

Tapio Kaartinen

LUOVANET ASIAKKUUKSIENHALLINTAOHJELMISTO

Opinnäytetyö

Kajaanin ammattikorkeakoulu

Tradenomikoulutus

Kevät 2007



**Kajaanin
ammattikorkeakoulu**

OPINNÄYTETYÖ TIIVISTELMÄ

Koulutusala Luonnontieteiden ala	Koulutusohjelma Tietojenkäsittely
Tekijä(t) Kaartinen Tapio	
Työn nimi Luovanet Asiakkuuksienhallintaohjelmisto	
Vaihtoehtoiset ammattiopinnot Ohjelmointi	Ohjaaja(t) Piirainen Veli-Pekka
	Toimeksiantaja Luovanet Oy
Aika Kevät 2007	Sivumäärä ja liitteet 41+0
<p>Tämä opinnäytetyö tehtiin Luovanet Oy:n toimeksiannosta. Toimeksiantona oli kehittää web-sovellus, jolla hallitaan Luovanet Oy:n ja Luovaliike Oy:n asiakkuustietoja. Sovelluksen käyttäjinä ovat Luovanetin ja Luovaliikeen henkilöstö.</p> <p>Sovelluksen toteuttamiseen käytettiin C# ohjelmointikieltä ja ohjelmointialustana toimi ASP.NET 2.0. Ohjelmointiympäristönä oli Visual Studio 2005 ja ohjelmistossa käytettävänä tietokantaohjelmistona oli SQL Server 2005. Osa ohjelmistossa käytettävistä tiedoista on XML-rakenteen muotoista ja tallennetaan SQL Server -tietokantaohjelmistosta löytyviin XML-kenttiin sellaisenaan.</p> <p>Toteutettavasta ohjelmistosta ei ollut aikaisempaa versiota olemassa, joten työn tekeminen aloitettiin ihan alusta lähtien. Projektin alussa pidettiin palaveri, jossa pääasiassa määriteltiin ohjelmistoon tulevat toiminnalliset vaatimukset. Toiminnallisten vaatimusten selvityksessä alkoi ohjelmiston suunnittelu, jonka jälkeen vasta varsinainen ohjelmoiminen voitiin aloittaa. Ohjelmiston valmistuessa toteutettiin vielä käyttäjän syötteiden testaaminen. Kyseinen testi paljasti ohjelmasta selviä virheitä, jonka takia ohjelmaa jouduttiin vielä korjaamaan, ennen kuin se oli täysin valmis.</p> <p>Toteutuneen ohjelmiston toiminnalliset vaatimukset vastaavat vaatimusmäärittelyn vastaavia asioita. Suunnitteluraporttiin määriteltäviin asioihin nähden toteutunut sovellus ei kuitenkaan ihan täysin vastaa määrittelyä. Syynä tähän on se, että ohjelmistosta saatiin parempi, kun suunnitteluraporttia ei noudatettu orjallisesti. Suurin muutos määrittelyyn nähden koski toteutuneen ohjelmiston tietokantarakennetta.</p> <p>Opinnäytetyöhön käytettävän ajan rajallisuuden takia ohjelmistoon ei asetettu kaikkia toiminnallisia vaatimuksia, joita siihen alussa oli tarkoitus laittaa. Toteutuneen ohjelmiston tietokantarakenne on kuitenkin hyvin yksinkertainen, joten ohjelmiston jatkokehittäminen on täysin mahdollista ja poistetut vaatimuksetkin voidaan lisätä ohjelmistoon myöhemmin.</p>	
Kieli	Suomi
Asiasanat	asiakkuuksienhallinta, asiakkuudenhallinta, tiedonlouhinta
Säilytyspaikka	<input checked="" type="checkbox"/> Kajaanin ammattikorkeakoulun Kaktus-tietokanta <input checked="" type="checkbox"/> Kajaanin ammattikorkeakoulun kirjasto

School Business	Degree Programme Data processing
Author(s) Kartinen Tapio	
Title Luovanet Customer Relationship Management Program	
Optional Professional Studies Programming	Instructor(s) Pirainen Veli-Pekka
	Commissioned by Luovanet Oy
Date Spring 2007	Total Number of Pages and Appendices 41+0
<p>The purpose of the thesis commissioned by Luovanet Oy was to build a web program for managing customer information of Luovanet Oy and Luovaliike Oy. The users of the program are the personnel in the Luovanet and Luovaliike.</p> <p>The program was built by using C# programming language and the programming platform ASP.NET 2.0. The programming environment was Visual Studio 2005 and the database software used for saving data was SQL Server 2005. Some of the data processed by the program is in the XML-format and it is saved directly to the XML-field found from the SQL Server database software.</p> <p>There were not any earlier versions of the program, therefore the work was started by determining all the functional requirements first. Designing the new program was started after that and when the design phase was finally finished, the programming work was started. The last phase for the completed program was to test the data inputs of the users. This test revealed some critical errors in the program which had to be fixed before the program was fully completed.</p> <p>The finished program has all the functional requirements determined in the requirements manual. Some of the requirements mentioned in the design report cannot be found in the finished program. The main reason is that the program is much better when the design report was not followed slavishly. The biggest modification from the design report to the finished program was effected in the database structure.</p> <p>Some of the functional requirements were removed from the program, because the time for making the thesis was so short. The database structure is really simple in the finished program. Therefore, the removed functions could be added to the program later. In addition, further development of the program is now possible because the database structure is so simple.</p>	
Language of Thesis Finnish	
Keywords	customer relationship management, CRM, data mining
Deposited at	<input checked="" type="checkbox"/> Kaktus Database at Kajaani University of Applied Sciences <input checked="" type="checkbox"/> Library of Kajaani University of Applied Sciences

