

Antti Hartala

OpenERP-tietojärjestelmän käyttöönoton suunnittelu sähköalan pienyrityksessä

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Sähkötekniikka

Insinöörityö

20.5.2014

Tekijä Otsikko Sivumäärä Aika	Antti Hartala OpenERP-tietojärjestelmän käyttöönoton suunnittelu sähköalan pienyrityksessä 39 sivua + 5 liitettä 20.5.2014
Tutkinto	insinööri (AMK)
Koulutusohjelma	sähkötekniikka
Suuntautumisvaihtoehto	sähkövoimatekniikka
Ohjaajat	työnjohtaja Teemu Autio lehtori Tuomo Heikkinen
<p>Tämän insinööriyön tavoitteena oli tehdä selvitystä ja suunnittelua OpenERP-toiminnanohjausjärjestelmän käyttöönottoprojektia varten, pienen sähköurakointiliikkeen toimeksiannosta. Työssä haettiin mahdollisimman pragmaattista lähestymistapaa käyttöönotossa esiintyviin ongelmiin ja joihinkin kohdeyrityksen järjestelmälle asettamiin erityisvaatimuksiin. Pääpaino oli selvittää, millä tavoin erityisesti suomalainen sähköurakointiliike hyötyy toiminnanohjausjärjestelmän käyttöönotosta, ja kuinka paljon aikaa ja räätelöintiä vaaditaan toimivan OpenERP-järjestelmäkokonaisuuden rakentamiseen pienyrityksen rajattuun erityiskäyttöön.</p> <p>Toiminnanohjausjärjestelmäksi päätettiin valita OpenERP. Kyseessä on avoimen lähdekoodin järjestelmä, jonka kehitysyhteisö on maailmalla laaja. OpenERP:tä koskevaa englanninkielistä materiaalia löytyy paljon verkosta, ohjeita perustason käytöstä aina vaativaan järjestelmän mukauttamiseen.</p> <p>Varsinainen käyttöönottoprojekti on hieman viivästynyt alkuperäisestä tavoitteesta, johtuen kohdatuista erinäisistä haasteista, mutta tämän selvitystyön valmistuttua sitä voidaan taas täysipainoisesti jatkaa.</p> <p>Tämä työ toimii toivottavasti hyvänä pohjana OpenERP-käyttöönottoprojektin tarpeisiin, ja sitä voidaan varmasti käyttää apuna ohjedokumentoinnin laatimiseen järjestelmän peruskäyttäjille sekä järjestelmänvalvojille.</p>	
Avainsanat	OpenERP, ERP, toiminnanohjausjärjestelmä, käyttöönotto

Author Title Number of Pages Date	Antti Hartala OpenERP – Planning of Deployment on Small Scale Electrical Contracting Business 39 pages + 5 appendices 20 May 2014
Degree	Bachelor of Engineering
Degree Programme	Electrical Engineering
Specialisation option	Electrical Power Engineering
Instructors	Teemu Autio, Project Manager Tuomo Heikkinen, Senior Lecturer
<p>The aim of this study was to make an analysis and plan an ongoing deployment project of the OpenERP-system. The work was carried out as an assignment for a small scale Finnish electrical contracting company. This study was intended to seek practical solutions for problems during deployment and fulfill some special system requirements placed by the company. The main goal was to clarify how especially Finnish electrical contracting business can benefit from using ERP-systems and how much time and modifications are required in developing a working OpenERP-system environment for specific small business use.</p> <p>The ERP-system selected for the deployment project is OpenERP. It is an open source system with an extensive worldwide developing community. Plenty of online material about OpenERP is available in English. These guides and tutorials range from covering basic use to highly advanced modification tasks of the system.</p> <p>The OpenERP deployment project was delayed slightly from the original target due to some challenges demanding special attention. However, finishing this study and extracting all the information and benefits from its results will assist in eventually finishing the project successfully.</p> <p>This study will hopefully act as a good foundation for the OpenERP deployment project and it can certainly be used as a base for instructions intended for basic users and administrators of the system.</p>	
Keywords	OpenERP, ERP, deployment

Sisällys

Lyhenteet ja käsitteet

1	Johdanto	1
1.1	ERP	2
1.2	Avoimen lähdekoodin ERP-järjestelmät ja OpenERP	3
1.3	Premium Sähkö Oy	6
2	Keskeiset tekijät sähköurakoinnissa	7
2.1	Toimintaprosessi	7
2.2	Hallinnollinen toimintaprosessi	9
3	OpenERP	11
3.1	Järjestelmään kirjautuminen	12
3.2	CRM	12
3.3	Projektinhallinta	13
3.4	Varastonhallinta	16
3.4.1	Varastotyytit	17
3.4.2	Varastonhallinnan työkalut	17
3.5	Sähkötuotteet	18
3.6	Palvelimen perustaminen	20
3.7	Järjestelmän laajennukset ja toimintaympäristöt	21
3.8	Järjestelmän räätälöinnin perusteet	22
3.8.1	Kehittäjätila	22
3.8.2	Räätälöinti moduulipohjaisesti	23
3.8.3	Python ja XML	23
3.8.4	Moduulin perusrakenne	23
3.8.5	Moduulien periytyminen	25
3.8.6	Esimerkkejä	26
3.9	Raportointi ja raporttien muokkaaminen	29
3.10	Kielen käännökset	31
4	Käyttöönoton suunnittelu	32
4.1	Avainkohdat	32
4.2	Kustannukset	33
4.3	Projektin tilanne ja kohdatut ongelmat	33
4.4	Toteutettuja muokkauksia	34

4.5	Järjestelmän testaaminen esimerkkityötilauksen avulla	35
5	Yhteenveto	36
	Lähteet	38
	Liitteet	
	Liite 1. Projektisuunnitelma	
	Liite 2. OpenERP-aloituspalaveri 3.10.2013	
	Liite 3. OpenERP-palaveri 16.1.2014	
	Liite 4. OpenERP-palaveri 3.2.2014	
	Liite 5. Esimerkkikuvia OpenERP-järjestelmästä	

Lyhenteet ja käsitteet

CRM	<i>Customer Relationship Management</i> . Asiakassuhteiden hallinta.
CSV	<i>Comma-separated values</i> . Taulukkomuotoinen tiedosto.
ERP	<i>Enterprise Resource Planning</i> . Yrityksen toiminnanohjaus.
Kanban	Kanban-näkymä. Visuaalinen luettelonäkymä.
Moduuli	Järjestelmän tai ohjelmiston osa, joka sisältää yhden tai useita toiminnallisuksia.
Pk-yritykset	Pienet ja keskisuuret yritykset.
PostgreSQL	Avoimen lähdekoodin relaatiotietokanta.
PY	Python-ohjelmointikielellä kirjoitettu tiedosto.
Python	Ohjelmointikieli.
RML	<i>Report Markup Language</i> . XML-muunnelma raporttipohjien luomiseen ja generointiin.
SQL	<i>Structured Query Language</i> . Kyselykieli, käytetään tietokantojen manipulointiin.
SXW	OpenOffice-kirjoitustiedosto.
XML	<i>Extensible Markup Language</i> . Rakenteellinen kuvaus- ja merkintäkieli.

1 Johdanto

OpenERP on avoimen lähdekoodin ERP-järjestelmä (Enterprise Resource Planning). ERP:stä voidaan käyttää sen suomenkielistä nimitystä: yrityksen toiminnanohjausjärjestelmä. Nimensä mukaisesti ERP ohjaa yrityksen juoksevia toimintoja ja lisäksi tarjoaa työkaluja tulevan toiminnan suunnitteluun.

ERP-ohjelmisto käsittää laajan tietojärjestelmäkokonaisuuden. Yhteistä eri ERP-järjestelmille on, että niiden rakenne on modulaarinen, tarkoittaen sitä, että yrityksen erilliset toiminnot kuten myynti, markkinointi ja logistiikka ovat myös järjestelmässä erillisinä moduuleina, eli osakokonaisuuksina.

Avoimen ERP-järjestelmän, kuten OpenERP, moduulit ja toiminnot ovat käyttäjän vapaasti muokattavissa. OpenERP:n kehittäjäkunta onkin kasvanut laajaksi, kun periaatteessa kuka tahansa voi vapaasti ryhtyä järjestelmän kehittäjäksi. OpenERP:n kanssa samankaltaisia, mutta suljettuja, ohjelmistoja on paljon markkinoilla, joista tunnetuimpina esimerkkeinä nousevat esiin ainakin SAP ja Oracle ERP. Verratessa OpenERP:tä näihin laajoihin ja monimutkaisiin ohjelmistokokonaisuuksiin, ensisilmäyksellä selkeimpiä eroja ovat OpenERP:n käytön helppous ja käyttöliittymän selkeys. Esimerkiksi SAP on luotu raskaan datan käsittelyyn, jota suuressa yhtiössä kertyy huomattavia määriä. Tällöin tärkeimmäksi nousee toimintojen optimointi, ei niinkään käyttöliittymäsuunnittelu.

OpenERP on enemmän suunnattu pienille ja keskisuurille yrityksille. Sen tekevät kilpailukykyiseksi edullisuus ja helppokäyttöisyys. OpenERP:n vahvuus on sen joustavuus, ja se kuuluu niihin harvoihin ERP-järjestelmiin, joita käyttää laaja-alainen pienten ja itsenäisten yritysten joukko. [1.]

Käyttäjä voi valita erilaisia moduuleja käyttöönsä tarpeen mukaan tai luoda omia moduuleita täysin vapaasti. Ainoa ehto on, kuten yleisesti avoimen lähdekoodin sovelluksissa, että uusi moduuli tulisi olla muidenkin vapaasti hyödynnettävissä. Tällä hetkellä käyttäjien suoraan ladattavissa on yli 900 moduulia. Yhteensä erilaisia käyttäjien kehitämiä moduuleita arvioidaan olevan reilusti yli 2000. [2.]

OpenERP:n kehittäjäyhtiö tekee rahaa lähinnä järjestelmän integrointi- ja räätälöintipalveluilla, jotka kattavat edistyneen ja vaativan käytön. Esimerkiksi kohdemaakohtaiset sovellukset sekä käytäntöjen ja verotuksen eroavaisuudet aiheuttavat tarvetta asiakas-kohtaiseen tuotesuunnitteluun. Mikään ei tietenkään estä asiakasta toteuttamasta itsenäisesti tarvittavia muutoksia järjestelmään. Pienen yrityksen tapauksessa se voi hyvinkin onnistua riittävällä menestyksellä.

Tässä työssä keskitytään erityisesti tutkimaan relevantteja mahdollisuuksia mukauttaa laajaa toiminnanohjausjärjestelmää pienyrityksen varsin rajattuun erityiskäyttöön. Projektia toteutetaan yhteistyössä sähköurakointiliike Premium Sähkö Oy:n kanssa, jonka tarpeisiin on tarkoitus toteuttaa järjestelmän osittainen käyttöönotto.

Premium Sähkö Oy:llä ei ole tällä hetkellä muita keskeisiä toiminnanohjaussovelluksia käytössä, joita pitäisi sovittaa yhteen OpenERP:n kanssa. Ei ole myöskään suuria valmiita tietokantoja, joiden tulisi keskustella järjestelmien välillä. Tällaiset seikat tuovat uusien järjestelmien käyttöönottoon erityisiä haasteita, joiden ratkaisemiseen tarvittaisiin jo asialle täysin omistautunut työryhmä.

1.1 ERP

ERP:n tarkoituksena on yhdistää liikeyrityksen toiminnot yhtenäisen tietojärjestelmän osiksi. Yhtenäisestä järjestelmästä seuraa huomattavia hyötyjä yritykselle, mikäli se otetaan käyttöön oikealla tavalla ja oikeassa mittakaavassa. Täysikokoinen ERP on yleinen suuryrityksissä, mutta pienemmissä organisaatioissa sen käyttö on harvinaisempaa. [3.]

Kun on olemassa yhteinen tietokanta kaikelle yrityksen sisällä kulkevalle tiedolle, voidaan helposti seurata esimerkiksi asiakastilausten etenemistä ja tarvittaessa käsitellä niitä nopeammin. Ilman ERP-järjestelmää toiminnasta puuttuu monesti tarvittava läpinäkyvyys. Tilauksia saattaa hukkua työntekijöiden sähköposteihin tai paperipinojen alle. Kukaan ei myöskään tiedä, onko tarvittavaa tuotetta varastossa, ennen kuin joku käy tarkistamassa. [4.]

Ennen yhtenäisiä ERP-järjestelmiä yrityksen eri toiminnoille oli erilliset ohjelmistot ja tietokannat. Näiden keskustelu keskenään oli hankalaa ja joskus täysin mahdotonta.

Suurin ongelma oli, että yritykset joutuivat turvautumaan useisiin järjestelmätoimittajiin, jotka kaikki toteuttivat pienen osan tarvittavasta kokonaisuudesta. Toisistaan erillisten ohjelmien rakenteet eivät olleet keskenään yhteensopivia, vaan vaadittiin monesti käyttäjältä aktiivisia toimia yhdistämään eri ohjelmien tarjoamaa tietoa keskenään. [5.]

Alla on esitetty hieman faktoja kolmesta tunnetuimmasta ERP-järjestelmätoimittajasta. [6.]

SAP

SAP on markkinaosuudella mitattuna suurin ERP-toimittaja. Sillä on asiakkaita 120 maassa. Järjestelmä on suunnattu suurille yrityksille ja se on hinnaltaan kalliimmasta päästä.

Oracle ERP

Oraclen ERP-ratkaisu on SAP:n jälkeen toiseksi käytetyin maailmalla. SAP:n tavoin järjestelmä on kallis ja lähinnä suurempien yritysten käytössä.

Microsoft Dynamics

Microsoftin ERP-ratkaisu on enemmän suunnattu pk-yrityksille. Microsoft käyttää ja tukee voimakkaasti partner-yrityksiä, jotka toimittavat järjestelmää asiakkaille. Järjestelmän hintaluokka on niin ikään maltillinen.

1.2 Avoimen lähdekoodin ERP-järjestelmät ja OpenERP

Avoimen lähdekoodin sovellusten pääperiaatteena on päästää myös ulkopuolisia mukaan sovelluskehitykseen. Avoimen sovelluksen käyttäjä voi näin itse kehittää ja muokata ohjelmistoa haluamukseen. Osa ohjelmistoista on ilmaisia, osassa on eri tavoin määräytyviä käyttömaksuja.

Avoimen ERP:n eduksi voidaan lukea halpa hankintahinta itse järjestelmälle. Se ei tosin takaa alhaisia käyttöönoton kustannuksia. Käyttöönoton kustannuksia on käsitelty laajemmin kohdassa 4.2: Kustannukset.

Muita avoimen järjestelmän etuja ovat järjestelmän mukautuvuus ja joustavuus. Mikäli avoimella järjestelmällä on tarpeeksi laaja käyttäjä- ja kehittäjäkunta, ongelmatilanteissa tukea saattaa saada jopa nopeammin kuin suljetun järjestelmän kehittäjiltä. [7.]

Avoimen ERP-järjestelmän käyttö vaatii kuitenkin käyttäjältä enemmän. Järjestelmän versio usein vaihtuu nopeaan tahtiin ja samalla osa sen toiminnallisuuksista saattaa muuttua tai poistua kokonaan uuden version myötä. Vanhaa versiota ei myöskään enää tueta aktiivisesti, vaan käyttäjäyhteisö siirtyy uuteen versioon heti kun se julkaistaan. Käyttäjän täytyy siis aktiivisesti seurata päivitystilannetta. [5.]

