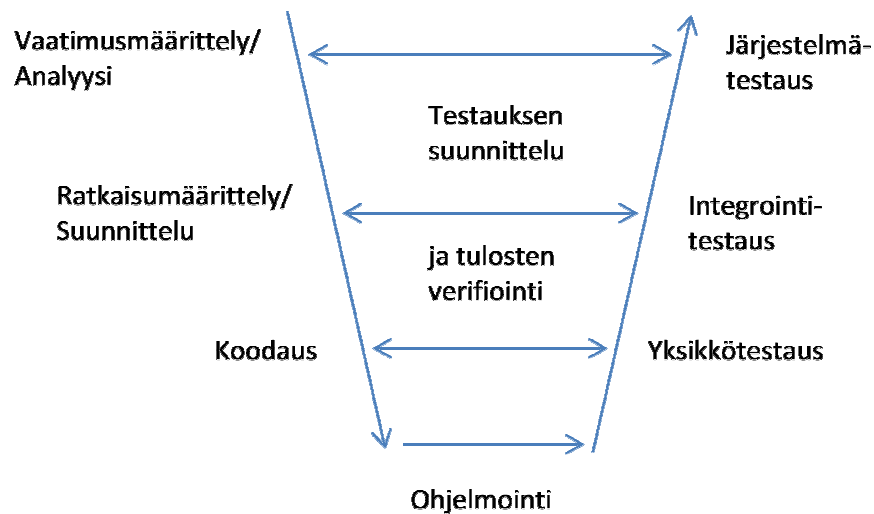
Tehokas testaus edellyttää, että organisaatiossa on testaukseen erikoistuneita henkilöitä, joiden päätoimi on testaus. Kappaleessa 2.5 on kuvattu tarkemmin testausorganisaation tarpeita. Testaaja ei itse ole kehittäjä, vaikka kehittäjän taustasta onkin paljon hyötyä testauksessa. Kehittäjä ei voi testata laadukkaasti itse kirjoittamaansa koodia, koska hänellä voi olla taipumus testata sopivalla ja oikealla aineistolla. Järjestelmien testauksessa tulee testata myös virheellisellä aineistolla ja pyrkiä jopa synnyttämään tilanteita, joissa järjestelmä saadaan toimimaan määrittelyn vastaisesti. Tätä kutsutaan negatiiviseksi testaukseksi. Ohjelmakoodissa voi olla virheitä väärinymmärryksistä johtuen, jotka jäävät helposti huomaamatta, mikäli kehittäjä testaa itse komponenttia. Käytettävyyden testaus tulee olla aina jonkun muun kuin kehittäjän tai suunnittelijan testaamaa. Testaajien tulee myös perehtyä testaukseen ja saada koulutusta testauksesta. (Burnstein 2002, 237.)

Myersin (2004, 16-17) testauksen periaatteiden mukaan testauksen suorittajien, eli testaajien, tulee olla riippumattomia kehittäjistä, jopa organisaation ulkopuolisia. Burnsteinin mukaan testaajat voivat olla samassa organisaatiossa erillisellä osastolla, mutta heidän tulee suhtautua testauksen kohteisiin objektiivisesti. Kuitenkaan tarkoituksena ei ole luoda vastakkainasettelua testaajien ja kehittäjien kesken, vaan tarkoitus on, että he yhdessä pystyvät tuottamaan asiakkaalle korkealaatuisempia sovelluksia. (Burnstein 2002, 30-31.)

#### 2.4.2 Testauksen vaiheet

Ohjelmistokehityksen aikainen testaus jaetaan eri vaiheisiin. ERP-järjestelmä koostuu komponenteista, jotka integroidaan yhdeksi kokonaisuudeksi. Aluksi testataan yksittäisiä komponentteja, joissa varmistetaan koodin oikea toiminta. Seuraavaksi testataan komponenttien integraatiota ja samalla testataan järjestelmän toiminnan vastaavan suunniteltua ratkaisua. Ohjelmistokehityksen viimeisessä vaiheessa testataan koko järjestelmää ja varmistetaan järjestelmän sopivan käyttötarkoitukseensa. Tämän jälkeen asiakas suorittaa hyväksymistestin. Jäljempänä

käyn tarkemmalla tasolla läpi testauksen eri vaiheet. Kuviossa 6 on havainnollistettu projektin eri vaiheiden ja testausvaiheiden yhteenkuuluvuus. Vaatimusmäärittelyä tehtäessä tulisi jo suunnitella järjestelmätestausta ja vastaavasti järjestelmätestausta tehtäessä tuloksia tulee verrata vaatimusmäärittelyyn. Vastaava yhteys on ratkaisumäärittelyllä ja integrointitestauksella sekä koodauksella ja yksikkötestauksella.



KUVIO 6. Projektivaiheen ja testausvaiheiden yhteenkuuluvuus. (Burnstein 2002, 15)

### Yksikkötestaus

Yksikkötestaus on testauksen ensimmäinen vaihe, jossa testataan yksittäistä komponenttia. Kokonaisuus syntyy komponenteista, joten jokaisen testaus on tärkeää. Mikäli jokaista komponenttia ei ole testattu, voidaan myöhemmässä vaiheessa todeta, ettei järjestelmä toimi halutulla tavalla, jolloin vian aiheuttajan löytäminen on hankalampaa järjestelmän koon vuoksi ja mahdollisten vian aiheuttajien suuren määrän vuoksi.

Yksikkötestauksessa haasteena voi olla se, että testauksen toteuttamiseksi saataan joutua tuottamaan paljon testimateriaalia tai kehittämään ohjelma, joka muo-



dostaa haluttua aineistoa. Yksikkötestauksessa tulisi jo testata erilaisella aineistolla ja myös virheellisellä aineistolla, jotta voidaan varmistaa komponentin toiminta. (Burnstein 2002, 133.)

### Integroititestausta

Integroititestausta on yksikkötestausta seuraava vaihe, jossa testataan komponenttikokonaisuuksia. Tarkoitus on testata komponenttien välinen toiminta ja varmistua siitä, että edellisen komponentin tuotos saa aikaan halutun lopputuloksen prosessin seuraavassa komponentissa. Mikäli yksikkötestausta on tehty huolellisesti, voidaan pääasiassa keskittyä komponenttien väliseen toimintaan. On kuitenkin huomioitava, että huolellisesti testauksesta huolimatta, voidaan testauksen myöhemmässä vaiheessa löytää vikoja, jotka on jäänyt edellisessä vaiheessa huomaamatta. (Burnstein 2002, 152.)

### Järjestelmätestausta

Järjestelmätestausta on testauksen vaihe, jossa testataan koko järjestelmää kokonaisuutena. Järjestelmätestausta voidaan aloittaa, kun kaikki komponentit ovat valmiita ja ne ovat liitetty yhteen, eli kokonainen järjestelmä on rakennettu. Järjestelmätestaustassa kiinnitetään erityistä huomiota prosessien käytettävyyteen ja suorituskäyttöön. Kuitenkin on huomioitava, että mitä aikaisemmassa vaiheessa suorituskäyttöön ja käytettävyyteen liittyvät viat tai puutteet havaitaan, sitä helpompaa ja edullisempää niiden korjaaminen on. Järjestelmälle tehdään myös stressitestejä, joiden tarkoituksena on saada järjestelmä tilaan, jossa se ei kykene käsittelemään pyyntöjä. Testauksella varmistetaan, että järjestelmä pystyy suoriutumaan stressitilanteista aiheuttamatta virheitä. Muita järjestelmätestaustan testattavia kokonaisuuksia on palautumistesti, jolla varmistetaan, että järjestelmä pystyy palautumaan itse virhetilanteesta. Palautumistestit ovat erityisen tärkeitä esimerkiksi sovelluksissa, joissa asiakas maksaa tuotteista tai palveluista. (Burnstein 2002, 163-175.)

Myöhäisessä vaiheessa havaittu suorituskäyttöongelma saattaa aiheuttaa mittavia muutoksia, koska tarvittava muutos saattaa edellyttää muutoksia myös toisiin komponentteihin. Aina muutosten jälkeen tulee myös tehdä regressiotestausta.

Regressiotestauksen tavoitteet ja tarkoitus on kuvattu myöhemmin. (Burnstein 2002, 176.)

### Hyväksymistestaus

Hyväksymistestaus on testauksen viimeinen vaihe, jossa asiakas testaa järjestelmän ennen käyttöönottoa. Aikaisemmat testauksen vaiheet ovat toimittajan suorittamia. Hyväksymistestauksen tarkoitus on nimensä mukaisesti antaa asiakkaalle varmuus siitä, että järjestelmä vastaa odotettua ja se voidaan ottaa tuotantokäyttöön. Myersin (2004, 144) mukaan hyväksymistestauksessa verrataan järjestelmää alkuperäisiin vaatimuksiin sekä loppukäyttäjän nykyisiin tarpeisiin.

Hyväksymistestauksessa ongelmana on monesti se, että järjestelmä on vielä uusi asiakkaalle eikä asiakkaalla ole riittävää osaamista kattavaan hyväksymistestaukseen. Asiakkaan resurssit voivat olla rajalliset. Testauksessa tulee varmistua paitsi järjestelmän toiminnasta tilanteesta, jossa lähdetiedot ovat virheettömät, myös tilanteesta jossa lähdetiedoissa on virheitä tai järjestelmää kuormitetaan muuten poikkeuksellisella tavalla. Virheellisistä lähdetiedoista tai kuormituksesta huolimatta järjestelmän tulee pystyä vähintään palautumaan itsestään normaalitilanteeseen. Toiminnanohjausjärjestelmän hyväksymistestauksessa tärkeässä roolissa on myös käyttöoikeustestaus, jossa varmistetaan eri käyttäjäroolien käyttöoikeuksien oikeellisuus.

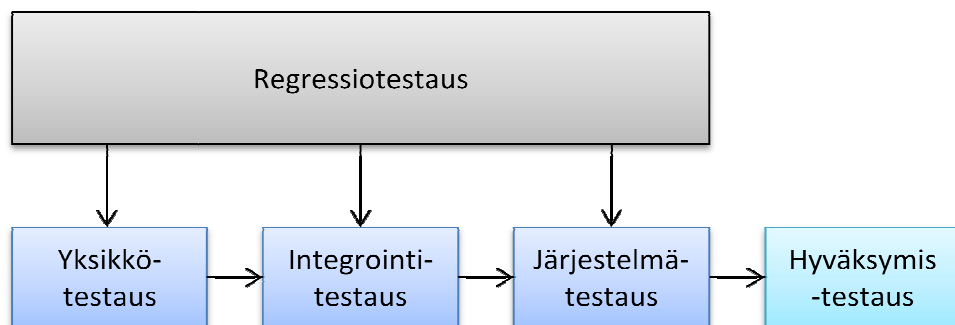
Hyväksymistestaus toteutetaan käyttöönoton kynnyksellä, mutta se tulee suunnitella jo hyvissä ajoin projektin aikaisemmissa vaiheissa. Aikaisempien vaiheiden testauksessa tulee olla jo tietoinen hyväksymistestauksessa vaaditusta tasosta, jota olisi vaikea määritellä ilman hyväksymistestaussuunnitelmaa. Hyväksymistestaus on aikaavievä projektin osa. Koska se on kuitenkin projektin yksi tärkeimmistä asioista, tulee sen etenemistä ja kattavuutta seurata huolellisesti. (Dunkinson & Birch 2013, 177.)

Tässä opinnäytetyössä asiakkaan mahdollisesti suorittama prosessitestaus ennen varsinaista hyväksymistestausta käsitetään osaksi hyväksymistestausta. Prosessitestauksessa asiakkaan loppukäyttäjät testaavat järjestelmän prosesseja, vaikka koko järjestelmä ei olisikaan vielä toimitettu tai järjestelmässä olisi vielä puutteita.

Testaus auttaa loppukäyttäjiä oppimaan järjestelmän käyttö ja siinä voidaan löytää virheitä, jotka muussa tapauksessa olisi löydetty vasta hyväksymistestauksessa.

### Regressiotestaus

Regressiotestaus on toistuva testaus järjestelmälle, kun järjestelmään on tehty muutos. Regressiotestaus ei ole siis testauksen vaihe vaan erillinen ohjelmistokehityksen rinnalla tapahtuva prosessi. Integroidussa ERP-järjestelmässä yhteen moduuliin tehty muutos voi aiheuttaa virheitä toisissa moduuleissa. Muutoksen jälkeen ei siis riitä pelkästään muutetun komponentin testaus, vaan koko järjestelmä tulee testata uudelleen. Regressiotestauksessa voidaan käyttää aikaisemmin tehtyjä testitapauksia ja myös testata automaattisilla testausmenetelmillä. Regressiotestaus on erityisen tärkeää, kun järjestelmä toimitetaan useammassa julkaisussa, koska tilaaja odottaa uuden toiminnallisuuden toimivan, mutta myös aiemmin testatun toimivan edelleen kuvatulla tavalla. (Burnstein 2002, 176; ISTQB:n testaussanasto 2007, 33.) Kuviossa 7 on kuvattu regressiotestaus erillisinä rinnakkaisina testausprosessina projektin testausvaiheiden aikana.