TIIVISTELMÄ

ABSTRACT

SISÄLLYS

1	JOHDANTO.....	1
2	ASIAKKUUKSIENHALLINTA	3
2.1	Asiakkuuksienhallinta lyhyesti	4
2.2	Asiakkuuksienhallinnan peruskäsitteet	6
2.2.1	Asiakkaan määrittely.....	7
2.2.2	Asiakkuuksien elinkaari	9
2.2.3	Asiakkaan arvontuotanto.....	12
2.3	Asiakkuuksienhallinnan osa-alueet	13
2.3.1	Operatiivinen asiakkuuksienhallinta	13
2.3.2	Analyttinen asiakkuuksienhallinta	14
2.4	Tietojärjestelmien suunnittelun lähtökohdat ja vaatimukset.....	14
2.4.1	Käsitteet, käsittemallit ja tietoyhtenäisyys	14
2.4.2	Tietotekniset lähtökohdat tietojärjestelmien kehittämiseen	16
2.4.3	Tietoteknologian mukautuminen.....	18
2.5	Asiakkuustietoon liittyvät tekijät	19
2.5.1	Tiedon määrä ja sen lisääminen	19
2.5.2	Tiedon sisältö ja rakenne.....	20
2.5.3	Tietovarastot (Data warehouse)	21
2.5.4	Paikallisvarasto (Datamart).....	22
2.5.5	Tiedonlouhinta (Data mining).....	23

3	LUOVANET ASIAKKUUKSIENHALLINTAOHJELMISTO.....	26
3.1	Taustaa	26
3.2	Ohjelmointikielet ja työkalut.....	27
3.3	Vaatimusmäärittely	27
3.3.1	Tietokanta.....	28
3.3.2	Käyttöliittymä.....	28
3.4	Työprosessi.....	29
3.4.1	Suunnittelu	29
3.4.2	Ohjelmointi.....	30
3.4.3	Testaus.....	31
3.5	Toteutunut ohjelmisto	32
3.5.1	Käyttöliittymä.....	32
3.5.2	Tietokanta.....	35
	POHDINTA	38

LÄHTEET

SYMBOLILUETTELO

Asiakaskannattavuus	Asiakaskannattavuudella tarkoitetaan asiakkuuden tarkastelujaksolla tehtyjen ostojen ja niihin liittyvien yrityksen kustannusten välistä erotusta. (Mäntyneva 2000, 125.)
CRM	Customer Relationship Management eli asiakkuuksienhallinta.
Tarjooma	Tarjooma on asiakkaalle ostettavaksi tarkoitettu kokonaisuus. Se voi koostua esimerkiksi tavaroista ja palveluista. (Saarikkomäki 2006, Käsitteet.)
XML	eXtensible Markup Language on rakenteellisten dokumenttien merkintäkieli, joka kuvaa tiedon rakennetta ja merkitystä, muttei sen muotoilua. (Helia 2007.)

1 JOHDANTO

Opinnäytetyön toimeksiantajana on Luovanet Oy, jonka palvelut kattavat kaikki www-tuotannon osa-alueet graafisesta suunnittelusta, teknisestä toteutuksesta, ohjelmiston- ja sisällöntuotannosta, teknisestä tuesta ja ylläpidosta aina kolutukseen ja konsultointiin saakka. Yrityksen toteuttamat ohjelmistot ovat Internet-selaimella toimivia sovelluksia. (Luovanet Oy 2007.)

Opinnäytetyön aiheena on Luovanet Asiakkuuksienhallintaohjelmiston suunnittelu ja ohjelmointi. Ohjelmiston tarkoituksena on helpottaa Luovanet Oy:n ja Luovaliike Oy:n asiakkaiden perustietojen ylläpitämistä sekä asiakkaalta laskutettavien tuotteiden ja palveluiden perimistä. Ohjelmiston käyttäminen hoidetaan Internet-selaimella.

Työn toteuttamiseen käytettiin Visual Studio 2005 ohjelmointiympäristöä. Ohjelmointikielenä oli C# ja ohjelmointialustana toimi ASP.NET 2.0. Ohjelmistossa käsiteltävien tietojen tallennukseen käytettiin SQL Server 2005 tietokantaohjelmistoa. Työn toteuttamisessa täysin uutena asiana tuli opetella XML-tietojenkäsittely, koska osa ohjelmassa käytettävästä tiedosta oli XML-muotoista.