OpenERP-projektin, joka aluksi tunnettiin nimellä TinyERP, aloitti belgialainen opiskelija Fabien Pinckaers vuonna 2002. Virallisesti OpenERP:stä tuli avoimen lähdekoodin järjestelmä vuonna 2005, ja sen käyttö- ja kehitysyhteisö laajeni nopeasti. Nykyään Pinckaers on OpenERP Group -nimeä kantavan yhtiön pääjohtaja. Yhtiö vastaa versionhallinnasta ja tarjoaa lisäpalveluita, räätälöintejä, koulutuspalveluita ja käyttöönottoprojekteja muiden järjestelmään vihkiytyneiden, ns. partner-yritysten ohella. [8.]

OpenERP Group tarjoaa verkkosivuillaan virallista OpenERP Enterprise -tuotekokonaisuutta kirjoitushetkellä perushintaan 35 €/kk jokaista järjestelmään kirjautunutta käyttäjää kohden. Hintaan sisältyy useita tukipalveluita ja ylläpito pilvipalveluna. Kantavana ideana markkinoinnissa on ns. ”valmis tuote suoraan laatikosta” ja sen edut asiakasyrityksille, verrattuna yritysten itsenäiseen järjestelmän käyttöönottoon ja oman palvelimen ylläpitoon. Ylläpidon ja päivitysten luvataan olevan automaattisia, mutta perushintaan ei luonnollisesti sisälly konsultointi- ja räätälöintipalveluja. [9.]

Tällä hetkellä käytössä oleva vakaa versio on OpenERP 7.0. Monet verkosta löytyvät ohjeet on suunnattu edelliselle pääversiolle 6.0/6.1, mutta ne ovat usein sovellettavissa myös uusimpaan versioon.

Avoimia ERP-järjestelmiä on markkinoilla paljon muitakin. Tämän työn tarkoituksena ei ole kuitenkaan verrata avoimia järjestelmiä toisiinsa, eikä vastaavia suljettuja, sillä päätös OpenERP:n käytöstä on kirjoitushetkellä jo tehty. Seuraavassa osiossa on kuitenkin esitetty lyhyesti esimerkkejä muista avoimista ERP-järjestelmistä. [10.]

Openbravo

Openbravo käyttää tällä hetkellä yli 6000 erikokoista organisaatiota useilla eri teollisuudenaloilla. Järjestelmä toimii kaikilla web-selaimilla. Vaikka Openbravo on avoin ohjelmisto, sen käyttöön tarvitsee ostaa lisenssi.

Apache OFBiz

Apache OFBiz ERP -järjestelmä on ilmainen. Kun se on asennettu, sitä voi käyttää web-selaimen kautta. Tässä suhteessa samanlainen ratkaisu kuin OpenERP.

Compiere

Compiere ERP -järjestelmää käytetään yli 60 maassa. Sitä voi käyttää lokaalisti tai pilvipalvelun kautta. Compiere:stä on tarjolla ilmainen rajoitettu versio, mutta täysi versio on maksullinen.

WebERP

WebERP on ilmainen ja se toimii kokonaisuudessaan verkossa. Sitä voi käyttää kaikilla web-selaimilla. Käyttöliittymä on yksinkertainen ja tehokas, mutta ei ehkä kuitenkaan riittävä laajemman kokoluokan toimijoille.

ERP5

ERP5 on ladattavissa ilmaiseksi, ja se sisältää paljon toimintoja. Järjestelmä ei ole kuitenkaan vielä laajassa käytössä, ja dokumentaatio on puutteellista.

Opentaps

Opentaps on tehokas ERP, jota voidaan käyttää suoraan mobiililaitteilla. Professional-versio on maksullinen.

Dolibarr

Dolibarr ERP -järjestelmää voi käyttää lokaalisti, omalla palvelimella tai siihen varatulla verkkopalvelimella. Perusversio on ilmainen, siihen voi halutessaan ostaa lisämoduuleita ja muita laajennoksia.

ADempiere

ADempieren kehitys erkaantui Compieresta, ja siitä tuli aidosti avoimeen lähdekoodiin perustuva, toisin kuin Compiere, jota hyödynnetään myös kaupallisesti.

PostBooks

Postbooks on ilmaisversio xTuple ERP -järjestelmästä. Samaan avoimeen lähdekoodiin perustuu kuitenkin myös kaupallisia sovelluksia.

1.3 Premium Sähkö Oy

Premium Sähkö Oy on tuore vuonna 2012 perustettu yritys, joka työllistää tällä hetkellä 6 henkeä. Kyseessä on tytäryhtiö, jonka omistavat puoliksi Teemu Autio ja Kari Pääkönen. Premium Sähkö Oy on osa Aution emoyhtiötä nimeltä TMA Solutions Oy, jonka palveluksessa on tällä hetkellä yhteensä 30 henkilöä. [11.]

Yrityksen kasvu on ollut nopeaa, n. 300 % vuodessa. Tällä hetkellä tytäryhtiön liikevaihto on n. 300 k€. Emoyhtiön liikevaihto on 1,2 M€. Premium Sähkö Oy:n tavoite on nostaa liikevaihto samalle tasolle emoyhtiön kanssa 3–5 vuoden sisällä. Yrityksen nopeat kasvuluvut ja ajankäytön hallinnassa ilmenneet haasteet puhuvat vahvasti sähköisen toiminnanohjausjärjestelmän käyttöönoton puolesta.

OpenERP:n käyttöönotto Premium Sähkö Oy:ssä on tarkoitus olla aluksi pienen skaalan kokeilu, jonka onnistuessa hyvin, voidaan laajentaa myöhemmin koko yhtiön käsitteväksi järjestelmäksi.

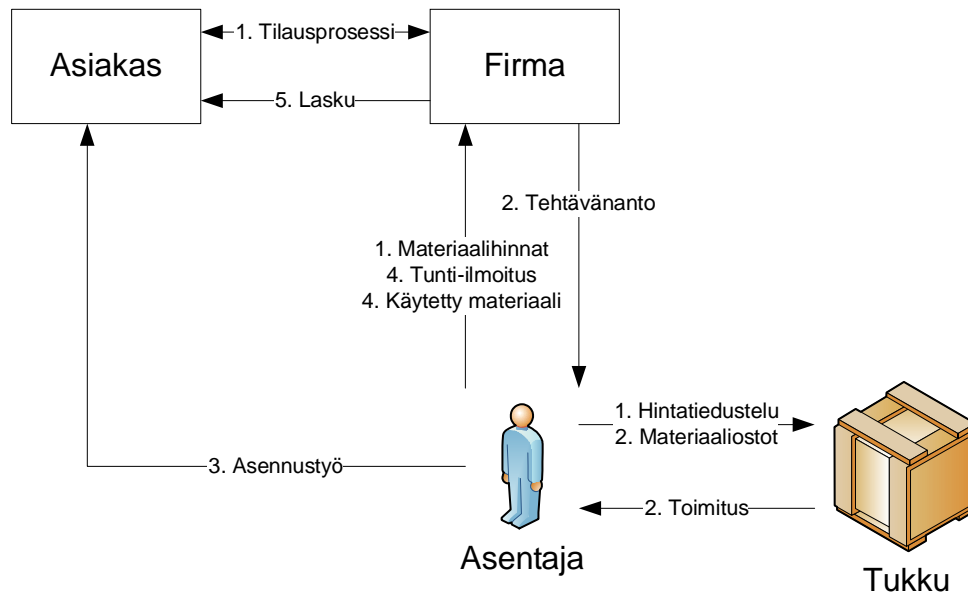
2 Keskeiset tekijät sähköurakoinnissa

Sähköurakointiliike myy ensisijaisesti asennus- ja korjauspalveluja sekä pienessä määrin toimittaa komponentteja asiakkaille. Pienen sähköurakointiliikkeen urakat ovat enimmäkseen lyhyitä, päivän kestäviä työkeikkoja, joten ei tarvita tarjousvaihetta tai laskutuserien sopimista kuten isomman kokoluokan urakoissa on tapana tehdä. Tärkeimmät alkuvaiheen työkalut ovat ainakin perustason asiakashallinta sekä projektinhallinta.

Varastohallinnan työkalut eivät ole heti alkuvaiheessa välttämättömiä. Premium Sähkö Oy:ssä on pieni kiinteä varasto sekä kaksi asentajien ajoneuvoa, jotka sisältävät kaapelikeloja ja muuta työssä usein käytettävää materiaalia. Usein asentajat käyvät ostamassa tarvittavat tuotteet suoraan tukusta autoon, minkä jälkeen ajetaan kohteeseen ja suoritetaan tuotteiden asennustyö asiakaskohteessa. Nämä tuotteet eivät varsinaisesti ehdi käydä varastossa, eikä niiden kirjaamiselle olisi muutenkaan perusteita. Kiinteään varastoon ei myöskään haluta sitoa liikaa pääomaa, joten sen käyttö on rajoitettua yrityksen jokapäiväisessä toiminnassa.

2.1 Toimintaprosessi

Alla olevassa kuvassa 1 on pyritty esittämään Premium Sähkö Oy:n tämänhetkistä toimintaprosessia.



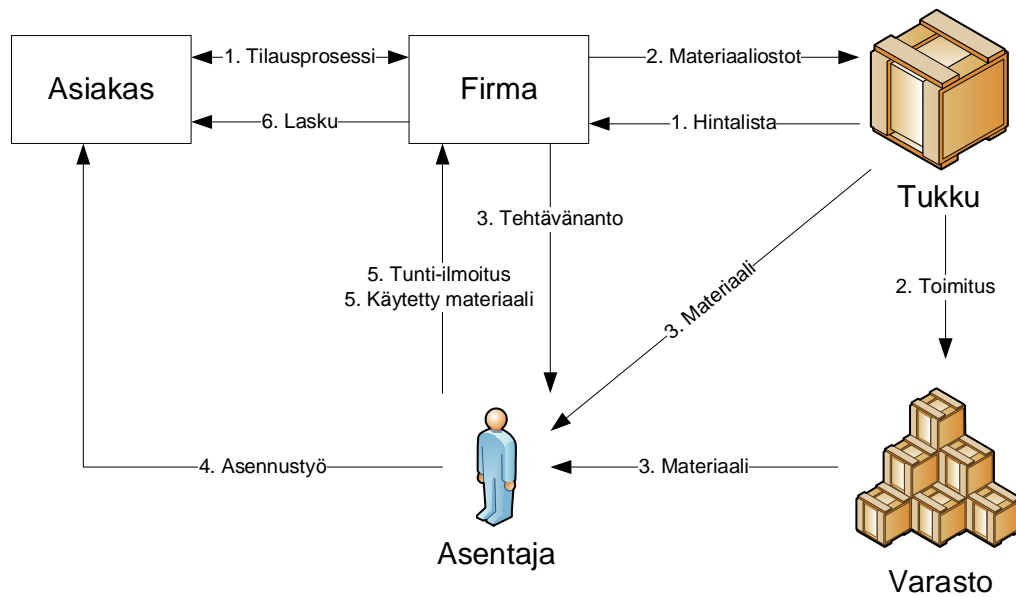
Kuva 1. Premium Sähkö Oy:n toimintaprosessi

Kuvasta voidaan havaita, että suurin ongelma asentajien suorassa tukkuasioinnissa on se, että ostettujen tuotteiden hinnat tiedetään vasta kun ostokset on jo tehty. Usein asentajat joutuvat tiedustelemaan erikseen työssä tarvittavien tuotteiden hintoja tukusta ja ilmoittamaan ne eteenpäin käsittelijöille.

Jos halutaan tiedustella yleisesti hintoja etukäteen, tukusta saa hintalistan, jossa on kaikki tuotteet lueteltuna. Tällaisen suuren listan läpikäyminen on kuitenkin hidasta. Nopeasti saatavilla olevat tarkat hintatiedot helpottaisivat asiakkaan kanssa käytävää tilausprosessia.

OpenERP mahdollistaa laajemman toimintaprosessin käyttöönoton. Yhtenäinen järjestelmä takaa sen, että kaikki relevantti tieto kulkee keskuskonttorin kautta ilman erillisiä kyselyitä työntekijätasolta, mikä tällä hetkellä vie työaikaa molemmilta osapuolilta.

Seuraava kuva 2 havainnollistaa kehittyneempää ja joustavampaa toimintaprosessia, jonka toiminnanohjausjärjestelmän käyttöönotto voisi saada aikaan.

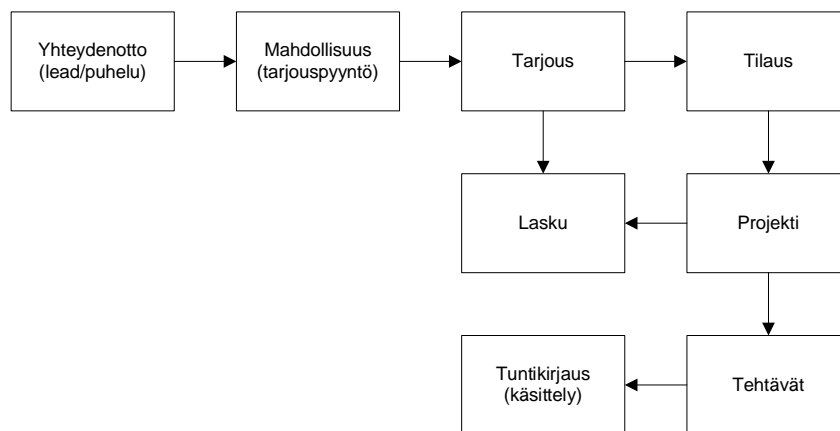


Kuva 2. Laaja toimintaprosessi

Uudessa prosessissa on olemassa varasto osalle materiaalia, mutta on myös mahdollista entiseen tapaan hakea työn vaatima materiaali suoraan tukusta. Ostot kuitenkin suoritettaisiin aina järjestelmän avulla.

2.2 Hallinnollinen toimintaprosessi

Alla olevassa kuvassa 3 on pyritty esittämään tilausvaiheen prosessin etenemistä. Asiakkaan loppulasku muodostuu kuitenkin vasta kun työ on tehty. Tarjousvaiheessa esitetään ainoastaan hinta arvio.



Kuva 3. Projektin eteneminen

Loppulaskutuksen tapaan myös asiakkaan kanssa käytävään tarjous-tilaus-prosessiin vaikuttavat erilaiset tuotepaketit, joita voidaan helposti lisätä järjestelmään tarpeen mukaan. Valmiit tuotepaketit helpottavat arviohinnan määrittystä. Esimerkiksi osalle sähkötuotteista voisi määrittää asennuskulut erikseen ja sitoa ne yhteen erillisiksi asennustyötuotteiksi. Pienet usein toistuvat työtehtävät, kuten esim. pistorasioiden asennus, voidaan määrittää helposti omiksi kokonaistuotteiksi, joita käytetään sellaisenaan tai tarvittaessa kerrannaisina työn kokonaishinnan määrittelyyn.