KUVIO 7. Testauksen vaiheet projektissa. (Naik & Tripathy 2008, 17.)

## Debuggaus

Debuggaus ei ole varsinaista testausta, vaan debuggauksella tarkoitetaan virheen paikantamista sen jälkeen kun testauksessa on havaittu, ettei ohjelma toimi odotetulla tavalla tai se ei tuota odotettua lopputulosta. Debuggauksen tekee normaalisti kehittäjä ja sen tarkoituksena on paikantaa virhe koodissa, korjata virhe ja testata uudelleen. Debuggausta ei siis tule sekoittaa esimerkiksi yksikkötestaukseen.

(Burnstein 2002, 7-8; ISTQB testaussanasto 2007)

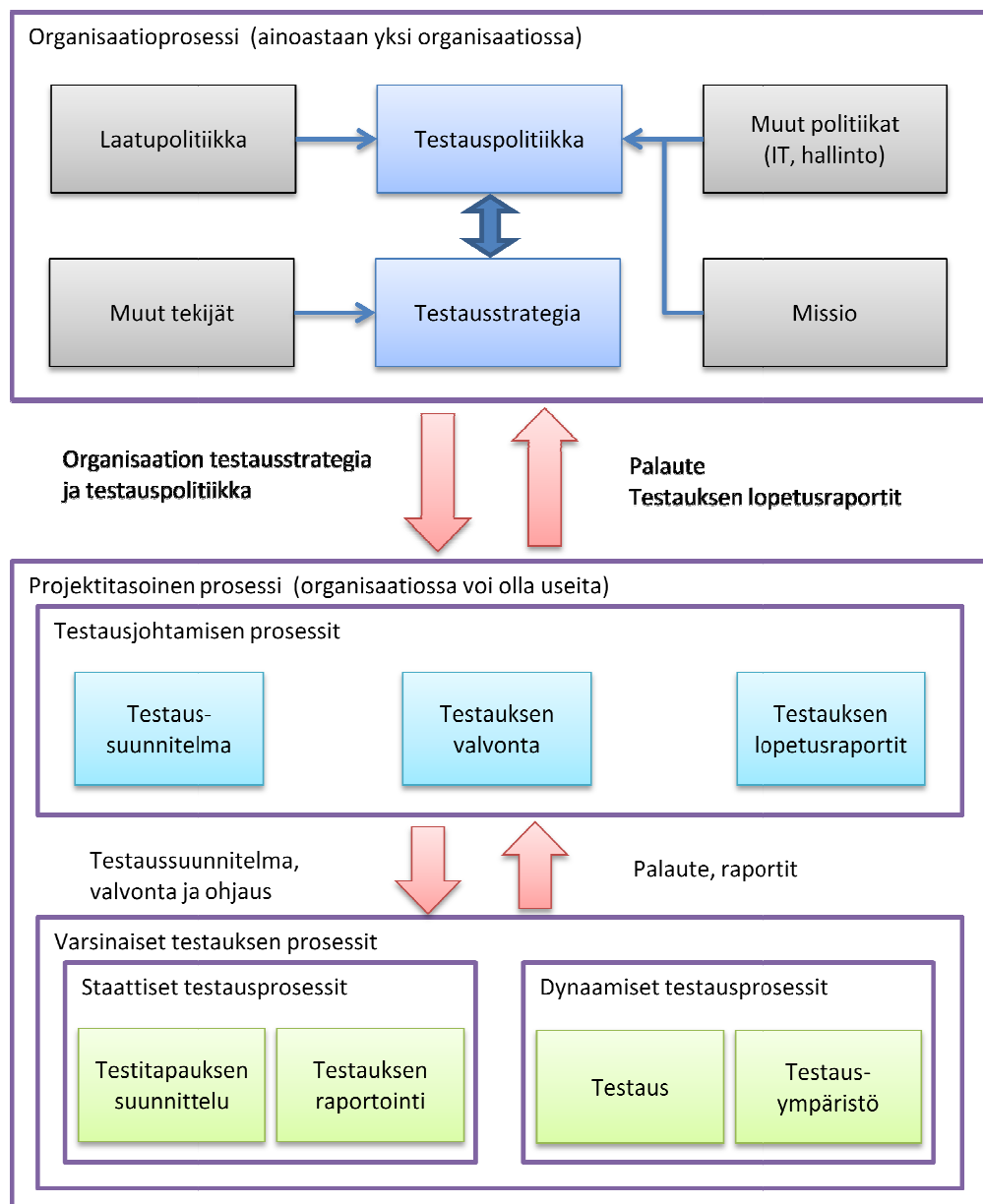
### 2.5 Testauksen organisoiminen

Testaus on yksi järjestelmäkehityksen ydintoiminnoista. Kasurinen (2011, 13) kirjoittaa väitöskirjassaan, että usein testaus tehdään kuitenkin vajain resurssein. On huomioitava, että testaus koituu kalliiksi ja tehottomaksi ilman etukäteen tehtyä suunnitelmaa. Kasurisen mukana organisaatioiden tulisi panostaa testaukseen, jotta investointi olisi mahdollisimman tuottava.

Testaukseen kuuluu osina testauspolitiikka ja -strategia organisatorisella tasolla, jotka viitoittavat millä tasolla organisaatiossa yleisesti testataan. Projektitasolla tehdään testaussuunnitelma, joka perustuu testauspolitiikassa ja -strategiassa asetettuihin tavoitteisiin ja muihin reunaehtoihin. Testaussuunnitelmaan kuuluu testauksen valvonta ja testauksen päättäminen. Projektissa vastuu testauksesta kuuluu testipäällikölle ja hän vastaa siitä, että testaajat tekevät testisuunnitelman mukaisesti varsinaisen testauksen, johon kuuluu testiraportointi, testitapausten suunnittelu ja testauksen suorittaminen. Projektin testipäällikkö tekee projektissa testausraportin. (Kasurinen 2011, 13, 23-24.)

Kuviossa 8 on kuvattu organisaation testausoiminnot ja niiden riippuvuus toisiinsa ja organisaation muuhun politiikkaan nähden. Testausoiminnot jakautuvat kahteen osaan, joista ylempänä on organisaatiotason testauspolitiikka ja -strategia, joihin vaikuttavat organisaation missio sekä muut politiikat. Testauspolitiikka ja -strategia vaikuttavat projektitasoiseen testausjohtamiseen ja testauksen suorittamiseen. Myöhemmin on kuvattu tarkemmalla tasolla kunkin osa-alueen sisältö ja merkitys. Organisaatiotason testauspolitiikka ja -strategia vaikuttavat ylhäältä alaspäin projekteihin ja yksittäisiin päätöksiin projekteissa ja vastaavasti alhaalta

ylöspäin projektitasolta organisaatiotasolle siirtyy palautetta, jotta testauspolitiikka ja –strategiaa voidaan tarvittaessa päivittää. Kunkin projektin testausta johdetaan projektille sopivalla tavalla, joten organisaatiossa on useita testauksen projektitasoisia prosesseja. Testausjohtaminen vaikuttaa varsinaiseen testaukseen ja vastaavasti suorittavan tason testauksesta siirtyy palautetta projektin testauksen johtoon, jotta esimerkiksi testaussuunnitelmaa voidaan tarvittaessa päivittää.



KUVIO 8. Testausprosessin tasot ja rakenne. (ISO/IEC 29119 2010.)

Testauspolitiikan tulisi määrittää seuraavat testaukseen vaikuttavat asiat:

- testauksen tarkoitus organisaatiossa
  - organisaatiossa sovellettava testausprosessi yleisellä tasolla
  - testausorganisaation rakenne
  - testausorganisaation jäseniltä edellytettävä koulutustaso
  - eettiset säännöt
  - noudatettavat standardit
  - testauksen mittaamiseen liittyvät asiat
- (Kasurinen 2011, 25-26)

Testausstrategissa tulisi määrittää seuraavat asiat hieman politiikkaa tarkemmalla tasolla:

- testauksen riskienhallinnan tekijät
  - testauksen aloittamisen ja päättämisen määrittävät tekijät
  - testausorganisaation riippumattomuus muista organisaation osista
  - testauksen dokumentointi
  - testauksen vaiheet
  - testautyyypit
  - testaustekniikka
  - testitapausten valinta ja priorisointi
  - testiympäristöt
  - automaattitestauksen käyttö ja sen työkalut
  - regressiotestauksen käyttö
  - testauksen päättämisen kriteerit
  - virheiden hallintastrategian
  - mahdolliset poikkeukset testauspolitiikkaan nähden
- (Kasurinen 2011, 25-26)

Projektissa testauspolitiikka ja testausstrategia ovat testaussuunnitelman lähtökoh-  
ta. Testaussuunnitelman ja testauksen lopetusraporttien sisältö on kuvattu myö-  
hemmin.

### 2.5.1 Testausorganisaatio

Testausorganisaation rakenne riippuu siitä kuinka hyvään ja kattavaan testaukseen organisaatiossa pyritään ja siihen vaikuttavat edellä kuvattu testauspolitiikka ja –strategia. Jotta testauksen taso voidaan varmistaa, tulisi organisaatiossa testaajat sijoittaa omaan yksikkönsä erikseen yksiköistä, jotka tuottavat sovelluksia. Näin pystytään varmistamaan testaajien objektiivisuus ja siten nostamaan sovellusten laatua. Testausorganisaatiossa tulee olla riittävät taidot, koska se on vastuussa seuraavaksi listatuista asioista:

1. testipolitiikan ylläpito ja soveltaminen käytäntöön
2. testausstandardien ylläpito ja soveltaminen käytäntöön
3. osallistuminen vaatimus-, ratkaisumäärittelyiden sekä koodin katselmointiin
4. testaussuunnittelu
5. testitapausten teko
6. testauksen mittaaminen
7. testauksen valvonta (tehtävät, aikataulu, kustannukset)
8. vikojen seuranta ja arkistointi
9. testityökalujen ja tarvikkeiden hankinta
10. uusien testausmenetelmien ja -työkalujen tunnistaminen ja soveltaminen
11. uusien testaajien koulutus ja mentorointi
12. testiraportointi

(Burnstein 2002, 235-236)

Testaajat suorittavat varsinaisen testauksen. Testaajia voi edesauttaa tehtävissään kehittäjän tausta, mutta testaaja ei silti ole kehittäjä, eikä hän näin ollen korjaa koodia. Testaajan tarkoitus on tuottaa lisäarvoa sovellukselle paremman laadun ja asiakastyytyväisyyden muodossa ja organisaatiossa on huomioitava testaajan tärkeä rooli edellä mainitussa. Jotta testaaja voi tuottaa lisäarvoa ja varmistaa sovellusten laadun, tulee testaajat kouluttaa tehtävänsä. (Burnstein 2002, 236-237.)

Testausorganisaatiolla on vastuu laadun tuottamisesta, joten sen tehtävä ei ole ainoastaan varmistaa sovelluksen toimivan kuvatun määrittelyn mukaisesti, mutta

myös varmistua siitä, että sovellus tai järjestelmä toimii sillä tavalla kuin sen on tarkoitus toimia. (Kasurinen 2011, 18.)

Testausorganisaatiossa voi olla useita tiimejä, jotka työskentelevät eri projekteissa. Jotta testaustehtävä tulee suoritettua mahdollisimman tehokkaasti, tulee organisaatiossa olla testaajien lisäksi testauspäällikkö tai – johtaja, jolla on vastuu johtamisesta sekä vastuu testauspolitiikasta ja –strategiasta. Pienemmissä organisaatioissa testauspäällikkö voi tehdä myös varsinaista testausta. Lisäksi organisaatiossa voi olla johtava testaaja, jolla on tiimissä tai projektissa vastuu esimerkiksi testaussuunnitelmasta, testauksen valvonnasta ja raportoinnista. Projekteissa testaajien määrä suhteessa kehittäjiin vaihtelee toteutettavasta sovelluksesta riippuen, mutta yleisiä suhdelukuja on 1:2, 2:3 tai 1:4. (Burnstein 2002, 245-247.)

### 2.5.2 Testauksen suunnittelu

ERP-projektissa projektin johto on vastuussa testaussuunnitelmasta. Käytännössä testaussuunnitelman voi tehdä testauspäällikkö tai johtava testaaja. Testausuunnitelma tehdään organisaation testauspolitiikkaan ja -strategiaan perustuen. On huomattava, että testaussuunnitelman lisäksi testauksessa suunnitellaan testit-paukset, joiden pohjalta varsinainen testaus suoritetaan. Kasurisen mukaan (2011, 23, 26) testaussuunnitelman tulee sisältää seuraavat tiedot:

1. projekti, prosessi ja tuote, johon testaussuunnitelma perustuu
2. testauksen kohteet, jossa kuvataan kohteen käyttötarkoitus sekä koskeeko testaus koko sovellusta tai integraatioita
3. testauksen rajaus, karkealla tasolla mitä sisällytetään tai jätetään testauksen ulkopuolelle
4. testausstrategia, kuten testauksen vaiheet, testityypit, testausympäristö, testityökalut sekä testausprosessin muut tuotokset
5. arviot testaustehtävistä ja osituksesta, etapeista, budjetista ja kustannuksista
6. projektissa käytettävät henkilöresurssit ja vastuut
7. testauksen aikataulu
8. projektin riskit suhteessa testaukseen



## 9. mahdolliset poikkeukset testistrategiaan nähden

(Kasurinen 2011, 26-27.)