Opinnäytetyön teoriaosuudessa on käsitelty asiakkuuksienhallintaan liittyvää käsitteistöä tarkemmin. Teoriaosuudessa pyritään antamaan laajempi näkemys asiakkuuksienhallintaan ja siihen liittyviin asioihin. On kuitenkin huomattava, että asiakkuuksienhallinta on erittäin laaja käsite, joten kaikkea siihen liittyvää ei ole mahdollista käsitellä tässä opinnäytetyössä kovin tarkasti. Sen vuoksi teoriaosuudessa on pyritty keskittymään asiakkuuksienhallinnan keskeisimpiin asioihin ja antamaan siitä pelkästään pohjatietoutta.

Asiakkuuksienhallintaan liittyy keskeisenä osana tietotekniikka, koska asiakkaista kerättävän tiedon käsitteleminen ja analysoiminen on huomattavasti kätevämpää tietotekniikan avustamana. Näin ollen teoriaosuudessa käsitellään etupäässä asiakkuuksienhallinnan tietoteknistä puolta ja tähän liittyviä asioita. Teoriaosuudessa käsitellään myös asiakkuustietoihin liittyviä asioita, kuten tiedon sisältöä ja rakennetta, tiedon määrää ja sen lisäämistä, tietovarastoja ja tiedonlouhintaa.

Asiakkuuksienhallinnasta kiinnostuneiden on kohtuullisen helppoa löytää Internetistä asiaa käsittelevää lisäaineistoa, jos halutaan saada tarkempi näkökanta tässä opinnäytetyössä käsiteltyihin asioihin nähden. Syöttämällä Internetin hakukoneisiin muun muassa seuraavanlaisia sanoja kuten asiakkuuksienhallinta, asiakkuudenhallinta, asiakkuusajattelu, customer relationship management ja crm. Suomen kielisissä hakusanoissa kannattaa kokeilla yhteen ja erikseen kirjoitettuja sanamuotoja. Asiakkuuksienhallintaa käsittelevää materiaalia löytyy myös painetussa muodossa.

2 ASIAKKUUKSIENHALLINTA

Customer Relationship Management (CRM) eli asiakkuuksienhallinta määritellään monella eri tavalla. Yleensä tähän vaikuttaa määrittelijän lähestymistapa asiaan ja kuinka asiaa halutaan käsitellä. Suomen kielisessä aineistossa asiasta voidaan hyvinkin puhua muun muassa seuraavilla nimillä: asiakkuuksienhallinta, asiakkuudenhallinta, asiakassuhteiden johtaminen, Customer Relationship Management tai vain lyhyesti CRM. On siis henkilöstä kiinni mitä termiä käytetään ja miten se ymmärretään.

Asiakkuuksienhallinta termiä käytetään laajasti myyntiin ja markkinointiin liittyen. Valitettavasti terminologia ei ole kovin yksiselitteistä. Jollekin se voi olla myyntityön tehostamista, toiselle se taas on raportointia ja kolmannelle se jo onkin asiakaspalvelun parantamista ja asiakassuhteen hallintaa. Verkkopalvelut voivat myös olla osa yrityksen asiakkuuksienhallinnan kokonaisuutta. (TIEKE ry 2007 a.)

Asiakkuuksienhallinnasta on kuitenkin tullut entistä tärkeämpi toiminto yrityksissä, sillä sen vaikutus kilpailukykyyn on havaittu. Saatetaan myös käyttää termiä asiakkuusajattelu, eli kuinka saadaan asiakas tuottamaan lisäarvoa itselleen. Tämän edellytyksenä on asiakkaan toimintatapojen tarkka tunteminen. Toisaalta se voi tarkoittaa myös sitä, että miten yritys voi muokata sen omia prosessejaan palvelemaan paremmin sen parhaita asiakkaitaan. Asiakkuuksienhallinta perustuu asiakastietojen keräämiseen ja hyväksikäyttöön. Kerätyn tiedon perusteella pyritään muun muassa tunnistamaan yritykselle parhaat asiakkaat, ohjaamaan markkinointia, myyntiä ja palvelua siten, että asiakkaat pystyttäisiin pitämään yrityksen asiakkaina pitkään ja asiakkaan uskollisuutta yritystä kohtaan lisäämään. (TIEKE ry 2007 b.)

Nykyään on siirrytty palvelemaan asiakkaita yhä enemmän asiakaskohtaisesti. Kaikilla toimialoilla ei tietenkään ole perusteltua tai kannattavaa hallita asiakkaita asiakaskohtaisesti, mutta asiakkuuksia voidaan hallita myös asiakassegmenteittäin. Asiakkuuksienhallinnan menetelmillä asiakkaista voidaan määritellä sopivia asiakassegmenttejä. Näin ollen voidaan esimerkiksi markkinointi kohdentaa vain tiettyyn asiakassegmenttiin tai tietylle asiakassegmentille voidaan myöntää tarjous jostain tuotteesta tai palvelusta muiden asiakkaiden tätä tietämättä.