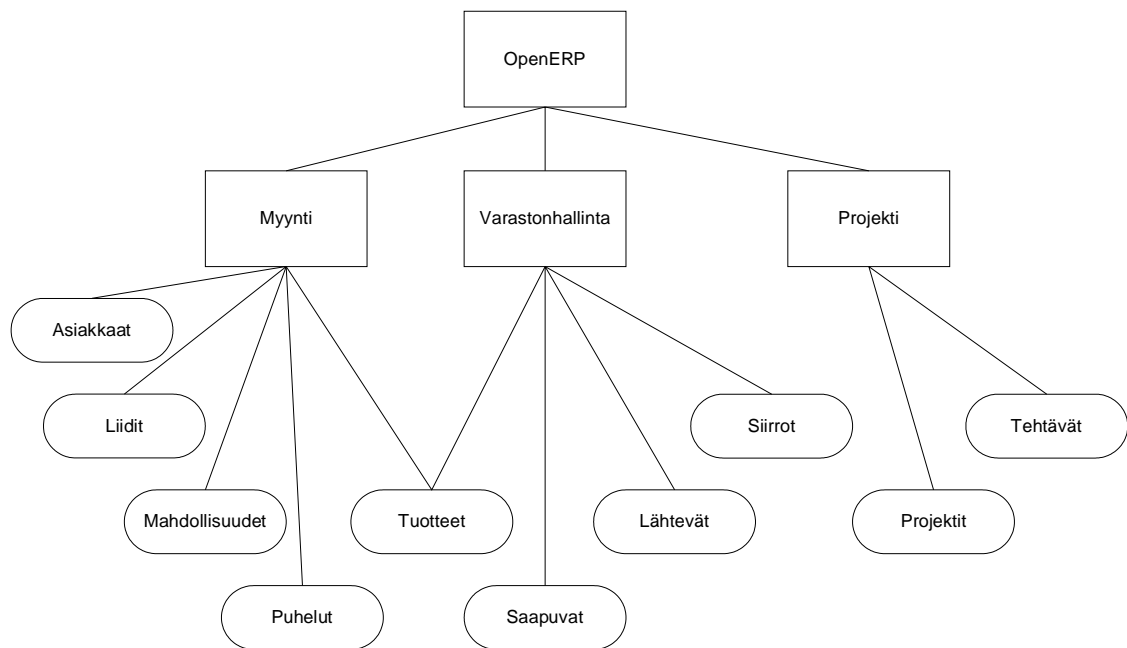
Tarjouksen määrittelyyn vaikuttavat tekijät, kun käytetään tuotepaketteja kokonaishinnan määrittelyssä:

- kokonaistuotepaketit (pienet urakat)
- materiaali (asennustyö arvioitava)
- asennustyötuotteet (vakioasennustunnit)
- suunnittelu (tuntiperusteinen/arvio)
- käsittelykulut (tuntiperusteinen/arvio)
- muut kulukorvaustuotteet.

Tuntiperusteiset hinnat ovat arvioita, jotka perustuvat järjestelmään talletettuun dataan aikaisemmista saman kokoluokan projekteista. Käsittelykulut voivat koostua myös sovitusta vakiohinnoista. Todelliset tilausten käsittelykulut on tietenkin tarkoitus pienentää järjestelmän käytöllä minimiin.

3 OpenERP

OpenERP tarjoaa laajan kirjon toiminnallisuuksia, joita voidaan rajata käyttäjän tarpeiden mukaan. Alla olevassa kuvassa 4 on esitetty kolmen keskeisen toiminnallisuuden muodostama valikkorakenne. Kaikki käyttäjät eivät kuitenkaan tule saamaan oikeuksia jokaiseen järjestelmän toimintoon, ainoastaan omaa työtehtävää sivuaviin. Esimerkiksi sähköasentajat eivät tavallisesti tarvitse käyttöoikeutta myynti-osion toimintoihin.



Kuva 4. OpenERP-valikkorakenne

Kuvasta voi havaita, miten tuotevalikkoon pääsee sekä myynnin että varastonhallinnan päävalikkojen kautta. Sen sijaan kuvasta ei ilmene esimerkiksi asiakkaat- ja tehtävät-alavalikkojen välistä yhteyttä.

OpenERP:n tärkeimmät ja useimmin käytetyt moduulit [7]:

- Asiakashallinta (CRM)
- Projektinhallinta
- Varastonhallinta
- Myynti (tarjoukset, laskutus).

3.1 Järjestelmään kirjautuminen

Jokainen uusi OpenERP:n käyttäjä saa kutsun sähköpostiin, josta avautuvasta linkistä pääsee kirjautumissivulle (Liite 5: kuva 1). Tähän asetetaan oma henkilökohtainen salasana. Samanlainen kutsu lähetetään, jos on myöhemmin tarvetta vaihtaa salasana. Ensimmäisen kirjautumisen jälkeen käytetään normaalia kirjautumissivua.

3.2 CRM

CRM-työkaluja voidaan ottaa käyttöön ensimmäisinä, koska ne ovat varsin valmiita sellaisenaan. Asiakashallinnan perustarpeet eivät eroa oleellisesti yrityksestä tai maasta toiseen mentäessä. Työtä nopeuttaisi oleellisesti järjestelmän integrointi yrityksen sähköpostiin sekä yrityksen internet-sivuilta löytyvään tilaus- ja tiedustelulomakkeeseen.

Asiakkaat

Tärkein yrityksen tietokanta on aina asiakastietokanta. Kun halutaan lisätä uusi asiakas, avataan asiakaslomake (Liite 5: kuva 2), johon voi asettaa keskeisimmät asiakkaan tiedot (ainoastaan nimi pakollinen).

Liidit

Liidit ovat myyntimahdollisuuksia, joita ei ole vahvistettu. Esim. asiakkaan ensimmäinen yhteydenotto voidaan kirjata liidiksi (Liite 5: kuva 3). Jos kyseessä oleva asiakas löytyy valmiiksi asiakasrekisteristä, voidaan se valita asiakas-kenttään. Jos asiakasta ei löydy valmiiksi rekisteristä, voidaan joko kirjata asiakkaan nimi liidin nimi-kenttään, tai luoda lennosta uusi asiakas liidin asiakas-kentästä.

Kun liidi vahvistetaan, se muutetaan mahdollisuudeksi painamalla Muunna mahdollisuudeksi -painiketta (Liite 5: kuva 4). Tästä avautuu ikkuna, jossa voi päättää luodaanko liidin pojalta uusi mahdollisuus vai yhdistetäänkö vanhaan mahdollisuuteen. Liidin pohjalta voidaan myös tarvittaessa luoda uusi asiakas.

Tehokkain tapa käyttää liidejä olisi suora yhteys asiakassähköpostiin, jolloin järjestelmä luo liidin jokaisesta yhteydenotosta. Neuvottelujen jälkeen liideistä on helppo luoda mahdollisuuksia tai vaihtoehtoisesti ne voi hylätä, mikäli kauppa ei toteudu.

Mahdollisuudet

Mahdollisuudet ovat alustavasti sovittuja kauppoja, jotka käyvät läpi erinäisiä vaiheita, kunnes mahdollisuus joko voitetaan tai hävitään. Vaiheita ei tarvitse käydä aina kaikkia läpi, vaan pienimuotoisessa kaupassa voidaan mahdollisuus voittaa vaikka heti. Tällöin kauppojen hylkääminen tehtäisiin jo liidi-vaiheessa.

Puheluloki

Puhelulokiin kirjataan kaikki vastaanotetut puhelinsoitot. Puhelulokin järjestelmällinen käyttö on tärkeää, koska puheluista ei automaattisesti jää jälkiä, kuten esim. sähköpostista.

Uuden puhelun kirjaus on nopea ja yksinkertainen prosessi. Kun uusi lokimerkintä luodaan, listaan aukeaa uusi tietue, johon lisätään lyhyt kuvaus puhelusta sekä soittajan ja vastaajan tiedot (Liite 5: kuva 5).

Puhelulokimerkinnästä voidaan luoda mahdollisuus tai tarvittaessa aikatauluttaa uusi puhelu.

Suunnitellut puhelut

Aikataulutetut puhelinsoitot voidaan lisätä manuaalisesti, tai luoda suoraan useista lähteistä: liideistä, mahdollisuuksista tai puhelulokista (Liite 5: kuva 6).

3.3 Projektinhallinta

Projektinhallinta-moduuli koostuu kahdesta päärakenneosasesta: projekteista ja tehtävistä. Moduuli on suunniteltu suurempia projekteja silmälläpitäen, joten sen käyttö sellaisenaan ei ole tässä tapauksessa tarkoituksenmukaista. Sen sijaan, että avattaisiin jokaiselle tilaukselle/työkeikalle oma projekti, parempi ratkaisu on avata yksi kokonais-

projekti esim. vuodelle 2014, jonka alle avataan oma tehtävä jokaiselle yksittäiselle työkeikalle.

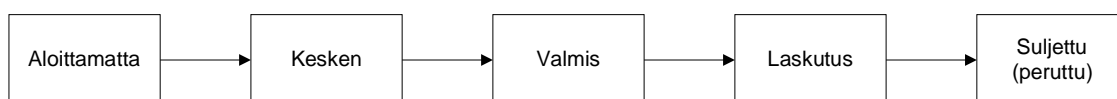
Tehtävä-yksikön rakenne on yksinkertainen, se sisältää perustiedot ja selitteen sekä tarvittaessa tietoa tehtävän suorittamiseen käytetystä työajasta. Työtunteja ja materiaalia varten tarvitaan laajennos tehtävärakenteeseen, joka mahdollistaa tuotepohjaisen tunti-ilmoituksen ja lisäksi myöhemmin tehtäväpohjaisen laskutuksen. Esimerkki tuotelistan lisäämisestä tehtävärakenteeseen on esitetty kohdassa 3.8.6: Esimerkkejä.

Vaikka kirjoitushetkellä ei olekaan tarvetta laajempien urakoiden hallinnalle, voidaan projektirakennetta yhtä hyvin käyttää myös siihen. Avataan suurelle erillisprojektille oma projekti ja käytetään tehtäviä osakokonaisuuksien suunnitteluun.

Premium Sähkö Oy:ssä halutaan ottaa toiminnanohjausjärjestelmä käyttöön ensisijaisesti pieniä keikkamuotoisia töitä tukemaan, joista kuitenkin suurin osa yrityksen tämänhetkisestä kassavirrasta koostuu.

Tehtävän perusvaiheet

Tehtävien vaiheet ovat projektikohtaisesti määriteltävissä. Kaikilla tietyn kokonaisprojektin alaisilla tehtävillä on siis aina samat perusvaiheet käytössä.



Kuva 5. Tehtävän vaiheet

Tehtävärakenteeseen kaavailut perusvaiheet on esitetty yllä olevassa kuvassa 5. Tehtävän valmis-tilasta voi tarvittaessa palata kesken-tilaan. Laskutus-tilasta ei voi enää palata taaksepäin. Suljettu-tilasta voi palata ainoastaan edelliseen tilaan (laskutus). Jos tehtävä halutaan perua, se tulee tehdä ennen laskutus-tilaan siirtymistä. Laskutus-tilassa tehtävän peruminen, eli siirtyminen peruttu-tilaan, ei enää ole mahdollista. Perutun tehtävän voi halutessaan aktivoida uudelleen, jolloin siirrytään takaisin kesken-tilaan.

Tarvittaessa tehtävärakenteeseen voidaan lisätä vaiheita käsin järjestelmän kautta. Voidaan esimerkiksi vielä lisätä odottaa-vaihe, joka ilmaisee, että työt ovat jäissä, mutta niitä aiotaan vielä jatkaa.

Tunti-ilmoitus

Pienten keikkatöiden osalta pätee, että tarkkaa hintaa ei sovita etukäteen, vaan työn valmistuttua asiakas kuittaa tunti-ilmoituksen, jossa on myös ilmoitettu käytetty materiaali. Asentaja lähettää tunti-ilmoituksen työnjohtajalle, joka hoitaa laskituksen ja muut jälkikäsitteilytehtävät. Paperinen tunti-ilmoitus ei tietenkään ole optimiratkaisu, se pitää toimittaa pääkonttoriin, jolloin niitä tulee usein kerralla iso nippu. Tämä aiheuttaa pullonkauloja käsittelyyn, kun yhdessä hetkessä pitäisi hoitaa useamman pienen keikan laskutus. Laskituksen viivästyminen vaikuttaa rahavirtojen kulkuun ja yrityksen käyttöpääoman määrään negatiivisesti. Avoimissa projekteissa ei saisi olla kovin paljon pääomaa kiinni. Tässä on kysymys ajankäytön hallinnasta, jota tukemaan tarvitaan toiminnanohjausjärjestelmä. Järjestelmän avulla pyritään minimoimaan yhdellä hetkellä käsittelyn tyhjäkäyntiä ja toisella hetkellä käsittelyn pullonkauloja.

Järjestelmän kautta vastaanotetaan sähköinen tunti-ilmoitus heti asiakkaan kuittauksen jälkeen ja voidaan viipymättä ryhtyä toimenpiteisiin. Jos vielä lasku saadaan suoraan ulos järjestelmästä oikeassa muodossa ja lähetettyä sähköpostilla, voidaan päästä hyvinkin lyhyisiin käsittelyaikoihin (tunti-ilmoitukseen laitetaan asiakkaan sähköpostiosoite/postiosoite laskutusta varten).

Tehtävän aiheuttamat varastovähennykset

Varastovähennykseen täytyy saada tieto siitä, mistä varastosta materiaali on peräisin (päävarasto, auto1,..). Jos varastoa ei ole ilmoitettu, vähennystä ei tehdä ko. materiaalin osalta. Tämä tarkoittaa sitä, että materiaali on haettu tukusta nimenomaista asiakastyötä varten, eikä sitä ole välissä lisätty pää- tai autovarastoon (ei tarvetta).

Varastotieto täytetään materiaalikohtaisesti tunti-ilmoitukseen. Tarvittaessa käytetään oletuksena arvoa tehtävän oletusvarasto-kentästä, joka on työn suorittajan oletuskäyttövarasto.

Tunti-ilmoituksen allekirjoitus

Tunti-ilmoituksen elektroniseen allekirjoitukseen tarvitaan jokin menetelmä, jos se halutaan saada toimimaan esimerkiksi Pad-pohjaisesti. Padille hankitaan sovellus, jolla voi signeerata vaikka PDF-tiedoston. Järjestelmästä voi sitten tulostaa tunti-ilmoituksen PDF-tiedostona, joka signeerataan sovelluksella (jokin PDF-lisäosa). Tämän jälkeen signeerattu PDF-tiedosto ladataan tehtävän liitteeksi.

Yllä kuvaillun toiminnallisuuden suora integroiminen osaksi järjestelmää ei ole kovinkaan helposti mahdollista, ellei siihen tarkoitukseen ole valmista moduulia olemassa.

Tunti-ilmoituksen täyttäminen

Jos halutaan minimoida asentajien järjestelmän käyttöön kuluva aika, voidaan myös täyttää tunti-ilmoitukset suoraan vaikka Word-pohjalle ja kuitata ne, jolloin tehtävään ladattaisiin tunti-ilmoitus ainoastaan liitteenä. Tämä aiheuttaa hieman lisätyötä toimistolla, kun lasku pitää muodostaa käsin liitetiedoston pohjalta. Yhtälailla liitetiedosto voi olla skannattu paperinen tunti-ilmoitus, mutta siinä tapauksessa tarvittaisiin asentajien mukaan kannettavat skannerit. Tai jokin muu väline, jolla A4-kokoisen sivun saa selkeästi näkyviin.