Testitapauksessa kuvataan yksittäinen testin lähtötilanne, toteutettavat toimenpiteet ja odotettu lopputulos. Myersin (2004, 14-15) testauksen ensimmäisen periaatteen mukaan testitapauksen suunnittelun yhteydessä on jo määritettävä odotettu lopputulos, koska muuten virheellinenkin lopputulos saatetaan virheellisesti hyväksyä.

Burnsteinin (2002, 197-198) mukaan testaussuunnitelman tekeminen tulisi aloittaa jo vaatimusmäärittelyvaiheessa ja projektin myöhemmissä vaiheissa testaussuunnitelmaa ylläpidetään. Suunnitelman välitavoitteiden avulla projektin johto pystyy seuraamaan projektin etenemistä. Testaussuunnitelmassa tulee myös päättää testauksen lopetusehdot, koska virheiden etsiminen siihen asti, että kaikki virheet on löydetty, ei ole taloudellisesti kannattavaa. Testaussuunnitelma voidaan tehdä erikseen yksikkötestaukselle, integrointitestaukselle ja järjestelmätestaukselle tai eri osa-alueet voivat sisältyä testauksen pääsuunnitelmaan. (Burnstein 2002, 197-199.)

### 2.5.3 Testauksen raportointi

Testauksesta tuotetaan testauksen lopetusraportti. Lopetusraportilla tuotetaan organisaatiolle palautetta toteutetusta testauksesta. Lopetusraportin tuottaminen edellyttää myös projektin aikaista raportointia ja se sisältää Kasurisen mukaan (2011, 27) seuraavat asiat:

1. yhteenveto toteutetusta raportoinnista testauksen eri vaiheissa ja eri testityypeistä
2. poikkeamat suunnitellusta testistä
3. testauksen lopetusehtojen toteutuminen ja selvitys mikäli ehdoista jouduttiin poikkeamaan
4. tekijät, jotka estivät testausprosessin riittävällä tarkkuudella
5. projektissa opitut asiat (lessons learned)

6. testaukseen liittyvät mittarit, kuten testitapausten, havaittujen vikojen ja havaintojen määrät
7. uudet ja muuttuneet riskit sekä kuinka niihin on varauduttu
8. testiympäristön tila testauksen päättyessä
9. testauksen tuotokset ja viittaukset niihin
10. tulevissa projekteissa käytettävissä olevat testaustyökalut sekä niiden sijainti ja käytettävyys
11. suositukset toteutetun testauksen pohjalta

(Kasurinen 2011, 27)

Testauksen aikana prosessissa voidaan tuottaa testiloki, joka on testaajan tuottama. Testilokin tarkoitus on mahdollistaa testien käyttämistä tulevaisuudessa testauksessa. Testiloki kertoo mitä testejä on tehty sekä testauksen tulokset ja näin sitä voidaan käyttää hyväksi kun testataan toteuttuja korjauksia koodiin tai tehdään regressiotestausta. Kun testauksessa havaitaan ohjelmiston toimivan yllättävällä tai selittämättömällä tavalla ja se edellyttää lisätutkimuksia, tuotetaan havaintoraportti. Havaintoraportin ja testilokin avulla kehittäjä voi selvittää ongelman. Näin ollen havaintoraportilta tulee käydä ilmi tarvittavat yksityiskohdat, joita ovat muun muassa aika, testaajat, ympäristö, syöte, odotettu tulos, toteutunut tulos, vakavuus, vaikutukset, poikkeamat sekä toistuvuus. Kunkin organisaation tulee silti arvioida mitkä näistä ovat relevantteja, koska niiden tuottaminen sitoo resursseja. (Burnstein 2002, 205, 222-224.)

Kuten on todettu, on testaus hyvin kriittinen osa sovelluskehitystä ja huolellinen testaus edellyttää paitsi ammattitaitoa myös sovittuja toimintatapoja, jotta testaus olisi mahdollisimman tehokasta. Vaikka organisaatiossa onkin testausosaamista, voidaan osaamisen myyminen sellaisenaan mieltää hankalaksi. Seuraavassa luvussa käydään läpi asiantuntijapalveluiden tuotteistamista, mistä voisi olla apua myös testauksen markkinointiin.

### 3 ASIANTUNTIJAPALVELUN TUOTTEISTUS

Seuraavissa luvuissa käydään läpi asiantuntijapalvelun erityispiirteitä ja niiden vaikutusta palvelun tuotteistamiseen. Lisäksi kuvataan asiantuntijaorganisaatioille palvelun tuotteistamisesta koituvat hyödyt.

#### 3.1 Asiantuntijapalvelu

Christian Grönroosin (2010, 76-79) mukaan palvelu on monimutkainen ilmiö. Palvelu-sanaa käytetään laajasti hieman erilaisiin tarkoituksiin. Mikäli fyysisistä tuotetta mukautetaan asiakkaan tarpeita vastaavaksi, muodostuu kokonaisuudesta palvelu, joskin itse kone onkin edelleen fyysinen tuote. Yleisesti ottaen palvelulla on Grönroosin mukaan kolme yleisluonteista peruspiirrettä.

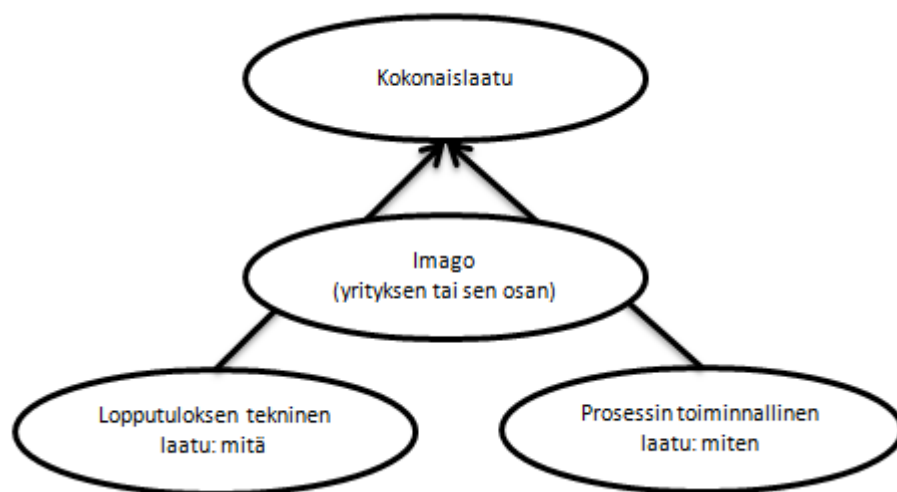
1. Ne ovat prosesseja ja koostuvat toiminnoista
2. Ainakin jossakin määrin ne tuotetaan ja kulutetaan samanaikaisesti
3. Asiakas osallistuu ainakin jossakin määrin palvelun tuottamiseen

Tärkein piirre palvelussa onkin sen prosessiluonne, eli palvelu koostuu useista toiminnoista ja tuottamiseen käytetään resursseja, kuten ihmisiä, tietoa, järjestelmiä, infrastruktuureja ja fyysisiä resursseja usein suorassa vuorovaikutuksessa asiakkaan kanssa. (Grönroos 2010, 76-79.)

Fyysisestä tuotteesta poiketen palveluja ei voida tuottaa varastoon, vaan palvelun tuotanto alkaa aina asiakkaan aloittaessa prosessin. Palvelunmyynti on tuotantoprosessi, jossa asiakkaalle tuotetaan palvelua, jota asiakas kuluttaa ja näin tuotetaan arvoa asiakkaalle. Tuotos voi olla myyjän nimeämä palvelutuote, joka on ennalta määritetty sisältävän tiettyjä kokonaisuuksia tai se on tuotos, jonka sisältöä ei ole ennalta määritetty, vaan se muodostuu tuotantoprosessissa myyjän ja ostajan vuorovaikutuksessa. Myös arvo muodostuu osittain myyjän ja ostajan vuorovaikutuksessa. Ostajan mukanaolo prosessissa sekä ostajana että arvontuottajana on huomattava. Myyjän näkökulmasta tuotos on hyvin merkittävä ja sen vuoksi myyjä keskittyy varsin paljon siihen ja pyrkii siihen, että se vastaa parhaiten markkinoiden odotuksia. Markkinoinnin vuoksi ja tuotantoprosessin tehostamiseksi tuotosta pyritään vakioimaan ja nimeämään. (Grönroos 2010, 48-51.)

Palvelun ostajan näkökulmasta tuotos on tärkeä, mutta palvelua arvioitaessa koko tuotantoprosessi on tärkeä ja joissakin tilanteissa jopa tärkeämpi kuin varsinainen prosessin lopputulos. Palvelun erityispiirteiden vuoksi tuotantoprosessi ei ole rinnastettavissa fyysisen tuotteen tuotantoprosessiin, jota pyritään virtaviivaistamaan kustannusten vähentämiseksi. Fyysisen tuotteen ostajalle ei tuotantoprosessin vaiheilla ole yleensä merkitystä. Palvelun ostaja puolestaan on läsnä palveluprosessin aikana ja siten prosessin vaiheilla on merkitystä ostajalle. Näin ollen palveluntuottajan tulee keskittyä palvelun tuotantoprosessin vaiheisiin ja osatekijöihin vähintään yhtä paljon kuin varsinaisen tuotokseen. Vaiheista tulee tunnistaa ne, jotka ovat muodostamassa lisäarvoa asiakkaalle ja ne, jotka eivät muodosta lisäarvoa. Prosessin aikana myyjän ja ostajan välille muodostuu jonkinlainen suhde. Hyvä myyjän ja ostajan välinen suhde on keskeinen osa uskollisuutta ja uskolliset asiakkaat puolestaan ovat lähes aina kannattavia. (Grönroos 2010, 30.)

Kuviossa 9 on kuvattu, kuinka palvelun laatu voidaan jakaa kahteen osatekijään, lopputuloksen tekniseen laatuun sekä prosessin toiminnalliseen laatuun. Lopputuloksen teknisellä laadulla kuvataan toimitetun palvelun lopputuotosta, eli mitä asiakkaalle on tuotettu. Prosessin toiminnallisuuden laadulla kuvataan puolestaan sitä, miten palvelu on tuotettu asiakkaalle. Nämä yhdessä muodostavat asiakkaan kokeman laadun. Asiakkaan kokemaan laatuun vaikuttaa myös osin muitakin asioita, kuten palvelun myyjän imago. (Grönroos 2010, 103.)



KUVIO 9. Kaksi palvelun laatu-ulottuvuutta (Grönroos 2010, 103)

Asiantuntijapalvelut ovat hyötyä tuottavia vaihdon välineitä, jotka ovat erikoisosaamiseen perustuvia aineettomia toimintoja. Erikoisosaaminen on tekijä, mikä tekee palvelusta asiantuntijapalvelun. Asiakas ei kuitenkaan osta palvelua, vaan palvelun tuottaman hyödyn. Tarkemmin sanottuna asiakas ostaa tuotteista, palveluista, tiedoista ja muista tekijöistä koostuvan tarjooman, joka on asiakkaalle palvelu ja asiakkaan kokema arvo perustuu tähän kokonaisuuteen. Kun palvelun tarkoituksena on tukea asiakkaan toimintoja ja prosesseja, koituu arvo asiakkaalle asiakkaan arvontuotantoprosesseissa, kun ostettua palvelua hyödynnetään. (Lehtinen & Niinimäki 2005, 9; Grönroos 2010, 25.)

Asiantuntijapalvelut ovat moniin muihin palveluihin verrattuna kaikkein aineettomimpia palveluita. Näin ollen asiakkaalle niiden arvon arvioiminen kesken tuotantoprosessin voi olla hyvin hankalaa, puhumattakaan arvon arvioimisesta ostohetkellä. Näin ollen myös asiantuntijapalveluun liittyvät riskit ovat asiakkaan näkökulmasta varsin suuret. Usein palvelun arvioiminen jälkeensä voi myös olla hankalaa. Asiakkaan tulisikin osallistua aktiivisesti tuotantoprosessiin. Lisäksi on huomioitava, että ostajan voi olla palvelun erityisluonteen vuoksi hankala arvioida tuotoksen laatua, joten sen vuoksi tuotantoprosessin aikana rakentuva ostajan ja myyjän välinen luottamussuhde on avainasemassa palvelua arvioitaessa. Palvelutarpeen määrittäminen ostohetkellä on usein asiakkaalle vaikein ja vaativin osa. Sen vuoksi asiakas ei usein kykene tekemään täsmällistä tilausta, koska hän ei myöskään kykene analysoimaan omaa tilannettaan. (Lehtinen & Niinimäki 2005, 11-12.)