Asiakkuuksienhallinnan etuina on kyky hallita asiakasinformaatiota sekä tunnistaa jokaisen asiakkaan ominaispiirteet. Tavoitteena on myös palvelua tai tuotetta tarjoavan yrityksen toiminnan kannattavuuden sekä asiakastyytyväisyyden tason tavoitteiden mukaisuus. Asiakkuuksienhallinnan ansiosta asiakastiedon saatavuuden sekä ylläpidon optimointi kyetään hoitamaan myös monimutkaisissa organisaatorakenteissa. (Vuopio 2002, 19.)

Yritykset ovat vaativassa tilanteessa, kun yritetään muuttaa tuote- ja markkinointikeskeistä toimintamallia asiakaslähtöiseksi. Yleisesti on käsitys, että yrityksen toiminnan muuttaminen asiakaslähtöiseksi on melkein mahdotonta sekä erittäin kallista. Yritysjohdon näkökulmasta edellytyksenä on vanhojen oppien sekä kokemusten kautta tutuksi tulleen ajattelutavan muuttamista. (Vuopio 2002, 21.)

2.1 Asiakkuuksienhallinta lyhyesti

Asiakkuuksienhallinnalle asetetut tavoitteet vaihtelevat yrityksittäin ja myös toimialoittain. Kaikilla yrityksillä ei välttämättä ole tarvetta toteuttaa äärettömän monipuolista asiakkuuksienhallintaa. Esimerkiksi pienemmillä yrityksillä asiakkuuksienhallinta voi olla vaikka vain asiakaskontaktienhallintaa. Luultavasti yleisimmin asiakkuuksienhallintaa käytetään apuna suuremmissa yrityksissä markkinointitarkoituksissa, mutta sitä voidaan hyödyntää myös muillakin tavoilla.

Asiakkuuksienhallinnan toteutus vaatii yritykseltä enemmän voimavaroja kuin aiemmin, yleensä tämä tarkoittaa osaamistarpeiden lisääntymistä. Tähänkin asiaan toki vaikuttaa, että kuinka monipuolista asiakkuuksienhallinnasta ollaan tekemässä, onko ennestään jo jonkin asteista asiakkuuksienhallintaa olemassa ja ollaanko aikaisempaa järjestelmää laajentamassa.

Tietysti myös yrityksen koolla on vaikutusta, sillä pienemmässä yrityksessä ei välttämättä ole järkevää ottaa kovin kattavaa asiakkuuksienhallintaa käyttöön, jos siitä saatavat hyödyt eivät ole sen vaivan arvoista.

Asiakkuuksienhallinnan tavoitteena on korostaa yrityksen asiakkuuksien määrätietoista johtamista. Asiakkuuksien korostaminen tarkoittaa asioiden tarkastelemista asiakkaan näkökulmasta. Yhtenä tavoitteena on saada asiakas osaksi palvelua tai tuotetta tuottavan yrityksen toimintaa. Asiakkaan sitoutumisella osaksi yrityksen toimintaan voidaan nähdä asiakassuhteen kustannukset. Vanhan asiakkuuden ylläpitäminen on yleensä kustannuksiltaan pienempää, mitä uuden asiakkuuden hankkiminen. Asiakkuuksienhallinta korostaa yrityksen aktiivista päätöksentekoa siltä osalta, että minkä asiakkuuksien kehittämiseen jatkossa halutaan panostaa. Kaikki asiakkuudet kun eivät ole saman vaivan arvoisia. (Mäntyneva 2000, 9, 11; Vuopio 2002, 19.)

Oman hankaluuden asiakkuuksienhallintaan tuo asiakkaiden halu olla ulkona kaikesta markkinointiviestinnästä ja kaiken tämän lisäksi ei myöskään haluta käytettävän tietoja mihinkään markkinointitarkoitukseen. Tämä rajoittaa paljon markkinointiviestinnän toteutusta ja menestyksellisyyttä. Asiakkaiden ymmärtämispyrkimyksen tavoitteena on juuri selvittää asiakkaiden ostoaikomuksia. Asiakkuuksienhallinta pyrkii lisäämään asiakkaiden ostoaikomusta nimenomaan sinun yritykseltäsi eikä kilpailevilta toimittajilta. (Mäntyneva 2000, 11.)