Optimitilanne olisi kuitenkin, että tieto siirtyisi järjestelmään mahdollisimman nopeasti ja helpoimmin käsiteltävässä muodossa, jolloin yksi toimihenkilö voi nopeasti hoitaa usean tehtävän laskutuksen ja muun jatkokäsittelyn.

3.4 Varastonhallinta

OpenERP:n varastonhallinta perustuu nykyään yleisessä käytössä olevaan ns. kaksinkertaiseen varastokirjanpitoon. Se tarkoittaa käytännössä sitä, että ei varsinaisesti käytetä käsitteitä katoaminen, kulutus tai hävikki, vaan aina on kyse varastosiirrosta paikasta toiseen. [1.]

3.4.1 Varastotyypit

Järjestelmä käyttää varastoista kolmea päätyyppiä: fyysiset varastot, asiakasvarastot ja virtuaaliset varastot.

Fyysiset varastot ovat nimensä mukaisesti yrityksen todellisia varastoja ja sisältävät todellista materiaalia.

Asiakasvarasto sisältää tuotteita, joita on myyty asiakkaille tai, jos kyseessä on tavarantoimittaja, se sisältää tuotteita, joita on ostettu. Tavarantoimittajien varastomerkinnot ovat tavallisesti negatiivisia, koska varastosiirtojen summa täytyy olla aina 0.

Virtuaalisia varastoja voidaan käyttää kuvaamaan esimerkiksi hävikkiä [1].

Asiakasvarastoja voisi halutessaan avata esimerkiksi jokaiselle suurelle asiakkaalle oman, mikäli on tarvetta seurata tarkasti myytyjä tuotteita asiakaskohtaisesti. Yleisemmin kuitenkin käytetään varastoja asiakaskategorian mukaan.

Premium Sähkö Oy:llä on tällä hetkellä kahdentyyppisiä fyysisiä varastoja: päävarasto ja ajoneuvovarastot. Järjestelmän toiminnallisuuden kannalta näillä varastoilla ei ole eroja. Varastoitavalle materiaalille on tärkeää määrittää varastopaikka oikein, jotta tiedetään jokaisen varaston kapasiteetti ennen töiden aloitusta. Varsinkin ajoneuvovarastot pitäisi pystyä pitämään mahdollisimman tarkasti ajan tasalla.

3.4.2 Varastonhallinnan työkalut

Varaston hallitsemiseen käytetään varastonhallinnan välilehdeltä löytyviä työkaluja. Tärkeimmät ovat saapuvan ja lähtevän materiaalin hallinta.

Saapuva materiaali

Saapuvan materiaalin hallintaan käytetään lomaketta. Tarkistetaan, että toimitetut tuotteet ovat kunnossa ja niitä on olemassa tilattu määrä. Tämän jälkeen kuitataan materiaali saapuneeksi varastoon. Mikäli materiaali hajautetaan usealle eri varastolle, voidaan kuitata osamääriä erilliskuitauksina.

Hankinnan lähtövarastoksi laitetaan tavarantoimittajien oletusvarasto ja kohdevarastoksi oma fyysinen varasto.

Lähtevä materiaali

Lähtevän materiaalin lähtövarastoksi laitetaan oma fyysinen varasto ja kohdevarastoksi asiakkaiden oletusvarasto. Tämä toiminto on tarkoitus toteuttaa automaattisesti tehtävän materiaalilistan pohjalta.

Määräykset ja keräily/varastointi

Määräys sisältää listan keräiltävistä/varastoitavista tuotteista. Määräyksen voi kuitata valmiiksi suoraan, jolloin listan tuotteiden määrät päivittyvät samanaikaisesti varaston tiedoissa. Vaihtoehtoisesti yksittäiset tuotteet voi kuitata ensin ulos tai sisään ja vasta sen jälkeen kuitata määräys valmiiksi. Jos yksi työntekijä hoitaa varastointia, riittää, että hän kuittaa määräyksen valmiiksi, kun tuotteet ovat oikeilla paikoillaan.

Varastosiirtolista

Siirtolista sisältää kaikkien tuotteiden liikkeet varastoihin tai sieltä pois. Listasta selviää lähtö- ja kohdevarastot, siirtopäivä, referenssi, yms.

Hävikki ja romutus

Materiaalin hävittäminen toteutetaan siirtämällä halutut tuotteet hävikki- tai romutettavarastoihin (virtuaalisia varastoja).

3.5 Sähkötuotteet

Sähkötuoteluettelo on tarkoitus tuoda järjestelmään tukkuliikkeen Excel-tiedostosta. Tässä luettelossa tuotekohtainen tunnus on suoraan tuotteen sähkönnumero, tämä on se tunnus, johon viitataan tuotetta valittaessa. Excel-listan tuonti aiheuttaa hieman työtä, koska lista on ensin saatava sellaiseen muotoon, jota OpenERP osaa tulkita. OpenERP osaa lukea CSV-taulukkotiedostoja, joita voi luoda Excelistä. Tuotelistan tiedot on kuitenkin ensin järjestettävä oikealla tavalla järjestelmän kannalta. Testatessa

tuotelistan tuontia järjestelmään huomattiin muutamia muitakin ongelmakohtia, mutta lopulta koetuonti saatiin onnistumaan virheettömästi.

Käytettäväksi kaavaillussa täysikokoisessa sähkötuotelistassa on n. 17000 riviä, eli sen täysimääräinen lataaminen järjestelmään on erittäin hidasta. Tästä syystä tulisi miettiä, tarvitaanko tuotelista järjestelmään kokonaisuudessaan. Vaihtoehtoisesti voitaisiin jättää listalta pois useita toistensa kanssa samankaltaisia tuotteita, joita väistämättä on listalla useita.

Tuotelista palvelee ensisijaisesti tarjous- ja laskutusvaiheen laskentaa. Tuotteista vain murto-osaa tullaan missään vaiheessa ostamaan varastoon. On tietenkin hyvä jos voidaan suoraan nähdä harvinaisemman tuotteen hinta listalta ja lähettää tilaus toimittajalle järjestelmän kautta.

Tuotteiden hankintahinnat

Tuotteiden hinnat muuttuvat tietenkin hieman vuosittain. Tästä syystä tuotelista tulisi päivittää erityisesti hintojen osalta tarpeen mukaan. Kun tiedetään tuotteen sähkönumero, joka on tuotelistan avaintieto, voidaan ajaa tuotetietojen tuonti ainoastaan hintojen osalta. Näin saadaan hinnat päivitettyä ajan tasalle, jättäen muut tuotteiden tiedot ennalleen. Tämä seikka on tärkeä, koska tuotteisiin on voitu tehdä käsin muutoksia ensimmäisen tuonnin jälkeen, esimerkiksi tarkennuksia selitekenttään. Mikäli tuotetiedot ajettaisiin joka kerta uudelleen täydellisenä tietokantaan, katoaisivat kaikki jälkeensä tuotteiden tietoihin tehdyt muutokset sen seurauksena.

Käyttöyksiköt

Sähkötuotteille tarvitaan peruskäyttöyksikköjä periaatteessa ainoastaan kaksi: kappale KPL ja metri M. Joissain tapauksissa voisi olla tarvetta myös kilogrammalle KG. Tukku- liikkeet käyttävät myyntiyksikköinä esim. pussi PSS tai paketti PKT sekä muita harvemmin käytettyjä yksiköitä. Yhteistä näille on se, että niiden peruskäyttöyksikkö on kuitenkin aina KPL. Järjestelmään tarvitaan ainoastaan peruskäyttöyksikkö sekä hankintakerroin, joka kuvaa kuinka monta kappaletta/metriä kyseistä tuotetta sisältyy yhteen myytävään perusyksikköön. Metritavaran osalta kerroin kuvaa siis vähimmäismetrimäärää. Sähköjohdot ostetaan tietenkin aina kelalla, joita voi myös kuvata kappaleit-

tain, tärkeintä on kuitenkin, että hankintahinta muodostuu oikein. Hankintayksikkö voidaan tarvittaessa määritellä tapauskohtaisesti.

Tukkuliikkeen tuotelistasta ei siis tarvitse tuoda kaikkea tuotekohtaista tietoa järjestelmään, ainoastaan keskeisimmät tiedot. Usein käytettyjen tuotteiden osalta tarvittavat lisätiedot voidaan lisätä myöhemmin käsin.

Tuotteiden myyntihinnan määrittely

Tuoterakenteessa on määritelty erilliset kentät myyntihinnalle ja hankintahinnalle. Hankintahinta tuodaan tukkulistasta, mutta myös myyntihinta tulisi määritellä automaattisesti järjestelmää varten. Yksi ratkaisu on määritellä tuotekohtainen katekerroin, jonka perusteella järjestelmä laskee myyntihinnan hankintahinnan perusteella. Tuotteille annetaan standardikerroin, esim. 1,2. Tuotaessa tuotteet CSV-taulukosta, voidaan kaikille määrätä sama kerroin. Sitä voi myöhemmin tarvittaessa muuttaa tuotekohtaisesti. Tuotteiden myyntihinnat voidaan myös määritellä suoraan tuonnin yhteydessä. Molemmat hinnat on hyvä lisätä suoraan järjestelmään, jotta voidaan myöhemmin toteuttaa kattavaa myynnin analyysiä.

3.6 Palvelimen perustaminen

Premium Sähkö Oy:llä on käytössä Linux-palvelin, joka sijaitsee fyysisesti Saksassa. Siihen päästään käsiksi verkon kautta käyttämällä tarkoitukseen sopivia apuohjelmia, joilla voi hallinnoida etäkäytöllä palvelimella olevaa dataa. Linux-palvelimelle voidaan tarvittaessa perustaa useita OpenERP-palvelimia. Tarkoitus on perustaa varsinaisen tuotantopalvelimen lisäksi testiympäristö, jossa voi kehittää ja testata uusia toiminnallisuuksia ennen niiden käyttöönottoa. Tarvittaessa olisi myös mahdollista tarjota jollekin toiselle yritykselle oma itsenäinen OpenERP-palvelin.

PostgreSQL

OpenERP-palvelimen taustalla pyörii aina PostgreSQL-palvelin. PostgreSQL on avoimen lähdekoodin tietokantojen hallintajärjestelmä. Kaikki OpenERP:ssä näkyvät tietokannat on todellisuudessa luotu PostgreSQL-järjestelmään, joka pitää sisällään kaiken

raakadatan. OpenERP puolestaan sisältää ERP-toimintalogiikan, ja sen kautta lähetetään tietokantoja hallinnoivia komentoja PostgreSQL-järjestelmään. [1.]

Muokkauksia suoraan PostgreSQL-tietokantoihin pääsee tekemään käyttämällä apuohjelmia Psql tai PgAdmin. Psql on komentorivillä käytettävä ohjelma, jonka kautta voi käyttää SQL-komentoja omiin tietokantoihin. PgAdmin on vastaava graafisella käyttöliittymällä varustettu ohjelma. [1.]

Tietokantojen muokkaaminen PostgreSQL:n kautta ei ole jokapäiväisessä käytössä tarpeellista, mutta tietyissä tilanteissa siitä on hyötyä. Tietokantojen varmuuskopiointi kiinteille onnistuu ainoastaan PostgreSQL:n kautta. OpenERP:n puolella on olemassa toiminto varmuuskopiointille, mutta se ei ainakaan tämän työn kirjoitushetkellä toimi. Myös tietokantojen poistaminen ja uusien luominen on nopeampaa PostgreSQL:n kautta. Joissain tilanteissa yksittäinen korruptoitunut tietokanta voi haitata OpenERP-palvelimen toimintaa tai estää sen jopa kokonaan. Ongelmallisen tietokannan saa kuitenkin aina poistettua PostgreSQL:n kautta.

Testipalvelimen perustaminen

Pelkkää moduulikehitystä varten kannattaa asentaa yhdelle tietokoneelle OpenERP:n Windows-asennuspaketti. Windows-asennus on yksinkertainen ja nopein suorittaa ja se palvelee pelkkää testaamistarkoitusta hyvin. Testikoneelle avataan paikallinen OpenERP-palvelin, jonka toiminta on päältä katsoen sama kuin millä tahansa muullakin OpenERP-palvelimella.

Uuden moduulin voi lisätä kiinteiden moduulien joukkoon OpenERP-asennuksen moduulikansioon, tai määrittää erillisen polun omille moduuleille. Testatessa omaa moduulia, siihen tehdyt muutokset täytyy päivittää järjestelmään erillisellä päivityspainikkeella. Ennen päivitystä OpenERP-palvelin tulee käynnistää uudelleen.

3.7 Järjestelmän laajennukset ja toimintaympäristöt

Uuden laajennusmoduulin asentaminen on tavallisesti suoraviivainen operaatio. Se tapahtuu parilla napin painalluksella. Kuitenkin, kun on kyse jo asennetun moduulin muokatusta versiosta, suoran päivittämisen lisäksi OpenERP-palvelin täytyy usein

alustaa, jotta vältetään mahdollisilta virheilmoituksilta. Myös tietokannassa jo valmiina oleva data saattaa aiheuttaa arvaamattomia tilanteita moduulien päivitysten yhteydessä. Näistä syistä on erittäin tärkeää hallinnoida useita tietokantoja samanaikaisesti. Tuotanto ja testiympäristö täytyy pitää erillisinä. Parhaassa tapauksessa tuotanto- ja testiympäristö pidettäisiin toisistaan erillisillä OpenERP-palvelimilla.

Uusien toiminnallisuuden käyttöönotto tuotantoympäristöön voidaan toteuttaa vähäisen käytön aikana iltaisin. Aina ennen uutta käyttöönottoa otetaan varmuuskopio tuotannon tietokannasta, jotta ongelmatilanteissa voidaan palauttaa alkuperäinen tietokanta kokonaisuudessaan.

3.8 Järjestelmän räätälöinnin perusteet

3.8.1 Kehittäjätila

OpenERP tarjoaa käyttäjälle räätälöintimahdollisuuksia usealla tavalla. Järjestelmän rakenteen ja toiminnallisuuden muokkaaminen on tehty osittain mahdolliseksi suoraan käytön aikana, käyttämällä kehittäjätilaa (*developer mode*). Kehittäjätilassa voi tehdä muokkauksia näkymiin (lomake, lista, kanban), mutta ei voi luoda uusia luokkia. Kuitenkin tällaisessa suorassa muokkaustavassa on huomattavia ongelmia, jotka tekevät sen epäkäytännölliseksi.

Kehittäjätilassa muokkaamisen ongelmat:

- Jos moduulin näkymää on muutettu kehittäjätilassa ja kyseinen moduuli joudutaan asentamaan uudelleen, moduulin muutokset menetetään.
- Jos koko tietokanta kaatuu, kaikki muutokset menetetään.
- Jos halutaan luoda uusi samanlainen tyhjä tietokanta, joudutaan muutokset lisäämään uudelleen käsin.
- Kehittäjätila on hidas ja tahmea.

Johtuen yllä esitetystä syistä, kehittäjätilan käyttö ei ole järkevää edes pienten muokkausten tekemiseen. Kehittäjätilasta on kuitenkin selkeää hyötyä kehittäjille siinä, että sen avulla pystyy toteamaan helposti esim. jokaisen kentän riippuvuudet ja luokan, johon se kuuluu.