Lehtisen (2005, 13-14) mukaan asiantuntijapalvelujen markkinointia ja tuotteistamista ei voida pitää korkeatasoisena. Monilla aloilla ei ole nähty tarvetta markkinointiin tai markkinointia ei ole pidetty sopivana ja sen vuoksi siihen ei ole panostettu. Joillakin aloilla tarvetta markkinointiin ei ole ylipäätään ollut pysyvän ylikysyntätilanteen vuoksi. Hyvän suoritettuna asiantuntijapalvelun on uskottu tuovan asiakkaita ja kasvattavan liiketoimintaa. Markkinoiden kansainvälistyessä ja kilpailun kiristyessä, markkinoinnin tarve lisääntyy. Markkinointia edellytetään myös tilanteissa, jossa asiakaskunnassa ei tunnusteta tarvetta myytävälle asiantuntijapalvelulle. Seuraavassa luvussa kerrotaan palvelun tuotteistamisesta ja sen hyödyistä asiantuntijaorganisaatioille.

### 3.2 Palvelun tuotteistus

Tuotteistus-sanaa käytetään enimmäkseen puhuttaessa palveluliiketoiminnasta. Fyysinen myytävä hyödyke on jo tuote, joten varsinaisesta tuotteistamisesta ei juurikaan puhuta, koska koko tuotekehitysprosessi keskittyy itsessään jo hyödykkeen tuotteistamiseen. Sen sijaan palvelu on prosessi, jossa asiakas on osallisena tuotantoprosessissa. Palvelutuotannossa tuotteistamisen avulla pyritään mahdollistamaan palvelun tai palvelukokonaisuuden kopioiminen, siten että palvelun tuotanto- ja myyntikulut laskevat. Koska asiakas on mukana palvelun tuotantoprosessissa, ei lopputulosta voida kuitenkaan määrittää tarkasti. Vaikka yritykset pyrkivät tekemään palvelusta tuotteita, on kyse kuitenkin ainoastaan vakioidusta suunnitelmasta. (Grönroos 2010, 92, Prodman 2013.)

Lehtinen & Niinimäki (2005, 30) kuvaavat asiantuntijapalvelujen tuotteistamisen tarkoittavan asiakkaalle tarjottavan palvelun määrittelyä, kuvaamista, suunnittelua, kehittämistä, tuottamista ja jatkuvaa parantamista. Heidän mukaan tuotteistamisessa on kyse paljon laajemmasta kuin pelkästään palvelun standardisoimisesta. Tuotteistuksen tavoitteena on kehittää palvelua, jotta se vastaisi paremmin asiakkaan tarpeita. Markkinoinnin näkökulmasta tuotteistettu tuote on helpommin markkinoitavissa kuin epämääräinen asiakasprojekti. Näin ollen markkinointi on merkittävässä osassa tuotteistamistyössä. Tuotteistettu palvelu puolestaan on usealle asiakkaalle myytävissä oleva palvelukokonaisuus.

Grönroos kirjoittaa laajennetun palvelutarjooman mallista, joka käsittää palvelun tuotteistamiseen liittyviä piirteitä. Palvelun tuotteistamisessa tulee ottaa lähtökohdaksi asiakkaan näkökulma. Grönroosin (2010, 222) mukaan liian usein tukeudutaan liiaksi sisäisiin näkökohtiin ja kiinnitetään liian vähän huomiota tutkimuksen tietoihin tai ei ymmärretä riittävästi asiakkaan näkökulmaa. Lehtinen & Niinimäki (2005, 31) puolestaan kirjoittavat, että tuotteistamista ei ole aina tarkasteltu kokonaisuutena ja sen vuoksi se on johtanut liialliseen standardisoimistyöhön. Tuotteistamisen tarkoituksena on muokata palvelua, jotta asiakashyödyt maksimoituvat ja asiantuntijaorganisaation tulostavoitteet saavutetaan.

Grönroosin termi (2010, 222-223) palvelupaketti kuvaa tuotetta siten kuin se tässä opinnäytetyössä käsitetään. Se on kokonaisuus, mikä koostuu konkreettisista tai

ainettomista palveluista. Palvelupaketilla voidaan kuvata palvelun luonnetta ja lopputulosta. Palvelu tuotteena on kuitenkin monimutkaisempi, koska asiakkaan kokemaan laatuun vaikuttaa paitsi lopputulos myös tuotantoprosessi, kuten aikaisemmassa luvussa on kuvattu. Palvelutarjooman tulee olla asiakaslähtöinen ja sen suunnittelussa on otettava huomioon sekä lopputuloksen tekninen laatu että palveluprosessin toiminnallinen laatu.

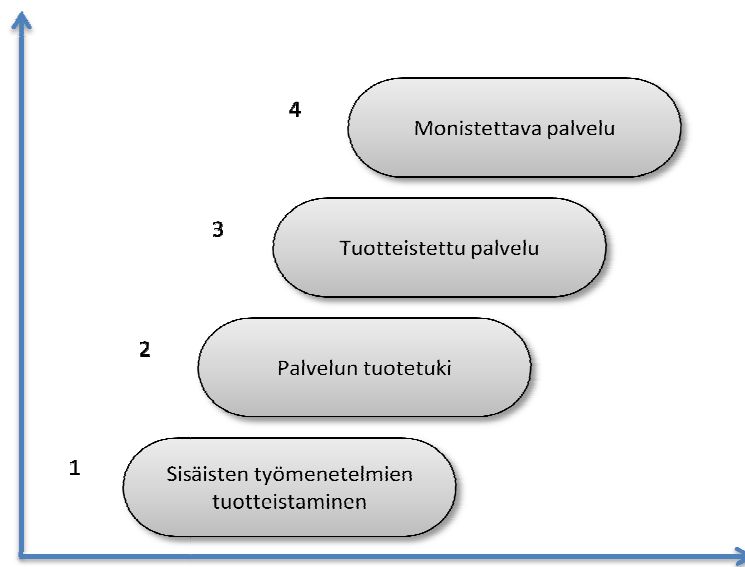
Perinteisesti asiantuntijapalveluita on tuotettu yksilöllisesti ja asiakaskohtaisesti ilman systemaattisia standardeja. Ajan kuluessa joillakin toimialoilla on kuitenkin havaittu, että palveluiden tuottaminen yksilöllisesti on niin kallista, etteivät asiakkaat ole halukkaita maksamaan niiden tuotantokustannuksia. Lisäksi palvelun laatu vaihtelee paljon asiakas- ja tilannekohtaisesti riippuen muun muassa asiakkaan osallistumisen asteesta ja osaamisesta sekä muista tekijöistä. Tasaisen laadun varmistamiseksi palveluita standardoidaan eri tavoin. Täysin standardoitu palvelu on aina samanlainen, mutta käytännössä palvelu on sitä erilaisempi mitä enemmän palvelutuotantoon liittyy inhimillisiä tekijöitä. Standardoitu palvelu on tuotantokustannuksiltaan edullinen. Standardoinnin etu on myös se, että palvelu on helpommin opetettavissa muille. Palvelu voi olla myös hyvin pitkälle räätälöity, eli yksilöity. Räätälöity palvelu sopii erityisesti tilanteisiin, jossa vuorovaikutusta on paljon sekä asiakkaan tarve on hyvin yksilöllinen. (Lehtinen & Niinimäki 2005, 34.)

Standardoidun ja räätälöidyn palvelun välimaastossa on massaräätälöity palvelumalli. Mallissa standardoidaan tiettyjä osia moduuleiksi, joita asiakas voi valita tarpeensa mukaisesti. Moduulien avulla tuotantokustannuksia voidaan pienentää ilman asiakkaan valinnanvaran pienentämistä. Toisaalta jos modulointia pyritään viemään liian pitkälle, saattavat tuotantokustannukset nousta lähelle räätälöidyn palvelun tuotantokustannuksia. (Lehtinen & Niinimäki. 2005, 34-35.)

Tuotteistaminen voidaan jakaa sisäiseen ja ulkoiseen tuotteistamiseen. Sisäinen tuotteistaminen on asiakkaalle näkymätöntä toiminnan kehittämistä ja sen tarkoituksena on varmistaa, ettei jo kertaalleen tehtyjä asioita jouduttaisi suunnittelemaan ja tekemään uudelleen. Ulkoinen tuotteistus edellyttää sisäistä tuotteistusta ja sillä tarkoitetaan palveluprosessin kuvaamista ja konkretisoimista, jotta asiak-

kaat voisivat nopeasti saada kuvan yrityksen toiminnasta ja siitä mitä hyötyjä yritys voi tuottaa asiakkailleen. (Lehtinen & Niinimäki 2005, 43-44)

Kuvio 10 kuvaa tuotteistamisen eri tasoja, jossa alimpana tasona on sisäisten työmenetelmien tuotteistaminen ja ylimpänä taas kokonaan monistettavissa oleva palvelu, kuten esimerkiksi lentomatka.



Kuvio 10. Tuotteistamisen tasot. (Sipilä 1996)

Palvelun tuotteistamisen tarkoituksena ei ole siis ainoastaan vakioida ja kopioida palveluprosessia ja siten pienentää yksikkökohtaisia tuotantokustannuksia eikä kyse ole myöskään tavasta löytää parempia keinoja markkinoida yrityksen palveluita. Sen sijaan tarkoituksena on näiden molempien tehostaminen sen kautta, että palvelua suunniteltaessa pyritään kehittämään tuote, joka parhaiten vastaa asiakaskunnan tarvetta.



### 3.3 Tuotteistamisen hyödyt yritykselle

Tuotteistamisen tavoitteena on parantaa tuottavuutta optimoimalla kustannusrakennetta ja hyödyntämällä resursseja paremmin organisaatiossa. Asiakkaille ei tuoteta räätälöityjä palveluita, joiden yksikkökustannukset ovat korkeammat vaan tuotteita, jotka ovat mahdollisimman pitkälle vakioituja. Tuotteistuksen tavoitteena ei ole myöskään tuottaa valmista tuotetta, kuten ohjelmistoa, vaan tuottaa osia tuotteista sekä luoda malleja, joiden avulla tuotetta valmistetaan. Voidaan puhua myös tuotteistusasteesta, eli siitä kuinka pitkälle tässä tapauksessa ohjelmisto on vakioitu ennen asiakaskohtaisia muutoksia. Resurssien käyttöä voidaan parantaa vähentämällä asiakaskohtaisen työn määrää siellä missä se on mahdollista. (Tyrväinen 2003, 22-23.)

Tuotteistamisen tuloksena syntyy valmiita toimintamalleja ja prosesseja, jotka ovat peräisin sisäisestä tuotteistamisesta. Nämä mallit ja prosessit edistävät organisaation oppimista. Nuoret aloittelevat asiantuntijat pääsevät melko helposti alkuun työtehtävissään, koska tehtävät ovat selkeitä. Tehtävät voivat olla jossakin määrin rutiinin omaisia. Näin toimien organisaation kehittyneet asiantuntijat pystyvät keskittämään osaamisensa hankalampiin tapauksiin. Organisaatio hyötyy, koska parhaat asiantuntijat pystyvät keskittymään tuottavimpiin työtehtäviin ja nuoremmat asiantuntijat pystyvät tekemään vähemmän tuottavia tehtäviä ja oppivat tekemisen kautta. (Lehtinen & Niinimäki 2005, 31.)

## 4 CASE: YRITYS X

Yritys X:n osasto A toimittaa Microsoftin Dynamics AX –ERP-järjestelmiä asiakkailleen. Myytäviin palveluihin kuuluvat muun muassa järjestelmän käyttöönotot, versionvaihdokset sekä järjestelmän ylläpitopalvelut. Asiakkaat ovat sekä julkiselta että yksityiseltä puolelta ja valtaosa niistä on Suomen mittakaavassa suuria yrityksiä. Suurina yrityksinä niiden liiketoiminnan erityisluonteiden vuoksi ohjelmistoon tulee tehdä usein laajamittaisia muutoksia ennen käyttöönottoa sekä ylläpidon aikana. Seuraavissa luvuissa käydään läpi yritysesityksen lisäksi tutkimuksen toteutus, tulosten analysointi ja johtopäätökset.

### 4.1 Tutkimuksen toteutus

Tutkimus toteutettiin teemahaastatteluina syys-lokakuussa 2014. Haastatteluun valittiin tiettyjen asiakkaiden edustajia sekä Yritys X:ltä eri tehtävissä toimivia henkilöitä. Yritys X:n edustajan kanssa valittiin henkilöitä, joita voitaisiin haastatella. Valituille henkilöille lähetettiin kutsu sähköpostitse ja haastattelu saatiin sovittua kuuden henkilön kanssa, joista kolme oli asiakkaiden edustajia ja kolme Yritys X:n edustajia. Kaikilla haastatelluilla on kokemusta Yritys X:n toimittamasta AX-projektista ja osalla myös muista tietojärjestelmien toimitusprojekteista. Yksi haastattelu tehtiin kasvotusten ja loput viisi tehtiin puhelimitse. Haastattelun teemat ja kysymykset ovat tutkimuksen liitteenä (liite 1). Kunkin haastattelun kesto oli 30-45 minuuttia ja haastattelut nauhoitettiin haastateltujen suostumuksella. Haastattelun jälkeen nauhoitus litteroitiin analysointia varten. Taulukossa 1 on listattuna toteutetut teemahaastattelut.