Asiakkuuksienhallinnan tuomat edut voidaan jakaa kahteen ryhmään. Ensinnäkin asiakkuuksienhallinta lisää yrityksen tietämystä ja ymmärrystä asiakkaista ja siitä miksi he ostavat. Toisaalta asiakkuuksienhallinta lisää myynnin ja markkinoinnin tehokkuutta ja vaikuttavuutta, mikä taas osaltaan vahvistaa markkinoinnin kokonaiskannattavuutta. On hyvä muistaa, että myynnin ja markkinoinnin tehokkuus ja vaikuttavuus eivät todennäköisesti parane itsestään vaan vain määrätietoisella asiakaslähtöisen toimintamallien ja prosessien kehittämisellä. Asiakkaiden lisääntyvä ymmärrys parantaa puolestaan asiakkuuksiin ja niiden johtamiseen liittyvien strategisten analyysien laatua ja sisältöä. Asiakaskannattavuus on entistä paremmin linkitettävissä markkinoinnin kampanjoihin, jotka puolestaan kyetään kohdentamaan yhä tarkemmin asiakkaille. Toisaalta tapahtumalähtöisyys edistää markkinointiviestinnän toteutuksessa osaltaan sitä, että yritys pystyy paremmin vastaamaan asiakkaiden tarpeisiin. (Mäntyneva 2000, 12.)

Yksi asiakkuuksienhallinnan vahvuuksista on mitattavuuden huomattava lisääntyminen. Mitattavuuden lisääntyminen mahdollistaa uusien vaihtoehtoisten menettelytapojen testaamisen ja kehittämisen markkinoinnissa käytettävien kilpailukeinojen mahdollisimman hyvän yhdistelmän muodostamiseksi. (Mäntyneva 2000, 12.)

Asiakkuuksienhallinnan menestymiseen vaaditaan yritykseltä entistä enemmän voimavaroja. Tässä yhteydessä huomio usein keskittyy osaamistarpeiden lisääntymiseen, informaatioteknologian hyväksikäyttöön, organisaation ja toiminnan rakenteen mukauttamiseen asiakkuuksienhallintaa tukevaksi sekä asiakkuuksiin liittyvän kokonaisvaltaisen ajattelun ja kulttuurin jalkautumiseen. (Mäntyneva 2000, 14.)

Asiakkuuksienhallinnan käyttöalueet puolestaan kohdentuvat erityisesti sellaisiin kilpailuihin markkinoihin, joilla yksittäisen asiakkuuden johtaminen ei ole perusteltua tai kustannustehokasta. Asiakkuuksienhallinta on merkittävästi enemmän kuin vain pelkästään teknologian tukema tapa toteuttaa markkinointia. Käytännössä kyse on siis asiakkaiden tarpeiden ja niiden mahdollisimman tehokkaan toteuttamisen jatkuvasta oppimisesta. (Mäntyneva 2000, 14.)

2.2 Asiakkuuksienhallinnan peruskäsitteet

Asiakkuuksienhallinnan kannalta on hyvä selvittää muutamia peruskäsitteitä, mitkä on hyvä ottaa huomioon, kun mietitään asiakkuudenhallintaa. Asiakkuudenhallinnassa keskeisessä roolissa ovat asiakkaasta kerättävät tiedot. Näin ollen on tärkeää määritellä asiakaskäsite oikein, että osataan kerätä tietoa oikeista asiakkaista. On myös hyvä tietää asiakkaan vaiheesta elinkaarellaan, jotta asiakkuutta voidaan hoitaa tilanteeseen sopivilla menetelmillä. Asiakkuuden kehittämisen kannalta on hyvä tietää asiakkaan arvontuotantoon vaikuttavia tekijöitä.

2.2.1 Asiakkaan määrittely

Asiakkuuksienhallinnassa on ensiarvoisen tärkeää määritellä asiakaskäsite. Asiakaskäsitteen määrittelyllä yrityksen tiedotus osataan suunnata oikeille kohderyhmille ja osataan myös kerätä tietoa oikeista henkilöistä ja organisaatioista yrityksen toiminnan kehittämiseksi. Asiakkaiden määrittelyminen muodostaa myös perustan eri tyyppisille asiakkuusstrategioille sekä se auttaa tunnistamaan millainen vuorovaikutussuhde asiakkaan ja yrityksen välille on pyrittävä luomaan. (Lehtonen & Puhto 2002, 13.)

Yleensä asiakkaalla on tarkoitettu henkilöä tai organisaatiota, johon on syntynyt asiakassuhde. Tämän tyyliässä ajatusmallissa asiakkuus kuitenkin edellyttää asiakassuhteen syntymistä eli ostamista. Näkökulmaa laajentamalla asiakkaaksi voidaan määritellä jokainen, jonka kanssa yritys on ollut kontaktissa. Näin ollen asiakkaalla tarkoitetaan kaikkia, jotka:

- ovat ostaneet eli on syntynyt asiakassuhde
- potentiaalisina asiakkaina kuuluvat segmenttiin ja ovat esimerkiksi olleet kontaktissa yrityksen myyntihenkilöön, mutta eivät kuitenkaan ole käyttäneet yrityksen palveluja
- ovat olleet yrityksen palveluiden kanssa välillisessä kontaktissa. (Lehtonen & Puhto 2002, 13.)

Yrityksen asiakkaita määriteltäessä on otettava huomioon, että kaikilla yrityksen toiminnoilla on omat asiakkaansa ja tietotarpeensa. Hyvin usein käy kuitenkin niin, että määrittely tehdään virheellisesti yhden yksittäisen toiminnon tai tietojärjestelmän näkökulmasta. (Lehtonen & Puhto 2002, 13.)