3.8.2 Räätelöinti moduulipohjaisesti

Yleinen käytäntö OpenERP:n kehittämisessä on luoda yksi tai useampia räätälöintimoduuleita, jotka sisältävät kaikki halutut järjestelmän muokkaukset. Esimerkiksi niinkin yksinkertainen muokkaus kuin jonkin listan järjestäminen oletuksena eri tavalla kuin normaalissa näkymässä, voidaan toteuttaa oman lisämoduulin kautta.

Idea omassa räätälöintimoduulissa on se, että se voidaan helposti asentaa uuteen tietokantaan. Koska yksi moduuli voi sisältää kaikki muutokset, niitä on helppo jälkikäteen korjata, kun kaikki löytyvät samasta paikasta.

3.8.3 Python ja XML

Moduulin runko rakennetaan käyttäen Python- ja XML-kieliä. Python on varsinainen ohjelmointikieli, jolla toteutetaan toimintojen ohjelmointi, esim. toimintopainikkeet tai funktionaaliset kentät. XML on rakennetta ja merkintää kuvaava kieli, jonka avulla luodaan näkymät, joissa toiminnot sijaitsevat määrätyillä paikoillaan. XML-tiedostoja muokkaamalla tehdään siis varsinaiset muutokset järjestelmän käyttöliittymään ja ulkoasuun.

3.8.4 Moduulin perusrakenne

Moduulit ovat järjestelmän osia, jotka lisäävät toiminnallisuutta perusjärjestelmän päälle. Suurin osa moduuleista on riippuvaisia jostakin toisesta moduulista, jolloin kyseinen moduuli pitää olla asennettuna ennen siitä riippuvaisen moduulin asennusta.

Moduuli on omassa kansiossaan, joka sisältää ainakin moduulin rakennetta ja kuvausta varten Python- ja XML-tiedostoja. Yhdessä moduulissa voi olla kymmeniä tiedostoja, jotka kaikki osaltaan lisäävät moduulin toiminnallisuutta.

Moduuleja säilytetään OpenERP-palvelimen addons-kansiossa, mutta omille moduuleille voi myös selkeyden vuoksi määritellä erillisen kansion ja hakupolun.

Yksinkertaisimmillaan moduuli sisältää kolme Python-tiedostoa:

__init__.py

Alustustiedosto, joka tulee yleisesti sisällyttää jokaiseen erilliseen kansioon. Jos se jätetään pois, OpenERP ei osaa lukea ko. kansion sisältöä. Tähän tiedostoon listataan moduulin käyttämät PY-tiedostot ja mahdolliset alakansiot (sisältävät oman alustustiedoston).

__openerp__.py

Tiedosto sisältää moduulin kuvaukset ja muut tekniset yksityiskohdat. Tähän tiedostoon mm. listataan moduulin riippuvuudet ja kaikki sen käyttämät XML-tiedostot.

oma-tiedosto.py

Tämä tiedosto on vapaasti määriteltävissä. Se sisältää varsinaisen järjestelmää muokauttavan moduulikoodin.

Alla on esitetty muita moduulien kehittämiseksi keskeisiä käsitteitä.

XML-tiedosto

Aputiedosto, joka sisältää näkymien kuvauksia. Käytännössä kaikissa moduuleissa on aina vähintään yksi xml-tiedosto. XML-tiedostolla voidaan myös ladata demo- tai oletusdataa tietokantaan moduulin asennusvaiheessa.

Luokka (class)

Luokka on perusrakenne Python-tiedoston sisällä, joka sisältää erilaisia kenttiä, ts. muistipaikkoja, sekä funktioita. Kaikki tietokannan rakenteet perustuvat luokkiin. Luokkia kutsutaan yleisesti myös objekteiksi tai olioiksi.

Uuden luokan voi periä vanhasta, jolloin uuden luokan määrittelyt ovat myös perityn luokan käytettävissä. Luokkien perimistä on käsitelty tarkemmin osiossa 3.8.5: Moduulien periytyminen.

Kenttä (field)

Kentät säilyttävät dataa. On olemassa useita erilaisia kenttätyppejä. Esimerkiksi kokonaisluku, teksti, valinta, jne.

Funktiot (def)

Tietokannan aputoiminnot toteutetaan funktioilla. Esimerkiksi lomakkeilla oleviin toimintopainikkeisiin on aina kiinnitetty jokin funktio.

Näkymät (XML)

Näkymät määrittelevät, miltä tietokannan rakenteet näyttävät käyttäjälle. Näkymiä muuttamalla voidaan määrittää uudelleen esim. yksittäisten kenttien sijainteja. Näkymät ovat keskessä asemassa järjestelmän käytettävyyden kannalta.

Näkymät ovat omassa XML-tiedostossa, johon voi määritellä useita tietueita (*record*). Myös tietueet ovat perittävässä. [12.]

3.8.5 Moduulien periytyminen

Periytyminen on keskeinen asia toimittaessa OpenERP:n moduulipohjaisessa kehitysympäristössä. Periytymisen oikeaoppinen hallinta on tärkeä osa räätälöintimoduulien toimivuutta.

OpenERP on avoin järjestelmä, jonka kaikki moduulit, myös sertifioidut, ovat periaatteessa käyttäjän suoraan muokattavissa. Kuitenkaan suoria muokkauksia kiinteiden moduulien rakenteisiin ei pitäisi tehdä. Syitä tähän on useita. Ensinnäkin moduulien versionhallinnasta tulisi käytännössä mahdotonta. Mikäli taas tulisi tarvetta avata uusia palvelimia, pitäisi kaikki muokatut moduulit aina kopioida uuden palvelimen addons-kansioon. Tämän lisäksi muokkaukset olisivat täysin hajallaan eri moduuleissa, mikä olisi erittäin epäkäytännöllistä ja vaikeuttaisi virheiden etsintää.

Tehokas periytyminen tarjoaa ratkaisun ongelmaan. Luodaan uusi moduuli, jonka luokat perivät valmiita luokkia kiinteistä moduuleista. Tällä tavoin voidaan lisätä erilaisia kenttiä tietueisiin tai korvata kenttä tai toiminto toisella. Näkymiä voidaan periä samalla

tavalla, jolloin päästään vaikuttamaan suoraan siihen, miten käyttäjä näkee ja kokee käyttöliittymän. Voidaan esim. poistaa näkyvistä turhia kenttiä tai lisätä uusia.

Periminen mahdollistaa täsmälleen samat muutokset kuin suoraan muuttamalla saataisiin aikaan, mutta käytettävyyden ja selkeyden suhteen muokkaustapana yksi moduuli, joka sisältää perittyjä luokkia, on ylivertainen.

Luokat voivat periä toisiaan kahdella tavalla. Lisämoduuleissa yleisesti käytetty tapa on periä kiinteä luokka ja antaa uudelle luokalle sama nimi kuin perityllä luokalla (sama toiminta jos nimen jättää pois). Tällainen uusi luokka ainoastaan laajentaa vanhaa luokkaa, eli ei oikeastaan ole uusi erillinen luokka ollenkaan. Uusi luokka on ainoastaan laajennusosa jo olemassa olevaan luokkaan.

Toinen perimistapa on luoda uusi luokka, jolla on eri nimi kuin luokalla, jonka se perii. Tällöin uusi luokka saa käyttöönsä kaiken sisällön peritystä luokasta sen omien määrittysten lisäksi. Peritty luokka ei näe uusia lisäyksiä, vaan luokat ovat täysin erillisiä. [12.]

3.8.6 Esimerkkejä

Tässä osiossa esitetään konkreettisia esimerkkejä pienistä muokkauksista, jotka ovat kuitenkin aina tärkeässä osassa järjestelmän joustavuuden ja helppokäyttöisyyden kannalta. Muokkaukset on tehty omaan räätälöintimoduuliin. Esimerkkien ei ole tarkoitus olla kattavia ohjeita, vaan ainoastaan hieman kuvata käyttöönottoprojektin vaatimaa jatkuvaa kehitystä ja testaamista.

Alla oleva Python-koodi järjestää varastosiirtolistan siirtojen kirjaamispäivän mukaan. Oletuksena lista järjestyi siirtojen määröpäivän mukaan.

```
***pm_muokkaukset.py
class pm_stock_move_fix(osv.osv):
    _inherit = 'stock.move'
    _order = 'date desc'
```

Täytyy siis luoda uusi luokka, joka perii (`_inherit`) objektin "stock.move". Uuden luokan objektinimeä (`_name`) ei tässä määritellä erikseen, vaan nimi on sama kuin kantaobjektilla. Varsinainen haluttu toiminto toteutetaan määrittelemällä "`_order`"-attribuutille uusi

arvo. Valittu arvo on siirron kirjaamispäivämäärä "date". "Desc"-lisäys tarkoittaa laskevaa järjestystä (*descending*).

Seuraava esimerkki on ote XML-koodista, jolla muutetaan kaksi harvemmin Suomessa tarvittavaa asiakaslomakkeen kenttää näkymättömäksi.

```
***pm_view_fix.xml
<!--korjaukset asiakas-formiin-->
<record model="ir.ui.view" id="pm_view_fixes">
  <field name="name">res.partner.form</field>
  <field name="model">res.partner</field>
  <field name="inherit_id" ref="base.view_partner_form"/>
  <field name="arch" type="xml">

  <!--osavaltio-kentta piiloon-->
    <field name="state_id" position="attributes">
      <attribute name="invisible">True</attribute>
    </field>
  <!--titteli-kentta piiloon-->
    <field name="title" position="attributes">
      <attribute name="invisible">True</attribute>
    </field>

  </field>
</record>
```

Kaikki näkyviin toteutettavat muutokset täytyy aina sisältyä XML-tietueeseen (*record*). Luodaan uusi tietue "pm_view_fixes", joka perii kantatietueen "base.view_partner_form".

Toinen XML-esimerkki, josta selviää, miten voidaan muuttaa kenttien paikkoja lomakkeilla. Muutos on tehty peritylle tehtävä-lomakkeelle. Asiakas-kenttä haluttiin saada lomakkeen etusivulle.

```
***pm_muokkaukset.xml
<field name="partner_id" position="replace"/>
```

```
<field name="date_deadline" position="before">
  <field name="partner_id" />
</field>
```

Asiakas-kentän uudeksi paikaksi määritetään sijainti ennen etusivun deadline-kenttää (kuva 6).

Kuva 6. Muokattu tehtävä-lomake

Esimerkki tuotelistan lisäämisestä tehtävärakenteeseen:

```
***pm_muokkaukset.py
class pm_tyotuotelista(osv.osv):
    _name = "project.task.products"
    _description = "Project Task products"

    _columns = {
        'task_id': fields.many2one('project.task', 'Task'),
        'product_id': fields.many2one(
            'product.product', 'Product'),
        'quantity': fields.float('Quantity'),
        'prod_uom': fields.many2one('product.uom', 'UoM'),
        'prod_name': fields.char('Name'),
```

}

Yllä olevasta esimerkistä on yksinkertaistuksen vuoksi jätetty pois osia toiminnallisuudesta, mutta siitä voi havaita, että tuotelistaa varten on luotu uusi luokka, joka on tarkoitettu lisätä kiinteän tehtävä-luokan sisälle erillisellä perimiskoodilla.

3.9 Raportointi ja raporttien muokkaaminen

Raportointi on keskeinen osa ERP:n toiminnallisuutta. Selkeät raportit ovat avainasemassa yrityksen ulkoisissa toiminnoissa, mutta ne oikein käytettynä myös tehostavat yrityksen sisäistä viestintää.

Automaattisesti järjestelmästä generoitavat raportit koostuvat RML-tiedostoista, jotka sisältävät raportin ulkoasuun ja sisältöön vaikuttavan informaation. RML-tiedosto on kirjoitettu erityisellä RML-kielellä, joka on automaattista raportointia varten kehitetty muunnelmä XML-kielestä. Raportin generointia varten järjestelmä hakee tietokannasta RML-tiedostossa määritetyt rakenteet, eli objektit, jotka sisältävät esitettävän datan, ja käyttää määrättyjä muotoiluja. Lopputuloksena on raportti, yleisimmin PDF-muodossa.

RML-pohjan muokkaaminen

RML-tiedosto sisältää muotoilumääreitä ja Python-viittauksia raportissa esitettävään dataan. Näitä voi halutessaan muokata suoraan RML-kielellä.

RML-pohjien muokkaamiseen on myös olemassa työtä helpottavia työkaluja. RML-tiedoston voi generoida OpenOfficen SXW-tiedoston pohjalta. SXW-tiedostoa voi muokata OpenOfficella kuten mitä tahansa tekstidokumenttia. Raportin luomiseen käytetyn SXW-tiedoston voi tavallisesti löytää moduulien reports-kansiosta, jossa se on saman nimisen RML-tiedoston ohella. OpenERP ei käytä SXW-tiedostoja raporttien generointiin, vaan ne ovat mukana ainoastaan kehittäjien apuna.

Raportin ylä- ja alatunniste

Useimmat oletusraportit käyttävät yhteisiä ylä- ja alatunnisteita, jotta eri raporttien ulkoasu olisi mahdollisimman yhtenevä. Yleinen ylä- ja alatunniste sisältää yrityksen nimen

ja logon sekä yhteystietoja. Näiden tietojen muokkaaminen on tehty helpommaksi suoraan järjestelmästä käsin. Kehittäjätilassa voi muokata tunnisteiden RML-koodia. Olisi kuitenkin suositeltavaa tehdä myös tunnisteiden muokkaukset lisämoduulin avulla.

Raporttipohjan korvaaminen uudella

Raportteihin tehtävät muutokset menevät samalla kaavalla kuin muutkin muokkaukset, eli omaan lisämoduuliin lisätään uusi luokka, joka perii olemassa olevan luokan. Esimerkiksi halutaan korvata järjestelmän käyttämä oletuslaskupohja uudella. Oletuslaskupohja ei usein ole sellaisenaan riittävä yritysten tarpeisiin, varsinkaan kun kyse on suorasta asiakaslaskutuksesta.

Lähes kaikki kiinteät moduulit sisältävät raporttipohjia. Laskutusraportit sijaitsevat kirjanpito-moduulissa (*accounting*). Raporttipohjat, eli RML-tiedostot ja niihin vaikuttava koodi laitetaan tavallisesti moduulin reports-kansioon.

Seuraavaksi etsitään Python-luokka, joka sisältää toiminnon, joka halutaan korvata (ts. ylikirjoittaa). Tässä tapauksessa "account.invoice"-luokka sisältää funktion "print_invoice". Funktio "print_invoice" on toiminto, joka kutsutaan OpenERP:stä aina kun halutaan generoida lasku. Mikäli lisämoduulin perityn "account_invoice"-luokan funktiolle annetaan sama nimi, se korvaa vanhan funktion. Muita muutoksia ei tarvita, tämän jälkeen järjestelmä generoi aina muokatun laskupohjan. Uuteen "print_invoice"-funktioon pitää ainoastaan vaihtaa sen käyttämä RML-tiedosto omaan muokattuun versioon.

Lopputuloksena lisämoduuli sisältää reports-kansion, johon laitetaan uusi RML-tiedosto sekä PY-tiedoston, joka sisältää yllä selostetun perijäluokan.