## TAULUKKO 1. Teemahaastattelut

<b>Titteli</b>	<b>Asiakas/toimittaja</b>	<b>Päivämäärä</b>
Projektipäällikkö	Asiakasyritys A	29.9.2014
Konsultti	Yritys X	30.9.2014
Pääkäyttäjä	Asiakasyritys B	1.10.2014
Testauspäällikkö	Yritys X	1.10.2014
Projektipäällikkö	Yritys X	21.10.2014
Projektipäällikkö	Asiakasyritys C	22.10.2014

Kaikille haastatelluille aihe oli varsin tuttu ja erityisesti testaukseen liittyvät haasteet olivat hyvin tiedostettu. Koska jokaisen haastatellun tausta ja rooli projektiryhmässä oli ollut erilainen, saatiin haastatteluista erilaisia näkemyksiä ERP-järjestelmän laatuun ja testaukseen liittyen sekä ideoita siitä kuinka testausta voidaan kehittää ja kuinka testauspalvelua voitaisiin tuotteistaa.

Tutkimus jakautui kolmeen pääteemaan, ERP-järjestelmän laatuun, ERP-järjestelmäprojektin testaukseen sekä toimittajan tarjoamiin testauksen lisäpalveluihin. Kukin teema koostui 3-5 avoimesta kysymyksestä kyseiseen teemaan liittyen. Pääasiassa kysymykset kysyttiin suunnitellussa järjestyksessä, mutta haastattelut olivat keskustelunomaisia, jossa tilanteen mukaan kysymysrungosta joustettiin.

Seuraavaksi on kuvattu Yritys X:n osaston A projektitoiminnot.

#### 4.2 Projektitoiminnot osastolla A

Yritys X:n osastolla A merkittävä osa toiminnasta kohdistuu meneillä oleviin käyttöönotto- ja versionvaihdosprojekteihin. Käyttöönottoprojektilla tarkoitetaan projektia, jossa asiakasyritys ottaa käyttöönsä MS Dynamics AX -järjestelmän.

Käyttöönottoprojektit ovat usein pitkäkestoisia ja ne kestävät kuukausista vuosiin. Pääosin projekteissa menetelmänä on vesiputousmenetelmä, mutta myös ketteriä menetelmiä on käytetty joissakin projekteissa. Projekteissa sovelletaan Microsoft Dynamics Sure Step –metodologiaa, joka on kuvattu myöhemmin omassa luvussaan tarkemmalla tasolla. Sure Step tukee projektipäällikön työtä tarjoamalla koostusti parhaita käytäntöjä projektin johtamiseen. Osastolla projektit vaiheistetaan Sure Stepin mukaisiin vaiheisiin.

Versionvaihdosprojektissa asiakkaan käytössä oleva AX:n versio päivitetään uudempaan versioon. Uudemmissa versioissa on aina paranneltua ja lisättyä toiminnallisuutta aikaisempiin versioihin nähden. Näin ollen versionvaihdoksessa otetaan käyttöön uutta toiminnallisuutta, korvataan vanhaa sekä varmistetaan, että aikaisemmat käyttöön jäävät ratkaisut ovat yhteensopivia uuden version ratkaisujen kanssa. Versionvaihdos kestää yleensä joistakin kuukausista noin vuoteen. Käyttöönottoprojektiin verrattuna versionvaihdos on jossakin määrin helpompi testauksen näkökulmasta. Erityisesti asiakkaan loppukäyttäjät pystyvät testaamaan tehokkaammin, koska järjestelmän logiikka on entuudestaan tuttua, vaikkakin uudessa järjestelmässä on uusia piirteitä.

Projektitiimi koostuu projektipäälliköstä sekä vaihtelevasta määrästä asiantuntijoita, kuten sovellusarkkitehti, tekninen arkkitehti, eri osa-alueiden konsultit sekä kehittäjät. Useimmissa projektitiimeissä ei ole erikseen testaajia, vaan vastuu riittävästä testauksesta on projektipäälliköllä ja testauksen suorittamisesta konsulteilla. Osastolla A ei ole erikseen tarkkaa ohjeistusta ja koulutusta testaukseen.

Myöskään testaussuunnitelmaa tai testausraportointia ei edellytetä projektitiimiltä, eikä sen vuoksi ole ohjeistusta siitä, missä vaiheessa testaus voidaan päättää. Projekteissa tehdään kuitenkin testauksen suunnittelua ja raportointia kulloinkin parhaaksi katsotulla tavalla. Projekteissa on käytetty niin sanottua testauspöytäkirjaa testauksen havaintojen dokumentointiin. Toiminnallisuuden ratkaisumääritysdocumentissa on mahdollisuus kirjoittaa toiminnallisuuden testausohje sekä suunnitella testitapauksia. Teoriaosuudessa luvussa 2.4 on kuvattu TMMi Foundationin ylläpitämä testausprosessin kehittämisen viisitasoinen portaittainen malli Test Maturity Model (TMM). Tällä viisitasoisella mallilla arvioituna Yritys X:n osasto A:lla suoritettu testaus on toiseksi alimmalla tasolla. Testaus suoritetaan debugga-

uksesta erillisenä toimintona ja sitä suunnitellaan jonkin verran, mutta testauksen suunnittelu aloitetaan pääasiassa vasta koodauksen jälkeen. Kolmannella tasolla oleminen edellyttäisi testausorganisaation eriyttämistä ohjelmistokehitystä suoritavasta osastosta.

Yksikön projekteissa tyypillisesti suoritetaan integrointitestausta ja järjestelmätestausta ohjelmistokehityksen eri vaiheissa. Testaus rajoittuu toteutettuihin muutoksiin, eikä perusjärjestelmää tavallisesti testata laajemmin, koska se on testattu jo Microsoftin puolesta osana järjestelmän kehitystä. Kuitenkin järjestelmätestauksessa testataan asiakkaan prosesseja, jolloin myös perusjärjestelmä tulee asiakkaan käyttämien prosessien osalta testattua. On myös huomattava, että yleisimmin asiakas ottaa käyttöön vain pienen osan järjestelmän toiminnallisuudesta, eikä koko järjestelmää edes kannata sen vuoksi testata. Testauksessa keskitytään ennen kaikkea siihen, että varmistutaan järjestelmän toimivan määritellyllä tavalla ja tuottavan oikeita tuloksia. Mikäli testissä havaitaan vikoja, debuggaa kehittäjä tarvittaessa virheen aiheuttaneen prosessin ja korjaa mahdollisen koodivirheen, jonka jälkeen testaus suoritetaan uudelleen.

Projekteissa regressiotestaus on tärkeätä, koska myöhäisemmässä vaiheessa toteutettu muutos saattaa vaikuttaa aikaisemmassa vaiheessa toteutetun muutoksen toimintaan. Regressiotestaus suoritetaan konsulttien toimesta, eikä siihen käytetä testausautomaatiota. Yritys X:llä on erillisessä yksikössä testauksen ammattilaisia, esimerkiksi suorituskyvyn ja tietoturvan testaukseen ja osittain heidän tuottamia palveluita on myyty projekteissa asiakkaalle.

Kun järjestelmä on otettu käyttöön, siirrytään ylläpitovaiheeseen. Asiakkaiden liiketoiminta muuttuu jatkuvasti ja sen vuoksi ERP-järjestelmään tehdään muutoksia, eli niin kutsuttua pienkehitystä. Testauksen näkökulmasta pienkehityksessä on samoja piirteitä kuin projektin aikanakin. Toiminnallisuus testataan ja samalla tulee suorittaa regressiotestaus. Myöskään pienkehitykseen liittyvässä regressiotestauksessa ei käytetä automaattitestausta.

### 4.3 Microsoft Dynamics Sure Step

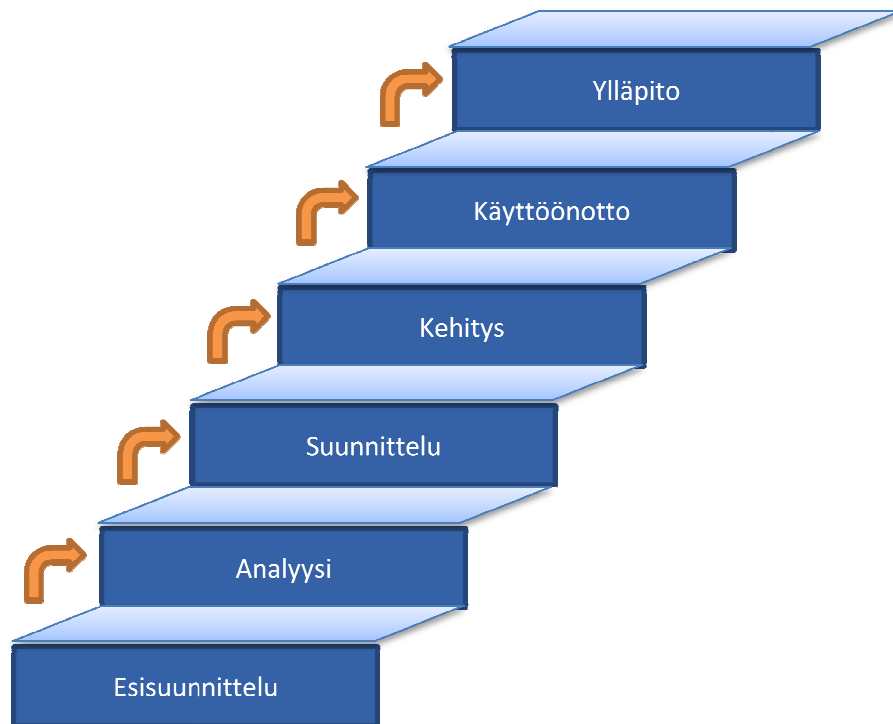
Microsoft Dynamics Sure Step on ohjelma, joka kokoaa vuosien tietotaidon sekä parhaat käytännöt projekteihin ja siten helpottaa projektipäälliköiden ja koko projektitiimin työskentelyä. Työkalu on tarkoitettu Dynamics –tuoteperheen tuotteiden toimituksiin ja sen avulla Microsoft haluaa varmistaa omien tuotteidensa loppukäyttäjien hyvän kokemuksen toimitusprojektista ja sitä kautta itse tuotteen. Ohjelman avulla työskentelyä voidaan tehostaa, koska se auttaa selkiyttämään projektitiimin vastuita ja tehtävien järjestystä. (Dunkinson & Birch 2013, 1-4.)

Työkalussa on kuvattu tarkkaan projektin eri vaiheet ja vastuualueet. Projektit on jaettu kuuteen vaiheeseen, joille ei ole virallisia suomen kielisiä käännöksiä. Projektin eri vaiheissa on tietyt päätehtävät. Vaiheet ja päätehtävät on kuvattu alla listalla sekä kuviossa 11.

1. esimäärittely (Diagnostics)
  - Tehdään karkean tason analyysi asiakkaan prosessista
2. määrittely (Analysis)
  - Tehdään tarkka analyysi asiakkaan tarpeista ja vaatimuksista
  - Kirjoitetaan vaatimusmäärittelydokumentit
3. suunnittelu (Design)
  - Löydetään tavat, joilla asiakkaan tarpeet ja vaatimukset saadaan sovitettua käyttöön otettavan järjestelmän ratkaisuihin.
  - Kirjoitetaan ratkaisumäärittelydokumentit
4. kehitys (Development)
  - Tehdään varsinainen kehitystyö, eli koodaus
  - Suunnitellaan datan siirto uuteen järjestelmään
  - Suoritetaan yksikkö- ja integrointitestausta
5. käyttöönotto (Deployment)
  - Tehdään käyttöönottoa edeltävät tehtävät, kuten ympäristön pystytys ja konfigurointi
  - Suoritetaan järjestelmä- ja hyväksymistestausta

- Loppukäyttäjien koulutus
6. ylläpito (Operation)
- Järjestelmän käyttö

(Gupta 2013)



KUVIO 11. Microsoft Dynamics Sure Stepin mukaiset projektivaiheet. (BSE Consulting 2014)

Sure Step –ohjelmasta saadaan myös tukea järjestelmän testaukseen excel- taulukon muodossa, jossa on listattu testattavat perusjärjestelmän ominaisuudet. Listaa voidaan käyttää hyväksi järjestelmä- ja hyväksymistestauksessa. Taulukossa on testausohje, ennakoitu tulos ja testaaja sekä tieto siitä, onko kohde läpäissyt testauksen. Järjestelmä- ja hyväksymistestaus suoritetaan Sure Stepin mukaan käyttöönottovaiheessa, mutta testauksen suunnittelu tulisi tehdä aikaisemmissa vaiheissa. Aikaisemmista vaiheista tulisi syntyä dokumentoitu testi- ja laatusuunnitelma, josta tulisi käydä ilmi testauksen hyväksyntäkriteerit. Suunnitelmassa

tulisi olla huomioituna testitapaukset, käytettävä data, odotettu tulos sekä hyväksyntäkriteerit. Tulokset tulee dokumentoida ja verrata tuloksia hyväksyntäkriteereihin ja arkistoida myöhempää tarvetta varten. (Dunkinson & Birch 2013, 177-178.)

#### 4.4 Tutkimuksen tulokset

Seuraavaksi esitellään teemahaastatteluissa saatuja tuloksia teemojen mukaisessa järjestyksessä. Teemoja oli kolme ja kukin teemoista jakautui 3 – 5 avoimeen kysymykseen. Ensimmäinen teema koski ERP-järjestelmien laatua, toinen koski testausta osana ERP-toimitusprojekteja ja kolmas toimittajan tarjoamia testauksen lisäpalveluita.