Asiakaskäsitteeseen on yhtenä näkökulmana esitetty, että yrityksen asiakkaita ovat kaikki ne osapuolet, jotka ovat jollain tavoin osallistuneet yrityksen tarjoamien tuotteiden hankintaan. Tämän määrittelyn mukaan asiakkaana voi olla yksilö, organisaatio sekä myös yksilöiden muodostama eturyhmä. Perusajatuksena siis on, että asiakaskäsitteeseen sisältyy muitakin osapuolia kuin ne, jotka varsinaisen tuotteen jossain vaiheessa omistavat. Kaiken lisäksi, että yritykselle määriteltäisiin pelkästään asiakkaat, voidaan eri asiakastyypeille määrittää

painoarvot sen perusteella, mikä on asiakkaan merkitys yrityksen liiketoiminnalle. Asiakaskäsitteen selventämiseksi on esitetty jaottelu, jossa asiakkaat jaetaan kahdeksaan luokkaan seuraavanlaisesti:

- Välittäjä (agent) – Välittäjät eivät osta tuotetta vaan välittävät sen myyjältä ostajalle ja kontrolloivat näin myyjän ja ostajan välistä suhdetta. Esimerkiksi vakuutuslalla on hyvin yleistä käyttää välittäjää. Välittäjän valitessa itse edustamansa yritykset, on yrityksen kyettävä vakuuttamaan välittäjä tuotteensa ylivertauudesta. Näin ollen välittäjä on yhtäläillä yrityksen asiakas kuin loppukäyttäjäkin, jonka tarpeet on tunnistettava ja pyrittävä täyttämään.
- Edunsaaja (beneficiary) – Edunsaaja ei välttämättä osallistu itse tuotteen hankintaprosessiin, mutta hyötyy kuitenkin jollain tavalla hankinnasta. Yritykset kuitenkin yleensä luokittelevat edunsaajat asiakkaiksi, jotka voivat tulevaisuudessa luoda merkittävää lisäarvoa yritykselle.
- Maksaja (bill payer) – Maksaja on sellainen asiakas, joka maksaa tuotteen, mutta ei itse välttämättä käytä tuotetta. Asiakkuuksienhallinnan kannalta yrityksen olisi kyettävä keräämään tietoa maksajasta ja itse käyttäjästä.
- Toimeksiantaja (customer) – Toimeksiantaja määritellään yleensä joko henkilöksi tai organisaatioksi joka omistaa tuotteen tai joka käyttää tuotetta. Toimeksiantaja voi myös samanaikaisesti edustaa jotain toista asiakastyypistä eli olla esimerkiksi tuotteen omistava työntekijä tai tuotetta käyttävä edunsaaja.
- Kilpailija (competitor) – Kilpailija voi olla myös tietyissä tilanteissa yrityksen asiakas. Kyseinen tilanne muodostuu muun muassa silloin, kun yritys hyödyntää kilpailijan infrastruktuuria toimittaessaan tuotettaan asiakkaalle. Esimerkiksi puhelinoperaattoreilla on tapana vuokrata verkkokapasiteettia kilpailijoiltaan. Kilpailija nähdään yrityksen asiakkaana myös silloin, kun yritys myy kilpailijan tuotteita omien tuotteidensa rinnalla. Tällaisessa tilanteessa yritys pyrkii säilyttämään ja kehittämään olemassa olevia asiakkuuksiaan. On nimittäin olemassa riski, että jos asiakas joutuu hankkimaan jonkun osakokonaisuuden yrityksen kilpailijalta, tilaa hän myös muut osakokonaisuudet samalta toimittajalta.

- Työntekijä (employee) – Työntekijä muistuttaa asiakastyypinä välittäjää. Yritykset luokittelevat työntekijänsä asiakkaakseen muun muassa siksi, että usein työntekijöillä on mahdollisuus ostaa yrityksen tuotteita työntekijäalennuksella. Yrityksen kannalta on hyvä tietää, miten kannattavaa tällainen toiminta on ja minkä tyyppiset henkilöt näitä etuja yleensä hyödyntävät. Työntekijöiltä palautetta keräämällä saadaan sellaista tietoa, jota voidaan hyödyntää kehitys-, myynti- ja markkinointitoiminnoissa. Toisaalta työntekijät ovat asiakkaita, jotka välittävät yrityksen tarjoamia tuotteita loppukäyttäjille. Tällöin yrityksen kyky kehittää työntekijöiden ja asiakkaiden välistä vuorovaikutusta on avainasemassa asiakkuuksien ja asiakastyytyväisyyden luomisessa.
- Takaaja (guarantor) – Takaaja on yksilö tai organisaatio, joka takaa luoton takaisinmaksun. Luotonantaja kerää tietoja takaajasta myöntäessään luottoa ja näiden tietojen avulla voidaan analysoida, että onko takaaja tulevaisuudessa potentiaalinen asiakas yritykselle. Takaajasta kerättävät tiedot ovat tietysti tarpeen myös yrityksen riskienhallinnan kannalta. Takaajan taustatiedot selvittämällä pyritään varmistamaan takaajan maksukyky tilanteessa, jossa luotonottaja ei kykene selviytymään velvoitteistaan.
- Talouskunta (household) – Talouskunta on ryhmä yksilöitä tai organisaatioita, jotka yrityksen näkökulmasta muodostavat asiakkaaksi luokiteltavan kokonaisuuden. Talouskunnassa päätökset tehdään ryhmänä, joten yrityksen kannalta tärkeämpää on tunnistaa jäsenten muodostama kokonaisuus kuin yksittäiset jäsenet. (Lehtonen & Puhto 2002, 14-15.)