Kun lisämoduuli asennetaan tai päivitetään tietokantaan, uudet muutokset laskuraportin osalta tulevat voimaan. Uusi lisämoduuli on perimisestä johtuen riippuvainen kirjanpito-moduulista, joka tulee siten olla asennettuna ennen lisämoduulin asennusta.

3.10 Kielen käännökset

OpenERP:n asetuksista voi ladata suomenkielisen käännösversion, mutta se on vielä tässä vaiheessa huomattavan puutteellinen. Puuttuvat ja tarvittavat käännökset joudutaan tekemään itse. Käännöksiä voidaan tehdä esimerkiksi erilliseen CSV-tiedostoon tai suoraan tietokantaan. Suoraan tietokantaan tehdyt muokkaukset eivät ole kannatettavia samasta syystä kuin muussakin muokkaamisessa: mikäli tietokanta menetetään, menetetään myös siihen tehdyt muutokset.

4 Käyttöönoton suunnittelu

Keskeisimmät toiminnallisuudet ovat töiden aikataulutus kalenteripohjaisesti, työkeikan tietojen tallennus (tehtävärakenne), materiaalin ja työtuntien ilmoittaminen ja työn laskutus.

Näitä toimintoja tukevat laaja tuotetietokanta ja varastonhallinta, kuten myös asiakastietokanta sekä myynnin työkalut.

Kun saadaan yhteydenotto potentiaaliselta asiakkaalta, pitäisi nopeasti pystyä katsomaan kalenterista työlle sopiva ajankohta. Pienistä työkeikoista pitäisi pystyä sopimaan nopeasti jopa yhden puhelun perusteella.

Kun työstä saadaan sovittua, kalenteriin avataan uusi työtehtävä, jolle määrätään vastuhenkilö. Työtehtävästä selviää tiedot kohteesta ja yksityiskohdista, joiden perusteella työ pitäisi pystyä suorittamaan. Työtunnit ja käytetty materiaali tarkentuu työn edetessä. Kun työ on valmis, asentaja täyttää tehtävän tunti/materiaalilistaan käyttämänsä tuotteet (materiaali, työtunnit, km-korvaukset, yms.). Asiakas kuittaa listan järjestelmän kautta, jonka jälkeen voidaan lähettää lasku asiakkaalle.

4.1 Avainkohdat

Suunniteltaessa uuden järjestelmän käyttöönottoa, asioille tulee määritellä tärkeysjärjestys, jotta saadaan osia järjestelmän potentiaalista käyttöön jo aikaisessa vaiheessa. Projektin alussa laadittiin projektisuunnitelma (Liite1), jossa esitettiin vaiheittainen käyttöönottojärjestys. Vaatimukset ja tilanne ovat sittemmin tarkentuneet vastaamaan tarkemmin yrityksen todellista tarvetta.

Tämän projektin osalta tarkentunut käyttöönottojärjestys on seuraava:

1. Kattavan tuotetietokannan luominen
2. Järjestelmään täytettävän tunti-ilmoituksen lisääminen (tehtävienhallinta)
3. Töiden aikataulutuksen lisääminen

4. Asiakashallinnan työkalujen käyttöönotto
5. Varastonhallinnan työkalujen käyttöönotto
6. Muut asiat.

4.2 Kustannukset

Vaikka OpenERP on jaossa ilmaisella lisenssillä ja sen voi ladata ja asentaa omalle palvelinkoneelle maksutta, ei kannata silti puhua yritykselle täysin maksuttomasta järjestelmästä. Lisenssimaksut ovat aina vain yksi osa tietojärjestelmän käyttökustannuksista. Todelliset käytönaikaiset kustannukset määräytyvät järjestelmän tarvitsemista räätälöinneistä ja tukipalveluista sekä henkilöstön koulutustarpeesta. Nämä palvelut voidaan tilata erikseen ulkopuoliselta toimittajalta tai tehdä itse, jos osaamista löytyy riittävästi yrityksen sisältä. Kustannukset voidaan laskea myös sisäisen käyttöönoton tapauksessa, koska kyse on tällöin perehtymiseen ja koulutuksiin kuluneesta ajasta, joka on pois varsinaisesta tuottavasta työajasta.

Mikäli avoimen järjestelmän käyttöönotossa vaaditaan paljon integrointia ja räätälöintiä, kustannukset voivat lopulta muodostua hyvinkin samansuuruisiksi verrattaessa suljettuun, mutta paremmin yrityksen tarpeisiin soveltuvaan järjestelmään. Avoimen järjestelmän valinneella yrityksellä on kuitenkin suurempi vapaus valita, miten ja mistä lopulliset käyttökustannukset muodostuvat. [13.]

4.3 Projektin tilanne ja kohdatut ongelmat

OpenERP:n käyttöönotto on ollut haasteellisempaa ja hitaampaa kuin projektin alussa ehkä uskottiin. Tärkeimmät ongelmat koskevat yrityksen sisäistä kommunikointia ja johtamista järjestelmän kautta sekä asiakkaalta vaadittavan työtuntien kuittauksen saattamista järjestelmään. Lisäksi osa käytettävistä raporteista vaatii vielä hieman muokkaamista, jotta ne kelpaavat lähetettäväksi asiakkaille.

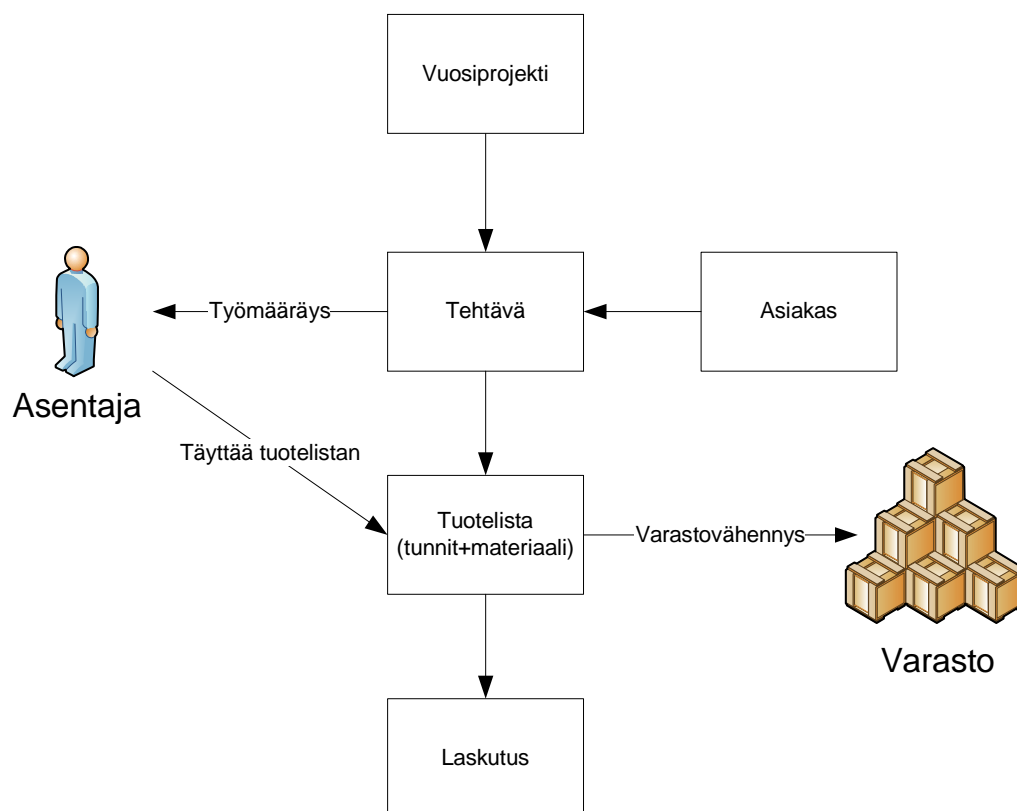
Järjestelmän yhdistäminen Google Business -palveluihin on rajoitettua, vaikka siihen tarkoitukseen on luotu useita lisäosia. Yrityksen kalenterin ja sähköpostin integroiminen on haastavaa ja aiheuttaa viivästystä järjestelmän täydelliseen käyttöönottoon.

4.4 Toteutettuja muokkauksia

Tähän mennessä muokkauksia on tehty yhden räätälöintimoduulin avulla, jotta muutosten asennus olisi mahdollisimman helppoa. Räätälöintimoduulia voidaan ensin testata testiympäristössä, kunnes se on valmis siirrettäväksi tuotantoympäristöön. Asennus onnistuu helposti kaikissa tapauksissa.

Projektinhallinta

Keskeisimmät muokkaukset on tehty projektinhallinta moduulin toimintoihin, jotta ne vastaisivat paremmin yrityksen tarpeita. Alla olevasta kaaviokuvasta 7 näkyy, miten projektinhallinta on tarkoitus järjestää, jotta saadaan maksimihyöty järjestelmästä. Jokainen tehtävä on aina sisällytetty vuosiprojektiin, mutta on myös tärkeää, että asiakasrakenteesta on suora yhteys vastaavan asiakkaan tilaamiin tehtäviin.



Kuva 7. Projektinhallinta

Projektinhallinnan työkaluista muutokset tulevat tehtävälomakkeelle. Tehtävälomakkeelle on lisätty tuotelista, johon kirjataan kaikki tehtävässä käytetyt tuotteet. Tuote voi

olla ainakin työtunti, kulukorvaus tai materiaali. Tarkoitus on, että asentajilla on oikeus täyttää tuotelista, mutta ei oikeutta laskutus- tai varastotoimintoihin.

Tehtävä-lomakkeelle on lisätty painonapit, joista voi luoda laskun ja varastovähennyslomakkeen. Tiedot siirtyvät tuotelistasta näin kätevästi eteenpäin muihin osioihin.

Raportointi

Tavallista laskun raporttipohjaa on tarvetta muokata suomalaisiin toimintatapoihin sopivaksi. Tämä tehdään muokkaamalla vastaavaa RML-tiedostoa halutulla tavalla. Raporttipohjien muokkaaminen on esitetty tarkemmin kohdassa 3.9.

Sähkötuotteet

Suuria tuotemassoja ei ole vielä ajettu järjestelmään, koska se vie paljon aikaa. Varsinkin, jos yritetään viedä kaikki tuotteet yhdellä kertaa. Tarkoitus on varmistaa, että järjestelmä pystyy riittävällä nopeudella käsittelemään suurtakin tuotetietokantaa. Tarkemmin sähkötuotteiden tuontia järjestelmään on käsitelty kohdassa 3.5: Sähkötuotteet.

4.5 Järjestelmän testaaminen esimerkkityötilauksen avulla

Mahdollisimman useassa käyttöönoton vaiheessa tulee testata esimerkkityötilauksen avulla järjestelmän toimivuutta sekä senhetkistä soveltuvuutta yrityksen tarpeisiin. Esimerkkityötilauksena voidaan alkuvaiheessa käyttää esimerkiksi jo toteutunutta työtilausta. Perinteisiä työmenetelmiä voidaan tällöin suoraan verrata esitettyihin uusiin toimintatapoihin.

5 Yhteenveto

Insinööriyön tavoitteena oli suunnitella ja aloittaa OpenERP-toiminnanohjausjärjestelmän käyttöönotto sähköurakointiliike Premium Sähkö Oy:ssä. Vaikka OpenERP:n käyttöönottoprojekti on edennyt suunnittelusta ja teknisten vaatimusten määrittelystä lähelle toteutuksen aloittamista, on silti vielä useita pieniä teknisiä ongelmia ratkaistavana. Projektia on tarkoitus jatkaa eteenpäin, ja kerätä samalla käyttökokemuksia jokaiselta yrityksen käyttäjäportaalta.

Tässä työssä on pyritty alusta asti selvittämään, mitä konkreettisia toimia tarvitaan, jotta saataisiin OpenERP-järjestelmä toimimaan juuri halutulla tavalla yrityksen määrättyyn tarpeeseen. Tästä syystä on ollut välttämätöntä tutustua varsin syvällisesti järjestelmän erityiskäyttöön ja räätälöinnin perusteisiin. Työn aikana karttunutta tietotaitoa ja materiaalia voidaan nyt alkaa soveltaa käyttöönoton eteenpäin viemiseksi. Suurin osa projektin keskeisistä potentiaalisista ongelmista on kuitenkin jo saatu selvitettyä, ja työn aikana hankitulla riittävällä järjestelmän tuntemuksella ratkaisut uusiinkin haasteisiin varmasti löytyvät aikaisempaa helpommin ja nopeammin.

Avoimen järjestelmän käytön hyödyt ovat edelleen hahmottuneet suunnittelutyön edetessä. Keskeisin etu on yrityksen vapaus valita itse oma toimintatapansa sekä tarvitsemansa palvelut. Joustava avoin järjestelmä parhaassa tapauksessa muuntuu käyttäjänsä näköiseksi, sen sijaan, että käyttäjä joutuisi itse mukautumaan toimimaan järkevän järjestelmän ehdoilla. Jo varsin vaatimattomalla OpenERP:n kehitysympäristön tuntemuksella käyttäjä pystyy tekemään monenlaisia muutoksia käyttöliittymään. Tärkeätkin parannukset on mahdollista saada käyttöön nopeasti ja ennen kaikkea kerralla oikein. Kaupallisten suljettujen järjestelmien kehittäjäorganisaatiot harvoin pystyvät tukitoiminoissaan vastaavaan joustavuuteen, nopeuteen ja tarkkuuteen.

OpenERP-järjestelmän toiminnallisuuden testaaminen ja sen mahdollisuuksien selvittäminen on ollut tähän mennessä erittäin kiinnostava kokemus. Erityisen mielekästä on ollut pohtia ratkaisuja teknisiin ongelmiin ja havaita, että suurimpaan osaan niistä yleensä löytyy ratkaisu, mahdollisesti jokin olemassa oleva järjestelmän laajennusosa tai omakohtainen räätälöinti. Vertailupohjana OpenERP:lle minulla on hieman käyttökokemusta SAP ERP-järjestelmästä. Vaikka käyttöalueeni onkin ollut suppea, yleinen SAP-käyttöliittymä on kuitenkin tuttu.

Vaikka tämän insinöörityön aihealue ei ole varmasti monenkaan sähköinsinöörin omin-
ta aluetta, on työtä osaltani helpottanut kiinnostukseni tietojärjestelmiin ja ohjelmointiin.
Ennen työn aloittamista minulla oli hieman kokemusta mm. Python-ohjelmoinnista.