##### 4.4.1 ERP-järjestelmän laatu

Haastattelun ensimmäinen teema käsitteli ERP-järjestelmän laatua. Järjestelmän testaus on ennen kaikkea varmistaa se, että järjestelmä vastaa laadultaan määritettyä ja asiakkaan odottamaa laatua. Teoriaosassa esiteltiin ISO-standardin 25010 mukaiset tietojärjestelmän laadun osatekijät, joita oli seitsemän. Kysyttäessä vastaajan mielipidettä tärkeimmistä laadun osatekijöistä ISO-standardin mukaisia osatekijöitä mainitsematta, pidettiin kaikissa vastauksissa oikeellisuutta ja käytettävyyttä tärkeimpinä osatekijöinä. Käytettävyyttä tärkeänä laadun osatekijänä piti neljä vastaajaa ja yhteentoimivuutta kolme vastaajaa. Taulukossa 2 on taulukoituna vastausten jakautuminen eri laadun osatekijöiden kesken. Huomiota herättävästi rikkomattomuutta, siirrettävyyttä ja ylläpidettävyyttä ei pidetty haastattelun perusteella tärkeinä osatekijöinä. On kuitenkin huomioitava, että haastateltavat ovat projektityöntekijöitä, eikä järjestelmän ylläpidosta vastaavia. Voidaan olettaa, että järjestelmän ylläpidosta vastaavat henkilöt pitäisivät esimerkiksi siirrettävyyttä ja ylläpidettävyyttä pidetään tärkeinä laadun osatekijöinä. Vastausten lukumäärän lisäksi suluissa on merkitty kuinka vastaukset jakautuivat asiakkaiden (A) ja toimittajan (T) edustajien kesken.



TAULUKKO 2. Laadun osatekijät teemahaastattelun perusteella.

Laadun osatekijä	Vastausten lukumäärä (A = asiakkaan edustaja T = toimittajan edustaja)
Oikeellisuus	6/6 (AAATTT)
Luotettavuus	6/6 (AAATTT)
Käytettävyys	4/6 (AATT)
Rikkomattomuus	
Siirrettävyys	
Ylläpidettävyys	1/6 (T)
Yhteentoimivuus	3/6 (AAT)

Lisäksi kysyttiin käyttöönotetun ERP-järjestelmän laatuun vaikuttavista tekijöistä ja niiden merkityksestä. Testauksella pyritään varmistamaan järjestelmän laatu ohjelmistokehityksen aikana, ennen järjestelmän käyttöönottoa. Testauksessa toiminnallisuutta verrataan asiakkaan vaatimus- ja ratkaisumäärittelyihin sekä asiakkaan että toimittajan toimesta. Vastaukset luokiteltiin viiteen luokkaan sen mukaan kuinka vaikuttavana tekijää pidettiin. Luokat olivat hyvin suuri vaikutus, suuri vaikutus, kohtalainen vaikutus, jonkinlainen vaikutus ja merkityksetön vaikutus.

Vaatimus- ja ratkaisumäärittelyiden vaikutusta lopulliseen laatuun vastaajat pitivät suurena. Toimittajan suorittaman testauksen vaikutus jakautui enemmän vastaajien kesken, sillä kolme vastaajaa piti sitä hyvin suurena, kaksi vastaajaa suurena ja yksi vastaaja merkittävänä. Asiakkaan suorittamaa testauksen vaikutusta kaikki vastaajat pitivät sen sijaan hyvin suurena. Erityisesti vastauksissa korostui se, että asiakkaalla on kaikki tietämys asiakkaan liiketoiminnan tarpeista ja vaatimuksista,

kun taas toimittaja ei välttämättä osaa painottaa oikeita asioita. Lisäksi pidettiin tärkeänä sitä, että asiakkaan testauksessa on mukana järjestelmän loppukäyttäjät.

*Mun mielestä kyllä se asiakkaan testaus on tärkeämpi, koska toimittaja ei voi tietää kaikkea tai ei pysty testaamaan kaikkea. Koska se ei tiedä asiakkaan kaikkia tarpeita, vaikka ne oiski käyty läpi, ni se ei kuitenkaa sillä tavalla ymmärrä niitä. Että asiakkaan testaus on ehdottoman tärkeää, mutta kun sitä testaamista on niin tajuttoman paljon, niin jos ei sitte toimittaja testaa mitään, ni sit se asiakas hukkuu niihin testaamisiin.*

*(asiakasyritys B edustaja 2014)*

Vastauksista käy ilmi se, että toimittajan testaus koetaan tärkeäksi erityisesti siitä syystä, että huolellinen testaus kehityksen aikana nopeuttaa asiakkaan suorittamaa hyväksymistestausta. Vaatimus- ja ratkaisumäärittelyiden merkitystä pidettiin tärkeänä, mutta toisaalta huomioitiin että erityisesti vaatimukset voivat muuttua projektin aikana, joten osin siitä syystä niiden vaikutusta ei pidetty niin suurena, kuin testauksen vaikutusta. Kuten teoriaosassa on kuvattu, tulisi testauksen tapahtua vaatimus- ja ratkaisumäärittelydokumentteja vasten. Vastauksista käy ilmi, ettei käytännössä testausta voida kuitenkaan välttämättä tehdä määrittelydokumentteja vasten, koska ne eivät välttämättä ole ajan tasalla. Myers mukaan puolestaan hyväksymistestausta tulisi tehdä alkuperäisiä vaatimuksia ja nykyisiä loppukäyttäjän tarpeita vasten. Osa haastatelluista korosti, että vaatimusmäärittelydokumentteja täytyy pystyä päivittämään projektin aikana, jotta ne vastaavat ostajan muuttuneita vaatimuksia. Lyytinen & Hirschheimin (1987) mukaan yksi projektin epäonnistumisen syistä johtuu siitä, ettei vaatimuksia ole kuvattu riittävän hyvin.

#### 4.4.2 ERP-järjestelmän testaus osana toimitusprojektia

Toisena teemana käsiteltiin projektin aikaista testausta. Ensimmäisenä kysymyksenä kysyttiin vastaajan näkemystä siitä, kuinka suuri toimittajan vastuu on, että järjestelmä vastaa asiakkaan vaatimuksia. Vastaukset luokiteltiin jälleen viiteen luokkaan, jotka olivat hyvin suuri vastuu, suuri vastuu, merkittävä vastuu, jonkinlainen vastuu ja merkityksetön. Toimittajan vastuuta piti suurena vastaajista neljä ja merkittävänä kaksi vastaajaa. Huomion arvoista on, että sekä ostajan että toimittajan edustajan vastaukset jakautuvat samassa suhteessa. Testaus on osa toimi-

tusprojektin palveluprosessia ja useissa vastauksissa korostuikin palveluprosessin merkitys suhteessa asiakkaan kokemukseen laadusta. Alla oleva sitaatti korostaa palvelun tuotantoprosessin tärkeyttä osana laadun muodostumista. Teknisen lopputuloksen laatu ei peitä tuotantoprosessin laatuongelmia, vaan asiakas arvioi laatua kokonaisuutena, jolloin sekä teknisen lopputuloksen laatu että tuotantoprosessin laatu yhdessä muodostavat kokonaislaadun.

*Sen asiakkaan kokeman yleisen laadun mielessä, että jos me toimitetaan sinne huonosti testattua ja se asiakas testaa paljon ja joutuu raportoimaan hirveesti virheitä, ni silloin se on, se laatumielikuva, jo huono ennenku se lähtee tuotantoon. Vaikka se tuotannossa olis tosi hyvä.*

*(toimittajan edustaja 2014)*

Teeman toisena kysymyksenä kysyttiin kuinka laajana asiakkaan testausvastuuta pidettiin. Puolet vastaajista piti vastuuta hyvin suurena ja loput kolme pitivät vastuuta suurena. Tutkimuksen perusteella siis ostajan vastuuta testauksesta pidettiin merkittävästi suurempana kuin toimittajan vastuuta. Teoriaosuudessa ei ole asetettu kummankaan osapuolen suorittamaa testausta toista tärkeämmäksi. Toimittajan testauksessa pyritään varmistumaan siitä, että ohjelma toimii ratkaisumäärittelyn mukaisesti ja testauksen suhteen edistyneemmissä toimittajaorganisaatioissa myös vaatimuksia vastaavalla tavalla. Ostajan testauksessa puolestaan korostuu se, että varmistutaan siitä, että ohjelma vastaa asiakkaan tarpeisiin.

Teeman viimeisenä kysymyksenä kysyttiin, mitä tekijöitä vastaajat pitivät projektin aikaisena suurimpana haasteena. Taulukossa 3 on kuvattu vastausten jakautuminen. Suurimpana haasteena pidettiin ostajan henkilöresursseja ja toiseksi suurimpina toimittajan henkilöresursseja ja asiakkaan testausosaamista. Vastaukset korostavat teoriaosuudessa esitettyä käsitystä, että testaus suoritetaan usein vajain resurssein.

TAULUKKO 3. Laatutekijät tutkimuksessa.

Testauksen suurin haaste	Vastausten lukumäärä (A = asiakkaan edustaja T = toimittajan edustaja)
Ostajan henkilöresurssit	5/6 (AAATT)
Toimittajan henkilöresurssit	4/6 (ATTT)
Asiakkaan testausosaaminen	4/6 (AATT)
Asiakkaan kyvykkyys käyttää ohjelmaa	3/6 (AAT)
Projektin kokonaisaikataulu, projektin läpivientiin liittyvät asiat	3/6 (AAT)
Toimittajan ja ostajan välinen kommunikaatio	2/6 (AT)

Testauksen vaatimat resurssit järjestelmän ostajalla oli monessa tapauksessa yllättänyt asiakkaan, sillä testaus edellytti huomattavasti enemmän aikaa kuin oli kuviteltu. Ostajan suorittamassa testauksessa on mukana työntekijöitä, jotka testaavat muiden tehtäviensä ohessa. Työntekijöitä saatetaan tarvita toisaalla organisaatiossa kesken testauksen, mistä johtuen useiden haastateltujen mukaan on tärkeää, että toimittaja pystyy pitämään kiinni sovitusta toimitusaikataulusta. Joissakin tapauksissa koettiin, että asiakkaan testausosaamisessa oli monissa projekteissa puutteita, minkä johdosta testauksen läpivienti ei ollut tehokasta. Siihen vaikutti haastateltujen mukana myös se, että monet asiakkaan testaaajista on loppukäyttäjiiä, joiden osaaminen ei ole tietojärjestelmien testauksessa. Myös uuden järjestelmän käytön opetteleminen oli vielä testauksen aikana asiakkaalla kesken, joten käyttö oli

myös siltä osin tehotonta. Toisaalta testauksen ja järjestelmän käytön opettelemisen kombinoimista pidettiin tärkeänä, jotta loppukäyttäjät osaisivat käyttää järjestelmää tehokkaasti käyttöönoton jälkeen.

Lisäksi korostettiin sitä, että toimittajan testauksessa tulisi käyttää todellisen tuntuista dataa ja ohjelman parametrien tulisi olla asetettu, kuten asiakkaan ympäristöissä, jotta virheet havaittaisiin paremmin toimittajan ympäristössä, eikä vasta asiakkaan aloittaessa testauksen. Sekä asiakkaan että toimittajan suorittamaan testaukseen viitaten huomautettiin, että testauksen tulisi kattaa myös negatiivisen testauksen.

*Ja sit semmonen periaatteellinen asia, et pitää ymmärtää että testaushan ei ole pelkästään sitä, että miten sen pitäisi mennä, vaan sitten se on myös sitä, et miten se ei saisi mennä. Että joutuu niinku ne negatiot sinne syöttään ja mitä sitten tapahtuu.*

*(asiakas A:n edustaja 2014)*

Negatiivinen testaus, eli tarkoituksellinen virheen aiheuttaminen esimerkiksi virheellisellä tai puutteellisella datalla, edellyttää testitapausten huolellista suunnittelua. Teoriaosuuden mukaan testitapausta suunniteltaessa tulee myös päättää tapausten oletettu lopputulos, minkä merkitys korostuu negatiivisessa testauksessa.

#### 4.4.3 Toimittajan tarjoamat testauksen lisäpalvelut

Haastattelun kolmantena teemana oli toimittajan tarjoamat testauksen lisäpalvelut. Ensimmäisenä haastattelukysymyksenä kysyttiin, kuinka toimittaja voisi paremmin tukea ostajaa testauksessa. Vastauksista nousi esiin kolme eri asiaa, testauksen kouluttaminen asiakkaalle, avustaminen testaussuunnitelman ja testitapausten tekemisessä sekä avoin kommunikaatio. Toimittajan edustajien vastausten mukaan erityisesti testauksen koulutukselle olisi tarvetta, sillä kaikki toimittajan edustajat olivat tätä mieltä, mutta asiakkaan vastauksista ei kohta noussut erityisesti esiin.