2.2.2 Asiakkuuksien elinkaari

Asiakkuuden vaihe elinkaarellaan vaikuttaa huomattavasti asiakkuuden syvyyteen ja asteeseen. Uudet asiakkuudet ovat huomattavasti herkempiä katkeamaan mitä vuosia kestäneet. Vanhat asiakkuudet ovat puolestaan yritykselle juuri henkisesti kaikkein tärkeimpiä. Asiakkuuden hankintavaiheessa asiakkuus on aina katkolla, koska asiakas joko tulee yrityksen asiakkaaksi tai ei tule. Asiakkuuden avauduttua on asiakas aktivoitava ostamaan enemmän yrityksen tuotteita ja palveluita. Asiakkuuden tullessa kypsään

vaiheeseen elinkaarellaan on kannattavaa keskittyä asiakkuuden säilyttämiseen liittyviin tekijöihin. (Mäntyneva 2000, 15-16.)

Asiakkuuksienhallinnan kannalta asiakkuuden elinkaari voidaan jakaa neljään vaiheeseen: hankinta, haltuunotto, kehittäminen ja säilyttäminen. Jokaisella vaiheella on omat erityispiirteensä, jotka on hyvä pitää mielessä, kun suunnitellaan asiakkuuksienhallintaan liittyvää tavoitteenasettelua ja käytännön toteutusta. Taulukosta 1. nähdään asiakkuuksienhallintaa kuvattuna elinkaaren eri vaiheista ja eri näkökulmista. (Mäntyneva 2000, 16, 18.)

Taulukko 1: Asiakkuuksienhallinta kuvattuna elinkaaren eri vaiheista ja eri näkökulmista. (Mäntyneva 2000, 19.)

Asiakkuuden vaihe	Hankinta	Haltuunotto	Kehittäminen	Säilyttäminen
Tavoite	Potentiaalisten asiakkuuksien hankinta	Uusien asiakkuuksien kannattavuus muiden tuotteiden lisämyynnillä	Asiakkuuden syventäminen	Olemassa-olevien asiakkuuksien säilyttäminen
Lähestymisen peruste	Demografinen profiili	Aiemmat ostokset	Potentiaalin realisointi	Asiakkaan ostohistoria ja profiili
Tarvelähtöisyys	Oletetut tarpeet	Viestityt tarpeet	Todelliset tarpeet	Todelliset tarpeet
Markkinointiviestinnän personointi	Vähemmänkin persoonallinen riittää	Pyrkimys persoonallisuuteen	Persoonallisuus välttämätöntä	Persoonallisuus välttämätöntä
Tarjouksen sisältö	Perustuu tuotelähtöisyyteen	Perustuu asiakkaan olemassa-oleviin tarpeisiin	Perustuu asiakkaan olemassa-oleviin tarpeisiin ja toiveisiin	Perustuu aiempaan asiakassuhteeseen
Kaupanteon onnistumistodennäköisyys	Alhainen	Kohtalainen	Hyvä	Suhteellisen korkea

Hankinta

Uusia asiakkaita hankittaessa tavoitteena on tehdä ensimmäinen kauppa mahdollisimman kannattavasti ja luoda samalla asiakkuus. Asiakkuuksienhallinnan haasteeksi puolestaan muodostuu tämän uuden asiakkuuden jalostaminen. (Mäntyneva 2000, 19-20.)

Haltuunotto

On hyvin selvää, että asiakkaiden prioriteetit vaihtelevat tuotteittain. Näin ollen ei voida yksiselitteisesti päätellä, miten asiakkuus pitäisi ottaa haltuun ainoastaan yhden tuotteen myyntiä lisäämällä. Vaan pikemmin on kyse koko tuotetarjonnan sovittamisesta asiakkaan tarpeisiin. Harvemmin vakioratkaisut täyttävät minkään asiakassegmentin vaatimuksia ja odotuksia riittävän hyvin. Näin ollen tuote- ja palvelutarjontaa kannattaa mukauttaa vastaamaan paremmin olemassaoleviin tarpeisiin. (Mäntyneva 2000, 20.)