Lähteet

- 1 Pinckaers, F., Gardiner, G., Vossel, E. 2013. OpenERP Book. <<https://doc.openerp.com/book/>>.
- 2 Open Source ERP Buyer's Guide. 2012. Verkkodokumentti. Enterprise Apps Today. <<http://www.enterpriseappstoday.com/erp/open-source-erp-buyers-guide.html>>. Luettu 19.4.2014.
- 3 Ritvanen, V., Inkiläinen, A., Bell, A., Santala, J. 2011. Logistiikan ja toimitusketjun hallinnan perusteet.
- 4 Wailgum, Thomas. OpenERP Definition and Solutions. 2008. Verkkodokumentti. CIO.com. <www.cio.com/article/40323/ERP_Definition_and_Solutions>. Luettu 10.4.2014.
- 5 Nieminen, Joel. 2013. Toiminnanohjausjärjestelmän valinta avoimen lähdekoodin näkökulmasta. Diplomityö. Tampereen teknillinen yliopisto.
- 6 Top 5 ERP Software Systems. Verkkodokumentti. <<http://www.erpsoftware360.com/erp-software.htm>>. Luettu 18.5.2014.
- 7 Open-source ERP 'most active' SourceForge project. 2010. Verkkodokumentti. <http://www.computerworld.com/s/article/9178761/Open_source_ERP_most_active_SourceForge_project>. Luettu 4.11.2013.
- 8 Avoimen koodin OpenERP tulee Suomeen. 2013. Verkkodokumentti. Tietoviikko. <http://www.tietoviikko.fi/kaikki_uutiset/avoimen+koodin+openerp+tulee+suomeen/a875255>. Luettu 30.4.2014.
- 9 OpenERP Enterprise. Verkkosivu. <<https://www.openerp.com/pricing>>. Viitattu 16.4.2014.
- 10 Top Open Source ERP Software. 2013. Verkkodokumentti. <<http://www.axzm.com/top-open-source-erp-software>>. Luettu 4.11.2013.
- 11 Premium Sähkö Oy. Verkkosivu. <<http://www.premiumsahko.fi/>>. Viitattu 16.4.2014.
- 12 OpenERP Developer Book. Verkkodokumentti. <<https://doc.openerp.com/6.0/developer/>>. <<https://doc.openerp.com/v6.1/developer/>>.

- 13 Aho, Jukka. 2009. Open ERP -tietojärjestelmän soveltuvuus yksityisen sektorin tarpeisiin Suomessa. Insinööriyö. Kemi-Tornion Ammattikorkeakoulu.
- 14 Basics of developing a simple module in OpenERP. 2011. Verkkodokumentti. Zesty Beanz. <<http://zbeanztech.com/blog/basics-developing-simple-module-openerp>>. Luettu 1.12.2013.
- 15 Adding additional fields using a custom module in OpenERP 7. 2013. Verkkodokumentti. Pixelite Magazine. <<http://www.pixelite.co.nz/article/adding-additional-fields-using-custom-module-openerp-7>>. Luettu 1.12.2013.

Projektisuunnitelma - OpenERP:n käyttöönotto Premium Sähkö Oy:ssä

Alustava vaiheittainen etenemissuunnitelma

1. Asiakastietokanta, asiakassuhteet (CRM)
2. Työntekijät, aikataulut, tehtävälistat (Employee Directory + Project Management)
3. Varastonhallinta, materiaaliostot (Warehouse Management + Purchase Management)
4. Laskutus, raportit (Sales Management)
5. Ajoneuvot, ajopäiväkirjat (Fleet Management)
6. Muuta tarvittavaa

Projektin aloitus

Tavoite

Projektin tavoitteena on ottaa käyttöön Open ERP -toiminnanohjausjärjestelmä Premium Sähkö Oy:ssä. Onnistuessaan hyvin, käyttöönotto helpottaa ja selkeyttää yrityksen hallinnollisia toimintoja, sekä nopeuttaa käsittelyaikoja. Projektien etenemisestä on myös mahdollista saada järjestelmän kautta tietoa, joka selkeyttää erityisesti monivaiheisten projektien hallintaa.

Toiminnanohjausjärjestelmän tarkoitus on erityisesti tukea ja kehittää yrityksen johtamisprosessia.

Käyttöönotosta

Tarkoitus olisi ottaa toimintoja käyttöön vähän kerrallaan, jotta niiden käyttö pystytään omaksumaan kunnolla. Projektin edetessä täytyy selvittää, minkälaiset toiminnot tukevat parhaiten juuri sähköurakointiliikkeen tarpeita.

Voitaisiin syöttää todellinen informaatio tietokantaan heti alkuvaiheessa, jotta saadaan heti luotua tarvittavat linkitykset, esim. käyttäjärühmät ja käyttöoikeudet. Tällä tavalla

voidaan myös testata toimivuutta suurella määrällä todellista dataa ilman, että sitä tarvitsee myöhemmin poistaa.

Henkilöstö voidaan lisätä passiivisiksi käyttäjiksi jo suunnitteluvaiheessa, mutta kutsu käyttäjäksi lähetetään vasta ensimmäisen vaiheen käyttöönottovaiheessa, jolloin käyttäjät myös määrittelevät oman henkilökohtaisen salasansa.

Ensimmäisen käyttöönoton jälkeen uusien ominaisuuksien testaamiseen käytetään testiympäristöä, joka kopioidaan jokaisen uuden käyttöönoton jälkeen varsinaisesta tietokannasta.

1. CRM

Ensimmäisessä vaiheessa otetaan käyttöön CRM-toiminnot, tärkeimpinä luettelo asiakkaista, liidit (*leads*), mahdollisuudet, puheluloki ja suunnitellut puhelut. Näitä on tarkoitus käyttää asiakassuhteiden ylläpitoon, tilausten vastaanottoon ja tiedusteluihin vastaamiseen.

Liidit voidaan integroida yrityksen sähköpostiin eri tavoin, on tarpeen selvittää, mikä olisi tässä tapauksessa toimivin tapa.

Asiakkaille voidaan esimerkiksi määritellä tyyppi-informaatio, jonka perusteella ne voidaan niin ikään jakaa ryhmiin.

Myyntiprosessin on tarkoitus edetä siten, että potentiaalisen asiakkaan ottaessa yhteyttä, luodaan uusi liidi tai jos asiakas ottaa yhteyttä puhelimitse, luodaan uusi puhelu puhelulokiin.

Mikäli asia etenee, voidaan liidistä tai puhelusta luoda mahdollisuus, joka tarkoittaa potentiaalista tuottoa. Tuottoa pyritään arvioimaan myyntiprosessin edetessä todennäköisyysprosentilla, joka kasvaa vaihe vaiheelta, kunnes kyseinen mahdollisuus voiteaan tai hävitään.

Pienessä yrityksessä ei ole suurta erillistä myyntitiimiä, tästä syystä ei liene tarvetta suuresti eriyttää toimihenkilöiden käyttöoikeuksia CRM-moduulissa. Työntekijöille CRM-funktiot eivät tule näkymään ollenkaan.

2. Projektinhallinta

Projektinhallinta voidaan ottaa käyttöön hyvinkin aikaisessa vaiheessa. OpenERP:n ratkaisu on hyvinkin suoraviivainen. Luodaan projekti, joka sisältää yhden tai useampia tehtäviä. Jokaisella tehtävällä voi olla sama tai eri tekijä. Näin saadaan eroteltua esim. työntekijöiden ja toimihenkilöiden tehtävät erilleen saman projektin sisällä.

Tehtävien hallinta toimii samaan tapaan kuin mahdollisuuksien hallinta CRM-moduulissa: kun tehtävä etenee vaiheesta toiseen, se voidaan siirtää taulukossa eteenpäin, kunnes tehtävä on valmis.

Tehtävä-taulukon 4 vaihetta voisivat olla muodossa:

Aloittamatta – tehtävä on luotu, mutta sen tekijä ei ole vielä vastaanottanut sitä

Vastaanotettu – tehtävä on vastaanotettu, mutta ei vielä aloitettu

Kesken – tehtävä on aloitettu

Valmis – tehtävä on valmis, tarvittaessa voidaan käyttää myös Peruttu-vaihetta

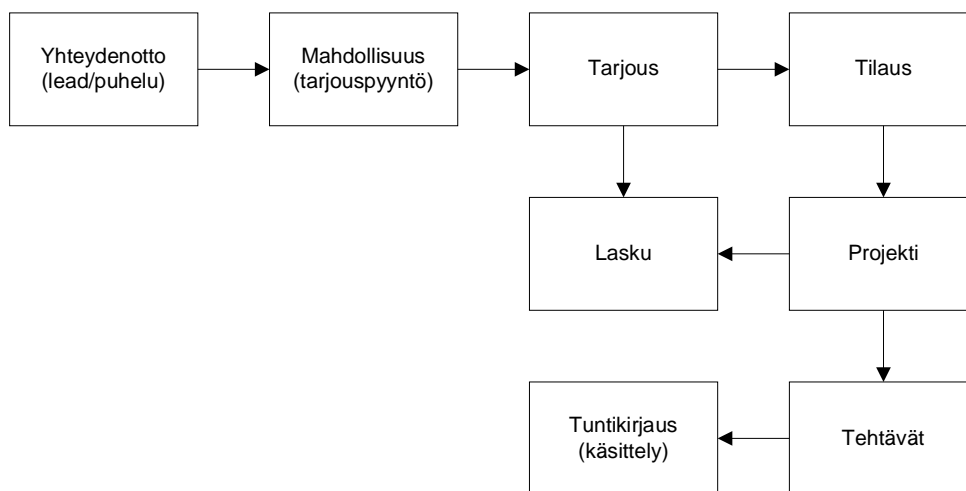
Projektinhallinta-moduuli on varsin yksinkertainen, mutta sen käyttö selkeyttää yrityksen toimintaa huomattavasti. Yksittäinen työntekijä näkee helposti, mitä hänen tulee milloinkin tehdä ja johtoporras on koko ajan kartalla, mitä kentällä tapahtuu.

Yksittäisen työntekijän on tarkoitus nähdä vain hänelle tai hänen tiimilleen suunnatut tehtävät, sekä mahdollisesti myös määrittelemättömät tehtävät, joissa ei ole määrättyä tekijää.

3. Hallinnollinen toimintaprosessi

Asiakkaan laskutus tapahtuu suoraan ennalta sovitun tarjouksen mukaan. Joissain tapauksissa kuitenkin projektin eteneminen vaikuttaa loppulaskun suuruuteen, esim. viivytykset.

- Tarjousprosessi (tuotepohjainen laskutus)
- Tuotepaketit (hintapaketit)
- käytetty materiaali (tuotekatalogi)
- asennustyötuotteet (vakiotunnit)
- suunnittelu (tuntiperusteinen/arvio)
- käsittelykulut (tuntiperusteinen/arvio).

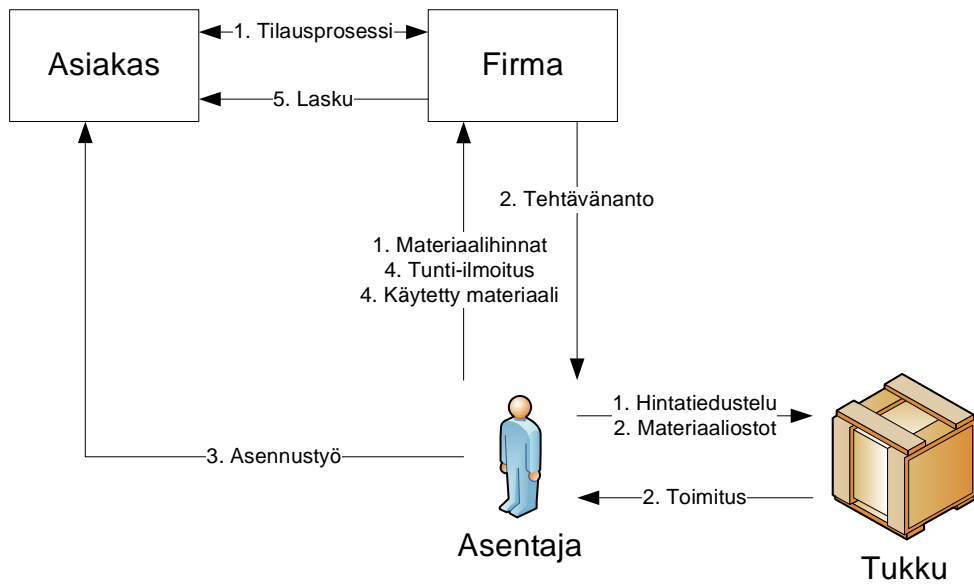


Kuva 1: Projektin eteneminen

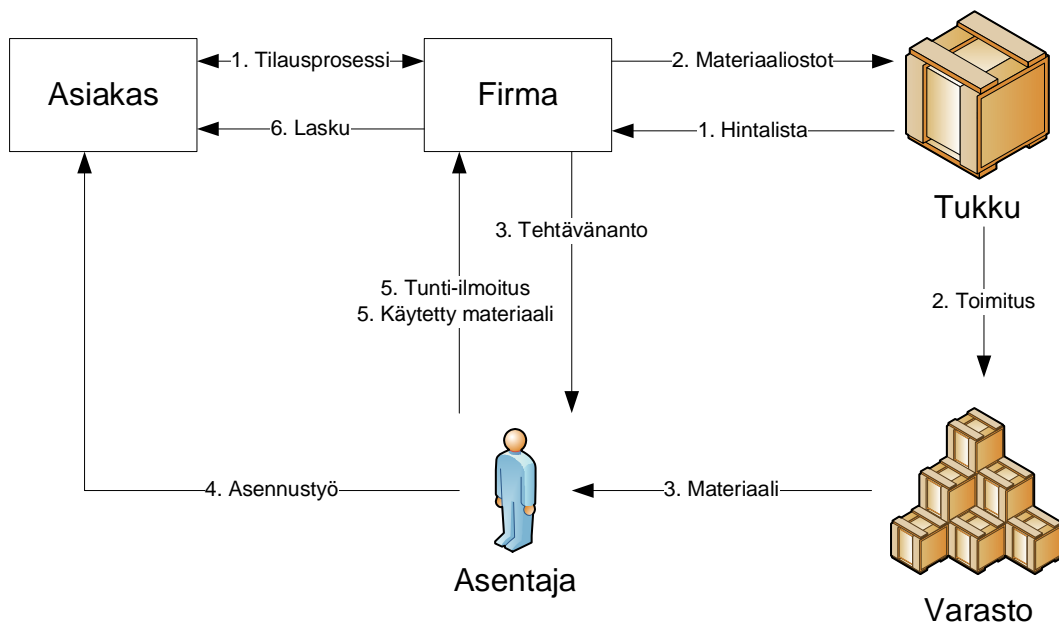
Tuntiperusteiset hinnat ovat arvioita, jotka perustuvat järjestelmään talletettuun dataan aikaisemmista saman kokoluokan projekteista. Käsittelykulut voivat koostua myös sovitusta vakiohinnoista.

4. Ulkoinen toimintaprosessi

Open ERP mahdollistaa laajemman toimintaprosessin käyttöönoton. Tällä hetkellä ongelmia aiheuttaa tavarantoimittajan kanssa asioiminen työntekijätasolla, sekä tästä johtuen tarjousvaiheen hankaluus (esim. erilliset hintatiedustelut). Yhtenäinen järjestelmä takaa sen, että kaikki relevantti tieto kulkee keskuskonttorin kautta ilman erillisiä kyselyitä työntekijätasolta, mikä tällä hetkellä vie työaikaa molemmilta osapuolilta.



Kuva 2: Toimintaprosessi nykyhetkellä



Kuva 3: Laaja toimintaprosessi

5. Aikataulus

Tarvitaan aikaa yleiseen ohjelmistoon tutustumiseen, sekä jokaisen vaiheen uusien toimintojen selvittämiseen.

Ennen jokaista käyttöönottoa täytyy olla valmiit toimintaohjeet ja selkeä eri toimintojen kuvaus saatavilla henkilöstölle. Ohjeiden tulisi käydä kohta kohdalta läpi käyttöprosessi eri tilanteissa. Saatavien käyttökokemusten pohjalta voidaan jatkuvasti oppia uutta OpenERP:n toimintaympäristöstä.