Yhden vastaajan kokemuksen mukaan toimittajalta tullut testauskoordinaattori sekä testausta suorittavat konsultit asiakkaan tiloissa ovat edesauttaneet huomattavasti testausta, koska kommunikointi on suoraa ja nopeaa sähköposteihin tai mui-

hin kommunikointivälineisiin verrattuna. Tällä tavoin sekä ostaja, että toimittaja voivat toimia avoimesti yhteistyössä ja näin toimien tehostaa ja nopeuttaa testausta. Vastauksissa korostuu selkeästi palveluprosessin merkitys palvelun myynnissä. Erityisesti asiakkaiden edustajien vastauksissa korostui tapa, jolla palvelua tuotetaan, eikä niinkään testauspalvelulla saavutettavat hyödyt. Toisaalta korostuu myös vaikeus, joka asiantuntijapalvelujen ostajalla on määritellesään omaa tarvettaan.

Teeman toinen kysymys tapoja, joilla ostaja voi tukea toimittajaa testauksessa. Vastauksista oli tunnistettavissa panostus määrittelytyöhön, avoin kommunikaatio sekä yhdestä vastauksesta testitapausten suunnittelu. Erityisesti avoimen kommunikaation ja testitapausten suunnittelun osalta tausta-ajatuksena näyttää vaikuttaneen se, että toimittajan on saatava toimittajan liiketoiminnan erityispiirteistä tietoa riittävästi, jotta testausta voidaan suunnitella. Määrittelyvaiheessa annettua tietoa ei koettu riittäväksi enää testausvaiheessa.

Haastatteluissa nousi esille kokemus siitä, että vaikka monet konsultit ja testauksen teknisestä suorittamisesta vastaavat toimittajan asiantuntijat osaavat testata ja ymmärtävät testauksen tärkeyden, ei kaikilla toimittajan asiantuntijoilla silti ole riittävä osaamista tehokkaaseen testaukseen. Haastatteluiden perusteella voitiin myös tehdä päätelmä, että eri projekteissa toimittajan suorittaman testauksen prosessi vaihteli melko paljon.

#### 4.5 Johtopäätökset

Tutkimuksen johtopäätöksenä seuraavaksi esitetään viisi testaukseen liittyvää ongelmaa, joita esiintyy Yritys X:n Dynamics AX –ERP-järjestelmien toimitusprojekteissa. Ongelman lisäksi on kuvattu ratkaisuehdotuksia ongelmaan. Ongelmat ratkaisemalla toimitettujen järjestelmien laatua voitaisiin parantaa ja siten edistää Yritys X:n liiketoimintaa. Testaus on tärkeässä roolissa ERP-järjestelmien laadunvarmistuksessa.

### **Testaukseen käytettävien henkilöresurssien riittävyys**

Testaukseen käytettävissä olevien henkilöresurssien riittävyys koettiin sekä toimittajan että asiakkaan puolella suurimmaksi testauksen haasteeksi. Asiakkaalla ongelmaksi koettiin ennen kaikkea se, että testauksen määrä toimitusprojektissa oli yllättävän suuri, eikä sitä oltu osattu arvioida etukäteen. Testauksessa suurin työ tehdään projektin loppuvaiheessa, ennen järjestelmän käyttöönottoa. Haasteltujen mukaan aikaisempien projektien vaiheiden viivästyminen saattaa lyhentää testaukseen käytettävissä olevaa aikaa, jolloin saatavilla ei ole riittävästi henkilöresursseja, vaikka alun perin testaukseen olisikin varattu riittävästi henkilötyöpäiviä. Ongelma johtuu pääosin siitä, että testausta ei ole aikataulutettu tarpeeksi tarkasti projektissa, jolloin ennalta testaukseen varatut resurssit ovat riittämättömät. Asiakkaan projektin johto tulee sitouttaa testaukseen jo aikaisessa vaiheessa projektissa ja testaus tulisi suunnitella jo aikaisessa vaiheessa projektissa, eikä vasta koodin valmistumisen jälkeen.

### **Asiakkaan testauksen tehottomuus**

Tutkimuksen johtopäätöksenä voidaan todeta asiakkaan testauksen tehottomuuden olevan ongelma toimitusprojekteissa. Testauksen tehottomuus johtuu muun muassa siitä, että ERP-järjestelmän testaus on jo itsessään hankalaa. Tehokas hyväksymistestaus edellyttää liiketoiminnan prosessien tuntemusta, uuden käyttöönotettavan järjestelmän tuntemusta sekä testausosaamista. Testauksessa on mukana usein asiakkaan loppukäyttäjiiä, mikä on hyvä asia sen kannalta, että loppukäyttäjät oppivat järjestelmän käyttöä testauksen aikana. Toisaalta loppukäyttäjillä ei ole poikkeustapauksia lukuun ottamatta esimerkiksi kykyä suunnitella ja luoda testitapauksia, koska järjestelmän osaaminen ja testausosaaminen ovat puutteellisia. Testaustulokset tulisi myös dokumentoida, jota ei usein asiakkaan hyväksymistestauksessa juurikaan tehdä. Sen seurauksena testaus on joko laadun näkökulmasta riittämätöntä tai ajankäytön näkökulmasta tehotonta. Ongelman ratkaisemiseksi asiakkaan testaajilla tulisi järjestää tukea testaukseen ja sen suunnitteluun.

## **Vaatimus- ja ratkaisumäärittelydokumenttien sisältö**

Testauksessa ongelmia tuottaa se, ettei projektissa tuotetut vaatimus- ja ratkaisumäärittelydokumentit ole aina käyttökelpoisia, koska niissä voi olla epäselvyyksiä, virheitä tai ne eivät vastaa enää vaatimuksia tai ratkaisuja. Määrittelydokumentteja tulisi käyttää toimittajan testauksessa. Burnsteinin (2002, 12-14) mukaan TMMi Foundationin Test Maturity Model (TMM) mallissa toiseksi alimmalla tasolla testaus suoritetaan pääasiassa ratkaisumäärittelyjä vasten ja vasta keskimäisellä, eli kolmannella tasolla, testaus suoritetaan vaatimusmäärittelyjä vasten. Erityisesti vaatimusmäärittelyn ongelmaksi koetaan sekä testauksen teoriassa että teemahaastatteluissa se, että vaatimukset eivät käy selkeästi ilmi tai ne eivät ole enää ajan tasalla. Ratkaisua määriteltäessä voidaan päätyä erilaiseen ratkaisuun, kuin alun perin vaatimusmäärittelyssä, jolloin aikaisempi vaatimus vaihtuu toiseen tai jää kokonaan pois. Ratkaisumäärittelydokumentissa ratkaisu voi olla kuvattuna epäselvästi tai niissä voi olla ristiriita. Ongelman ratkaisemiseksi määrittelydokumentteja tulisi päivittää projektin aikana. Testauksen suunnittelun aloittaminen jo projektin analysointi- ja suunnitteluvaiheissa voisi myös edesauttaa, jotta tuotettavat määrittelydokumentit olisivat myös testauksen näkökulmasta käyttökelpoiset.

## **Toimittajan testausprosessien puutteet ja testauksen laadun vaihtelu**

Tutkimuksen perusteella ongelmana on myös toimittajan testausprosessin puutteet ja testauksen vaihteleva taso eri projekteissa. Toimittajan testausprosessiin liittyviä ongelmia tutkimuksen perusteella on joidenkin testausta suorittavien henkilöiden puutteellinen osaaminen testauksesta sekä toimittajan ja asiakkaan testausjärjestelmien erilainen data sekä parametointi. Toimittaja testaa testausta varten tuotetulla sopivalla datalla, mikä ei välttämättä vastaa oikeaa dataa. Erityisesti datan suhteen tulee huomioida konvertoitava data, eli data, joka tuodaan vanhasta järjestelmästä Dynamics AX:aan. Mikäli toimittajan ja asiakkaan testausympäristöjen parametointi eroaa, saattaa testitapaus läpäistä toimittajan testin, mutta asiakkaan ympäristössä testaus ei onnistukaan. Toimittajan testauksen vaihteleva taso projekteissa johtuu osin jo mainitusta mahdollisesta puutteellisesta testausosaamisesta. Myös yhtenäistettyjen prosessien ja käytäntöjen puuttuminen



johtaa siihen, että testauksen tasoon vaikuttaa monet eri tekijät, kuten asiakas, projektin johto, projektitiimi ja aikataulu. Ongelman ratkaisuksi testaus voitaisiin eriyttää muusta kehityksestä omaksi testaukseen keskittyneeksi tiimiksi, jossa työskentelisi ammattitestaajat. Testauksen eriyttämisellä saavutettaisiin tasalaatuisempi ja objektiivinen testaus projekteista riippumatta, koska testausta suorittaisi ainoastaan ammattitestaajat. Mikäli omaa testaukseen keskittynyttä tiimiä ei ole, tulisi osastolle luoda yhtenäiset testauksen toimintaohjeet, jotka sisältäisi soveltuvin osin testauspolitiikassa ja –strategiassa määritetyt asiat. Lisäksi testausta suorittaville henkilöille tulisi järjestää testauskoulutusta sekä testauksen suunnitteluun että varsinaiseen testaustyöhön.

### **Testauksen kattavuus**

Viimeisenä ongelmana on testauksen kattavuus. Dynamics AX on ERP-järjestelmänä hyvin laaja ja testauksen kattavuus koetaan ongelmaksi. Asiakas käyttää vain osia järjestelmästä, joten kaikkea ei kannata testata, mutta kaikki käyttöönotettavat prosessit ja asiakaskohtaiset sovitukset on testattava. Testitapausten suunnittelu on hankalaa siten, että kussakin projektissa testataan kaikki tarpeelliset toiminnot. Ongelman ratkaisemiseksi testitapausten suunnittelun tehostamiseksi voidaan käyttää Microsoft Sure Stepissa olevia perusjärjestelmän testitapausten listausta. Listausta voitaisiin muokata siten, että siinä on huomioitu suomalaiset erityistarpeet, jotta kyseistä listaa voidaan käyttää soveltuvin osin kaikissa projekteissa. Projektikohtaisesti testitapausten listaukseen voidaan lisätä asiakaskohtaisiin sovituksiin liittyvät testitapaukset.

Johtopäätöksenä voidaan lisäksi todeta, että testauspalvelua voidaan tuotteistaa ennen kaikkea sisäisen tuotteistuksen keinoin. Sisäisellä tuotteistuksella tarkoitetaan sisäisten prosessien ja toimintatapojen vakioimista ja yhtenäistämistä palvelun tuotantokustannusten alentamiseksi. Asiakkaan edustajien oli hankalaa kuvata omaa testauspalvelun tarvetta, vaikka samanaikaisesti asiakkaat kokivat omat henkilöresurssit ja omien testaajien osaamisen osittain puutteelliseksi. Asiakkaan suorittaman testauksen merkitystä pidettiin hyvin suurena. Asiakkaille myytävää testauspalvelua voidaan myös tuotteistaa. Tuotteistamistyössä tulee ottaa näkökulmaksi asiakkaan tarve ja pyrkiä ratkaisemaan asiakkaan ongelma, eikä keskit-

tyä omiin sisäisiin tavoitteisiin. Testauspalvelu voisi koostua moduuleista, joita asiakas voisi yhdistää oman tarpeensa mukaan. Moduuleina voisi olla esimerkiksi testauksen koulutuspäivät hyväksymistestaukseen osallistuville, hyväksymistestauksen testitapausten suunnittelu workshop-tyylisesti asiakkaan tiloissa, asiakkaan testaussuunnitelman katselmointi sekä testauskoordinaattorin palvelut.

## 5 YHTEENVETO

Tämän opinnäytetyön tarkoituksena oli tarkastella Microsoft Dynamics AX -toiminnanohjausjärjestelmän toimitusprojekteihin liittyvää testausta kokonaisuutena järjestelmiä toimittava Yritys X:n näkökulmasta. Tavoitteena oli kartoittaa kokevatko yrityksen asiakkaat tarvetta laajemmalle tuotteistetulle testauspalvelulle järjestelmän käyttöönoton yhteydessä ja millä tavalla palvelua voidaan tuotteistaa sekä selvittää kuinka järjestelmän kehitystyön aikaista testausta voidaan kehittää toimittajan näkökulmasta toimitusprojektien laadun parantamiseksi. Työn tavoitteet saavutettiin ja tutkimuksen johtopäätöksenä on esitetty useita kehitysehdotuksia toimitusprojekteissa havaittujen ongelmien ratkaisemiseksi ja laadun parantamiseksi. Samoin tutkimuksen johtopäätöksissä on esitetty näkemys, mistä testauspalvelu voisi koostua.

Työn teoriaosuus muodostui ensiksi ERP-järjestelmien testauksesta, jossa tarkasteltiin ensin ERP-järjestelmiä ja niiden laatutekijöitä sekä järjestelmän testaukseen liittyvää teoriaa. Toinen osuus käsitteli asiantuntijapalvelun tuotteistamista ja siitä koituvia hyötyjä yrityksille. Teoriaosuudessa on käytetty lähteenä useita tietojärjestelmien testausta ja asiantuntijapalveluiden tuotteistamista käsitteleviä teoksia sekä tietojärjestelmien testausta ja laatua käsitteleviä ISO-standardeja. Lisäksi lähteenä on käytetty muun muassa Jussi Kasurisen väitöskirjaa.