Alustava aikataulu

vk 35	Alustava sopiminen projektista
vk 36-39	Projektisuunnitelma
vk 39	Ilmoitus koululle
vk 40	Aloituspalaveri koululla
.	
.	
.	
vk ??	Vaihe 1 dokumentit
vk ??	Ensimmäinen käyttöönotto

OpenERP – Projektin aloituspalaveri**3.10.2013**

Läsnäolijat:

Antti Hartala	Työn tekijä
Teemu Autio	Työn ohjaaja, Premiumsähkö Oy
Tuomo Heikkinen	Ohjaava opettaja

Aika:

3.10.2013

Paikka:

Metropolia Albertinkatu, luokka B109

Sisältö:

Puhuttiin aluksi yleisistä firmaan liittyvistä asioista. Premiumsähkö on tuore vuonna 2012 perustettu yritys, joka työllistää tällä hetkellä 6 henkeä. Kyseessä on tytäryhtiö, jonka omistaa puoliksi Autio ja Kari Pääkkönen.

Emoyhtiön palveluksessa on tällä hetkellä 30 henkeä. Emoyhtiön omistaa kokonaan Autio.

Kasvu on ollut nopeaa, n. 300 % vuodessa. Tällä hetkellä tytäryhtiön liikevaihto on n. 300 k€. Emoyhtiön liikevaihto on 1,2 M€. Aution mukaan tavoite on nostaa Premiumsähkön liikevaihto samalle tasolle 3-5 vuoden sisällä.

Työn keskeisin tavoite on ottaa käyttöön toiminnanohjausjärjestelmä, joka palvelisi ensisijaisesti lyhyitä keikkatöitä. Tarkoitus olisi näin testata järjestelmää ensin pienessä mittakaavassa tytäryhtiössä.

Aution mukaan kaikista tärkeintä olisi päästä nopeasti eroon ylimääräisestä paperinpyörittelystä. Paperiset tunti-ilmoituslomakkeet täytyisi korvata sähköisellä lomakkeella, jota asentajien olisi helppo täyttää esim. käyttäen mukana kulkevaa tablettitietokonetta. Tähän olisi myös helppo pyytää asiakkaan kuittaus tehdyistä tunneista.

Autio totesi tilausprosessista, että se etenee lyhyissä keikkatöissä siten, että asiakkaan ottaessa yhteyttä sovitaan työn suorittamisesta ja annetaan karkea arvio hinnasta, joka kuitenkin myöhemmin tarkentuu loppulaskussa. Tähän karkeaan hinta-arvioon voisi käyttää OpenERP:n tarjouslomaketta, jossa valitaan tuotteita listaan.

Asiakas ja asentaja sopivat tehdyistä tunneista ja asentaja lähettää asiakkaan kuittamaan tunti-ilmoituslomakkeen, jossa myös ilmoitetaan käytetystä materiaalista.

Autio totesi suurempien urakoiden hinnoittelusta, että silloin laaditaan virallinen tarjous, joka pilkotaan osamaksueriin, joista laaditaan laskuja asiakkaalle urakan edetessä. Suuremman urakan tilausprosessi ei ainakaan vielä ole tähän projektiin liittyvää asiaa.

Autio totesi varaston roolista, että olisi hyvä jos asentaja voisi hakea suurimman osan materiaalista varastosta, mutta se on kuitenkin pyrittävä pitämään mahdollisimman pienenä, jottei se sido liikaa pääomaa. Tästä johtuen monesti joudutaan vastakin ajamaan tukun kautta asiakaskohteeseen.

Varastonhallinnassa täytyy myös ottaa huomioon, mitä tavaraa on autoissa. Esim. johdotkelat tulee olla aina autoissa valmiina, joten myös niiden kulutusta täytyisi seurata. Järjestelmään täytyy luoda päävaraston lisäksi jokaiselle (asennus) ajoneuvolle oma varastokanta.

Heikkinen totesi, että työn aihe on varsin hyvä ja monipuolinen, vaikkei se olekaan aivan koulutuksen ydinaluetta. Kyseessä on kuitenkin sähköurakointiliikkeen kannattavuuden ja työn tuottavuuden hallinta. Nämä asiat liittyvät yleisesti yrittäjyyteen ja taloudellisen osaamiseen, joiden hallinta on insinöörille tärkeää.

OpenERP – palaveri 16.1.2014

Läsnäolijat:

Antti Hartala

Teemu Autio

Harri Kallioniemi

Aika:

16.1.2014

Paikka:

Premiumsähkö Oy:n pääkonttori

Sisältö:

Tutkittiin järjestelmää ja puhuttiin mahdollisuuksista.

Ei voitu vielä ottaa käyttöön uutta projektinhallinta-laajennosta, mutta sovittiin, että seuraavalla viikolla pidettäisiin uusi palaveri, jossa käytäisiin läpi esimerkkitapaus ja sen eteneminen tietokannassa. Kokeilun perusteella voidaan selvittää edelleen korjaus ja laajennostarpeita.

Autio lupasi selvittää, kuinka voisin päästä suoraan käsiksi firman serverille (sijaitsee Saksassa) verkkoyhteyden kautta.

Serveri on Linux-pohjainen, joten Open ERP:n asennustiedostot ovat hieman eri tavalla järjestyneet, mutta löydettiin oikea addons-polku uusien kustomoitujen moduulien asennusta varten.

Addons-polku: `usr\lib\pymodules\python2.6\openerp\addons`.

Nousi esille valmis moduuli, josta voisi olla hyötyä järjestelmän tunti-ilmoitus-rakenteeseen: Web-kit report engine (asennus onnistui).

Firmassa on käytössä google kalenteri ja sähköposti. Täytyy selvittää, minkälaisia mahdollisuuksia on integroida näitä OpenERP:n kanssa (CRM ja projekti). Yritettiin asentaa yhtä hollantilaista google-moduulia, mutta se ei asentunut.

Järjestelmän asiakirjapohjien ei kuulemma tarvitse olla samanlaisia kuin tällä hetkellä käytettävät, kunhan ne sisältävät tarvittavat tiedot. Asiakirjapohjien muokkaaminen ja lisääminen järjestelmään on selvítettävä ja laadittava ohje.

Päätettiin, että projektirakenne toteutetaan siten, että yksi kokonaisprojekti lisätään vuosittain/kvartaaleittain, tai jopa kuukausittain. Kokonaisprojektin alle avataan tehtäviä, eli työkeikkoja.

Kallioniemi totesi tärkeäksi, että asiakas-lomakkeelta on suora pääsy asiakkaalle avattuihin tehtäviin. Todettiin, että ko. yhteys löytyy järjestelmästä valmiina

OpenERP – palaveri 3.2.2014

Läsnäolijat:

Antti Hartala

Teemu Autio

Harri Kallioniemi

Aika:

3.2.2014, klo 18

Paikka:

Premiumsähkö Oy:n pääkonttori

Sisältö:

Ennen palaveria oltiin saatu yhteys firman serverille, jota kautta saatiin ladattua lisämoduuli omaan erilliseen kansioon.

Testattiin projektinhallinta-laajennosta, joka todettiin lupaavaksi, mutta pieniä korjauksia täytyy vielä tehdä.

Korjattiin uusien moduulien asennusongelma. Johtui siitä, että clientillä ei ollut kirjoitusoikeutta serverin addons-kansioon.

Suurimpana ongelmana/puutteena nousi esille kalenteritoiminnot. Kalenterin pitäisi olla yhteinen, josta työnjohtaja näkisi kaikkien työntekijöiden työtilanteen, eli tehtävät pitäisi näkyä kalenterissa. Myös tehtävien luominen kalenterista olisi hyvä olla mahdollista. Ladattiin uusi moduuli "Shared Calendar", joka vaikuttaa lupaavalta, mutta sitä täytyy testata ja selvittää tarkemmin sen toiminta. Autio totesi, että firman käyttämän Google- ja Openerp-kalenterien synkkaaminen keskenään onnistuu, kunhan saadaan ensin Openerp:n kalenteri toimimaan halutulla tavalla.

Tuotelista on päivitetty, tällä hetkellä se sisältää jopa yli 17000 sähkönimikettä. Näistä Autio haluaisi mahdollisimman monta tietokantaan. Tuotetietokanta saattaa kuitenkin hidastua näin suuresta määrästä tuotenimikkeitä. Päätettiin, että ensisijaisesti O-toimitusajan tuotteet pyritään ajamaan tietokantaan (uudessa listassa on toimitusaikasarake).

Olellaiset raportit pitää saada kuntoon, jotka ovat: laskupohja, sekä tunti-ilmoitus.

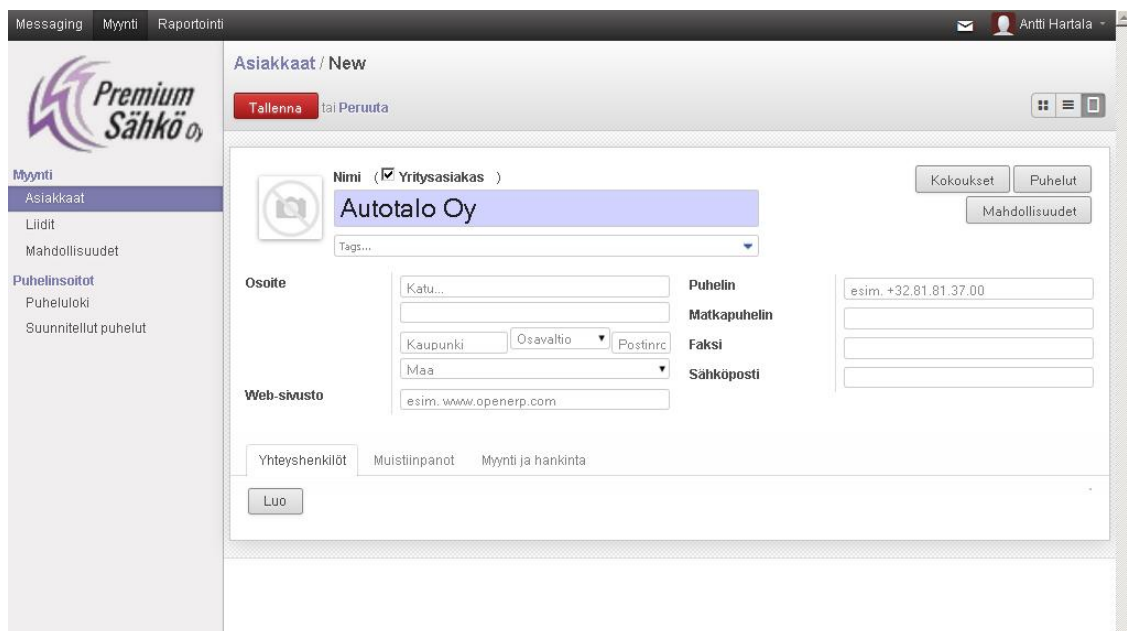
Tunti-ilmoitukselle kaavailtiin paikkaa asiakkaan allekirjoitukselle. Allekirjoitus toteutettaisiin järjestelmästä käsin siten, että tehtävä-valikossa painetaan allekirjoitus-nappia, joka avaa erillisen ulkoisen linkin, johon piirretään allekirjoitus (padilla). Linkki tallentaa allekirjoituksen kuvana, joka ladataan tietokantaan ko. tehtävälle. Tehtävästä saa auki raportin, jossa allekirjoitus näkyy. Tämä tehtäväraportti laitetaan laskun liitteeksi asiakkaalle.

Esimerkkikuvia OpenERP-järjestelmästä



The image shows a login form for OpenERP. At the top, the logo reads "OpenERP" with "OPEN SOURCE BUSINESS APPLICATIONS" underneath. The form fields are: "Nimi" (Name) with the value "Antti Hartala", "Username (Email)" with the value "anttiha", "Salasana" (Password) with masked characters, and "Confirm Password" with masked characters. Below the fields are two buttons: "Reset password" and "Paluu kirjautumiseen". At the bottom, it says "Powered by OpenERP".

Kuva 1. Kirjautumisikkuna



The image shows a screenshot of the OpenERP web interface for creating a new customer. The browser address bar shows "Antti Hartala". The page title is "Asiakkaat / New". There are buttons for "Tallenna" (Save) and "Peruuta" (Cancel). The form is for a customer named "Autotalo Oy" (checked as "Yritysassiakas" - Business customer). The form fields include: "Osoite" (Address) with sub-fields for "Katu..." (Street), "Kaupunki" (City), "Osavaltio" (Region), "Postinrc" (Postal code), and "Maa" (Country); "Web-sivusto" (Website) with the value "esim. www.openerp.com"; "Puhelin" (Phone) with the value "esim. +32.81.81.37.00"; "Matkapuhelin" (Mobile phone); "Faksi" (Fax); and "Sähköposti" (Email). There are also buttons for "Kokoukset" (Meetings), "Puhelut" (Calls), and "Mahdollisuudet" (Possibilities). At the bottom, there are tabs for "Yhteyshenkilöt" (Contacts), "Muistiinpanot" (Notes), and "Myynti ja hankinta" (Sales and Purchasing), and a "Luo" (Create) button.

Kuva 2. Asiakkaan lisäyslomake

Liidit / New

Tallenna tai Peruuta

Uusi

Aihe
Tarjouspyyntö

Aikatauluta/rekisteröi soitto

Yrityksen nimi	<input type="text"/>	Kontaktin nimi	<input type="text"/>
Asiakas	Autotalo Oy	Sähköposti	<input type="text"/>
Osoite	Katu...	Toiminto	<input type="text"/>
	Helsinki Osavalt	Puhelin	<input type="text"/>
	Postinro	Matkapuhelin	<input type="text"/>
	Maa	Faksi	<input type="text"/>
Myyjä	Antti Hartala	Prioriteetti	Normaali
Myyntitiimi		Kategoriat	

Muistiinpanot Lisätiedot

Kuva 3. Uuden liidin lisäyslomake

Liidit / Tarjouspyyntö

Muokkaa Luo More

Muunna mahdollisuudeksi Hylkää

Uusi

Tarjouspyyntö

Aikatauluta/rekisteröi soitto

Yrityksen nimi	Autotalo Oy	Kontaktin nimi	.
Asiakas		Sähköposti	
Osoite	Helsinki	Toiminto	
		Puhelin	
		Matkapuhelin	
		Faksi	
Myyjä	Antti Hartala	Prioriteetti	Normaali
Myyntitiimi		Kategoriat	

Muistiinpanot Lisätiedot

Kuva 4. Liidin muuntaminen mahdollisuudeksi

Puheluloki

Tallenna tai Peruuta

<input type="checkbox"/>	Päiväys	Soiton yhteenvedo	Yhteyshenkilö	Vastuullinen
<input type="checkbox"/>	12/11/2013 20:3		Autotalo Oy	Antti Hartala

Kuva 5. Puhelun lisäys

Suunnitellut ... / New

Tallenna tai Peruuta

Merkitse soitetuksi Peruuta soitto Vahvistettu Soitettu

Muunna mahdollisuudeksi Aikatauluta muu soitto Ajoita tapaaminen

Soiton yhteenvedo

Puhelin

Päiväys 12/11/2013 20:39:40 Vastuullinen Antti Hartala

Kesto 00:00 Myyntitiimi

Yhteyshenkilö

Kategoria Matkapuhelin

Prioriteetti Normaali Liidi/mahdollisuus

Kuvaus...

Kuva 6. Suunnitellun puhelun lisäys