Tutkimusaineisto kerättiin kvalitatiivisella eli laadullisella tutkimusmenetelmällä. Tutkimuksen aineisto kerättiin teemahaastatteluilla, jotka toteutettiin syksyllä 2014 sekä tutkijan omana osallistuvana havainnointina. Haastattelurunko perustui teoriaosuuteen ja teemoina oli ERP-järjestelmien laatu, ERP-järjestelmien testaus osana toimitusprojektia sekä toimittajan tarjoamat testauksen lisäpalvelut. Haastateltavana oli kolme toimittajan edustajaa sekä kolme eri asiakkaiden edustajaa. Kaikki haastateltavat ovat olleet mukana Yritys X:n toimittamassa Dynamics AX -toimitusprojektissa.

Tutkimusongelmia oli neljä, joihin vastaukset on esitetty alla:

**Onko yrityksen Dynamics AX-toimitusprojekteissa suoritettu testaus laadunvarmistuksen näkökulmasta riittävää?**

Tutkimusongelmaan sisältyy oletus siitä, että laadunvarmistuksella pyritään varmistamaan järjestelmän laadun vastaavan asiakkaan odottamaa laatua. Testaus käsittää sekä toimittajan että asiakkaan suorittaman testauksen projektin eri vaiheissa. Toimittajalta puuttuu yhtenäiset testauksen prosessit, joten testauksen taso vaihtelee eri projekteissa. Vastaavasti asiakkaan testaukseen käytettävät resurssit ovat usein puutteelliset ja testaajien osaaminen on puutteellista. Vaikka monissa projekteissa saavutetaan hyvä laatu, ei testaus ole laadunvarmistuksen näkökulmasta riittävää. Erityisesti testauksen prosessien yhtenäistämiseen ja testauksen suunnitteluun tulisi panostaa, jotta voidaan varmistua siitä, että testaus on riittävän kattavaa ja sillä voidaan varmistaa järjestelmän korkea laatu.

**Onko asiakasyrityksillä riittävä osaaminen kattavaan ja tehokkaaseen testaukseen?**

Tutkimuksen perusteella voidaan todeta, että yleisesti ottaen asiakasyrityksillä ei ole riittävää osaamista kattavaan ja tehokkaaseen testaukseen. Microsoft Dynamics AX –ERP-järjestelmän käyttöönottoon liittyvänä erityispiirteenä testaajilta edellytetään osaamista paitsi yrityksen olemassa olevista prosesseista sekä uudesta käyttöönotettavasta järjestelmästä myös tietojärjestelmien testauksesta. Useimmissa tapauksissa osaaminen on riittämätöntä, jolloin asiakkaan suorittama testaus ei ole riittävän kattavaa ja tehokasta. Vaikka yrityksessä olisikin osaamista tietojärjestelmien testauksesta, on useimpien testaajien osalta testauksen osaaminen silti riittämätöntä. Asiakkaan suorittamassa hyväksymistestauksessa käytetään paljon järjestelmän loppukäyttäjiä, joilla ei yleensä ole juuri lainkaan osaamista testauksesta.

### **Kokevatko asiakkaat tarvetta laajennetulle testauspalvelulle?**

Asiakasyrityksissä koetaan tarvetta testauksen tuelle. Testauksen tuen muotoina asiakkaat esittivät muun muassa tukea testaussuunnitelman tekemiseen ja testitapausten suunnitteluun. Lisäksi asiakkaat toivoivat muun muassa parempaa kommunikointia ja avointa testausprosessia puolin ja toisin. Kuitenkaan asiakkaat eivät välttämättä kokeneet tarvetta tietylle nimetyille tai vakioidulle testauspalvelulle. Asiakkaiden haastattelussa korostui se, että palvelua arvioitaessa tuotantoprosessin laatu koetaan mahdollisesti jopa tärkeämmäksi kuin varsinaisen palvelun lopputuloksen laatu. Tutkimuksen johtopäätöksissä on esitetty ratkaisuja testauspalvelun tuotteistamiselle, jossa käsitellään myös asiakkaalle myytävät laajennetut testauspalvelut.

### **Kuinka toimittaja voi kehittää Dynamics AX-toimitusprojektien testausprosesseja?**

Toimittajan testauksen prosesseja voidaan kehittää, jotta testauksella pystytään paremmin varmistamaan toimitettavien ratkaisujen laatu. Merkittävin ehdotus on testaustoiminnon eriyttäminen omaksi tiimikseen, jossa testauksen suorittaisi testauksen ammattilaiset. Erillisen testaustiimin suorittama testaus olisi nykyisen käytännön mukaista testausta objektiivisempaa. Mikäli testausta ei voida eriyttää omaan tiimiinsä, tulisi testausta suorittaville henkilöille järjestää testauskoulutusta. Lisäksi testauksen prosesseja tulisi yhtenäistää projekteissa ja testaussuunnitelman tekeminen tulisi aloittaa jo aikaisessa vaiheessa.

Tutkimuksen luotettavuutta mitataan validiteetilla ja reliabiliteetilla. Tutkimuksen validiteetilla arvioidaan sen pätevyys, eli ovatko saadut tulokset oikeita. Reliabiliteetilla tarkoitetaan tutkimuksen tulosten toistettavuutta. (Saaranen-Kauppinen & Puusniekka 2006.)

Tutkimusta voidaan pitää pätevänä, koska haastatellut henkilöt ovat kokeneita useissa tietojärjestelmähankkeissa mukana olleita henkilöitä. Koska heidän henkilöisyytensä ja edustamansa yritys ei selviä tutkimuksesta, voidaan olettaa heidän puhuneen haastattelutilanteessa vapaasti omista kokemuksistaan. Haastateltavat, joita oli yhteensä kuusi, valittiin yhdessä Yritys X:n edustajan kanssa. Haastattelut

järjestettiin yhtä lukuun ottamatta puhelimitse, joten haastateltu sai olla tutussa ympäristössä. Puhelimitse järjestetyt puhelinpalaverit ovat alalla hyvin yleisiä. Haastattelut äänitettiin ja litteroitiin ja pyrittiin analysoimaan tarkasti. Tulosten uskotaan olevan toistettavissa, mikäli tutkimus tehtäisiin uudelleen. Kokonaisuutena arvioiden aineiston luotettavuutta ja toistettavuutta voidaan pitää hyvänä.

Tutkimuksessa keskityttiin toimitusprojektin aikaiseen testaukseen. Regressiotestausta ei juurikaan käsitelty, vaikka se on toimitusprojekteissa erittäin tärkeä testauksen prosessi. Jatkotutkimusaiheena voitaisiin tutkia, kuinka regressiotestauksen automatisointia voitaisiin kehittää ja tuotteistaa.

## LÄHTEET

### KIRJALLISET LÄHTEET

Burnstein, I. 2002. Practical Software Testing – A process oriented approach. New York, NY, Yhdysvallat: Springer Science+Business Media, Inc.

Dunkinson, K. & Birch, A. 2013. Implementing Microsoft Dynamics AX 2012 with Sure Step 2012. Birmingham, Iso-Britannia: Packt Publishing Ltd.

Grönroos, C. 2010. Palvelujen johtaminen ja markkinointi. Helsinki: WSOYPro Oy.

ISO/IEC 29119-2. 2010. Software and Systems Engineering – Software Testing – Part 2: Test Process. Geneva, Sveitsi: International Organization for Standardization.

ISO/IEC 25010. 2011. Systems and software engineering - Systems and software Quality Requirements and Evaluation - Systems and software quality models. Geneva, Sveitsi: International Organization for Standardization.

Kasurinen, J. 2011. Software Test Process Development. Lappeenranta: Lappeenrannan teknillinen yliopisto.

Lehtinen, U. & Niinimäki, S. 2005. Asiantuntijapalvelut – tuotteistamisen ja markkinoinnin suunnittelu. Helsinki: WSOY.

Lyytinen, K. & Hirschheim, R. 1987. Information systems failures – a survey and classification of the empirical literature. Oxford Surveys in Information Technology, Vol. 4, 257-309. Oxford, Iso-Britannia: Oxford University Press.

Monk, E. & Wagner, B. 2007. Concepts in Enterprise Resource Planning. 3. painos. Boston, MA, Yhdysvallat: Course Technology Cengage Learning.

Myers, G. 2004. The Art of Software Testing. New Jersey, Yhdysvallat: John Wiley & Sons, Inc.

Naik, K. & Tripathy, P. 2008. Software testing and quality assurance: Theory and practice. New Jersey, Yhdysvallat: John Wiley & Sons, Inc.

Sipilä, J. 1996. Asiantuntijapalvelujen tuotteistaminen. Helsinki: WSOY.

Tiensuu, V. & Kari, T. 2006. Työvälineitä tuotteistamiseen. Lahti: Lahden Ammattikorkeakoulu.

Tyrväinen, P. 2003. Ohjelmistojen tuotteistaminen. Teoksessa Hyvönen, E. (toim.) Ohjelmistoliiketoiminta. Helsinki: WSOY.

#### ELEKTRONISET LÄHTEET

BSE Consulting. 2014. Méthodologie. [viitattu 06.09.2014]. Saatavissa:

<http://www.bse-c.co.kr/fr/company/methodology>

Gupta, A. 2013. Microsoft Dynamics Sure Step Training. [viitattu 16.10.2014].

Saatavissa: <http://www.slideshare.net/guptaamarnath/arvato-sure-steptraining>

International Software Testing Qualifications Board. 2007. ISTQB:n testaussanasto. [viitattu 29.06.2014]. Saatavissa:

[http://fistb.ttlry.mearra.com/sites/fistb.ttlry.mearra.com/files/istqb\\_sanasto.pdf](http://fistb.ttlry.mearra.com/sites/fistb.ttlry.mearra.com/files/istqb_sanasto.pdf)

Legge, D. 2011. ERP Testing Methodology Approach and Leading Practices. Hitachi Consulting Corporation. [viitattu 07.06.2014]. Saatavissa:

[http://www.hitachiconsulting.com/files/pdfRepository/WP\\_Best\\_Practices\\_in\\_ERP\\_Testing.pdf](http://www.hitachiconsulting.com/files/pdfRepository/WP_Best_Practices_in_ERP_Testing.pdf)

Microsoft. 2014. Microsoft Dynamics ERP. [viitattu 08.06.2014]. Saatavissa:

<http://www.microsoft.com/fi-fi/dynamics/erp.aspx>

Palmquist, S., Lapham, M., Miller, S., Chick, T. & Ozkaya, I. 2013. Parallel Worlds: Agile and Waterfall Differences and Similarities. [viitattu 18.10.2014].

Saatavissa:

[https://resources.sei.cmu.edu/asset\\_files/TechnicalNote/2013\\_004\\_001\\_62918.pdf](https://resources.sei.cmu.edu/asset_files/TechnicalNote/2013_004_001_62918.pdf)

Prodman. 2013. Mitä tuotteistus tarkoittaa? [viitattu 11.8.2014]. Saatavissa:

<http://prodman.fi/mita-on-tuotteistus>



Saaranen – Kauppinen, A & Puusniekka, A. 2006. KvaliMOTV - Menetelmäopetuksen tietovaranto [verkkojulkaisu]. Tampere: Yhteiskuntatieteellinen tietoarkisto. [viitattu 17.10.2014]. Saatavissa:

<http://www.fsd.uta.fi/menetelmaopetus/kvali/index.html>

Sycor GmbH. 2014. From Axapta 1.0 to Microsoft Dynamics AX 2012. [viitattu 16.10.2014]. Saatavissa: [http://en.sycor-group.com/dynamics-](http://en.sycor-group.com/dynamics-ax/solutions/dynamics-ax/history/#.VEAi9WeSx8E)

[ax/solutions/dynamics-ax/history/#.VEAi9WeSx8E](http://en.sycor-group.com/dynamics-ax/solutions/dynamics-ax/history/#.VEAi9WeSx8E)

Thorley, L. 2014. What is ERP? Where Does This Term Come From and Can It Help My Business? [viitattu 17.10.2014]. Saatavissa:

<http://www.erpsoftwareblog.com/2014/09/erp-term-come-can-help-business/>

Tietojärjestelmien hankinta Suomessa 2013 Tutkimusraportti. 2013. Celkee Oy & Tietotekniikan liitto ry. & Ohjelmistoyrittäjät ry. [viitattu 08.06.2014]. Saatavissa:

<http://www.ttlry.fi/sites/ttl.ttlry.mearra.com/files/Tietoj%C3%A4rjestelmien%20hankinta%20Suomessa%202013.pdf>

Wailgum, T. 2008. ERP Definitions and Solutions. CXO Media Inc. [viitattu 07.06.2014]. Saatavissa:

[http://www.cio.com/article/40323/ERP\\_Definition\\_and\\_Solutions](http://www.cio.com/article/40323/ERP_Definition_and_Solutions)

## SUULLISET LÄHTEET

Asiakasyritys A:n edustaja. Haastattelu 29.9.2014

Toimittajan edustaja. Haastattelu 30.9.2014

Asiakasyritys B:n edustaja. Haastattelu 1.10.2014

Toimittajan edustaja. Haastattelu 1.10.2014

Toimittajan edustaja. Haastattelu 21.10.2014

Asiakasyritys C:n edustaja. Haastattelu 22.10.2014