Kehittäminen

Asiakkuuksien kehittämiseen liittyen on mahdollista tehdä ryhmäkohtaisia suunnitelmia ja toimintamalleja, joiden tarkoituksena on asiakkuuksien syventäminen. Asiakkuuksien syventämisen yhteydessä pyritään puolestaan lisäämään yrityksen osuutta asiakkaan kokonaisostoksista. Asiakkuuksienhallinnan keinoin muodostetut segmentit muodostavat riittävän massan, jotta toimintamallien luominen ja käyttöönotto on perusteltua. (Mäntyneva 2000, 21.)

Säilyttäminen

Asiakkuuksien säilyttämiseen liittyvät toimet perustuvat asiakkaiden syvälliseen ymmärtämiseen. Tällä siis tarkoitetaan sitä, että on tunnettava asiakkaat ja heidän todelliset tarpeensa. Tämän jälkeen mallintamisella pyritään selvittämään ne asiakkaat, jotka todennäköisesti ovat vaihtamassa toimittavaa yritystä ja näin ollen lopettavat asiakkuutensa yrityksen kanssa. Kun tämä todennäköisyysmallinnus yhdistetään asiakkaan taloudellisen

kannattavuuden ja potentiaalinen analysointiin, voidaan paljon perustellummin tehdä asiakkuuksien säilyttämiseen tähtäviä päätöksiä. Esimerkiksi voidaan päätellä kuinka paljon kannattaa panostaa siihen, että asiakkuus säilyisi eikä asiakas vaihtaisi toimittajaa. (Mäntyneva 2000, 22.)

2.2.3 Asiakkaan arvontuotanto

Asiakkuuden kehittäminen edellyttää, että asiakkaan itselleen tuottaman arvoon vaikuttava prosessi tunnetaan syvällisesti. Prosessitarkastelussa yksittäiseen ostotapahtumaan vaikuttava merkitys pienenee. Asiakkuuksien johtamisen tavoitteena ei ole yksittäiseen kauppaan liittyvän tuoton maksimointi vaan yhteistyössä asiakkaan kanssa rakentaa kestävää asiakkuutta. Asiakkuusajattelun ideana on toimia asiakkuuden ehdoilla ja pyrkiä sovittamaan prosesseja niin hyvin toisiinsa, että arvoa syntyy molemmille osapuolille. Asiakkuusajattelussa kilpailukyky ei perustu ainoastaan hintakilpailukykyyn vaan myös yrityksen kykyyn auttaa asiakasta tuottamaan arvoa itselleen. Asiakkuuden onnistumisen keskeisimpiä edellytyksiä ovat aito vuorovaikutus ja pyrkimys sellaiseen asiakkuuteen, jossa molemmat osapuolet saavat etua panostaessaan asiakkuuteen. Yritys vahvistaa omaa kilpailukykyään, jos se kykenee jatkuvasti arvioimaan ja määrittämään prosessejaan uudestaan. Asiakkaalle tämä tarkoittaa sitä, että yritys pystyy tarjoamaan uusia mahdollisuuksia sen arvon tuottamiseen. (Lehtonen & Puhto 2002, 16.)

Arvo voidaan jakaa kahdeksi tekijäksi; ydinarvoksi ja lisäarvoksi. Ydinarvolla tarkoitetaan ydinratkaisun hyötyjä verrattuna siitä maksettavaan hintaan. Lisäarvo taas tuottavat suhteen lisäpalvelut verrattuina ajan mittaan toteutuviin suhdekustannuksiin. Lisäarvo voi olla joko positiivista tai negatiivista. Positiiviseksi lisäarvoksi voidaan mainita muun muassa nopeat toimitukset ja hyvä asiakaspalvelu, jolloin vaikutus asiakkaan kokonaisarvoon on myönteinen. Negatiivinen lisäarvo aiheuttaa asiakkaalle tarpeettomia tai odottamattomia suhdekustannuksia. Negatiivista lisäarvoa aiheuttavat muun muassa turhan monimutkaiset järjestelmät, vaikeaselkoinen tekniikka, epäystävälliset tai ammattitaidottomat työntekijät ja myöhästyneet toimitukset. Tällöin ei enää ole kysymys lisäarvosta vaan arvonvähennyksestä. (Lehtonen & Puhto 2002, 16; Saarikkomäki 2006, 3.)

Asiakkuuslähtöisen yrityksen pyrkimyksenä on lisätä kaikilla keinoilla tietouttaan siitä, miten asiakkaat tuottavat itselleen arvoa. Tämän tiedon pohjalta on helpompaa arvioida, miten yritys olemassa olevalla osaamisellaan voi auttaa asiakasta. Arvontuotantoa ymmärtämällä voidaan asiakkuutta kehittää sekä asiakkaan että yrityksen lähtökohdista. Yrityksen on tunnistettava asiakkaidensa hankintatavat, hankintastrategiat sekä hankinnan taustalla vaikuttavat vaikuttimet, ennen kuin kyetään tuottamaan arvoa asiakkaalle. Arvon tuottaminen edellyttää toimittajan ja asiakkaan prosessien sopimista yhteen. On kuitenkin hyvä muistaa että asiakkaat kokevat arvon eri tavalla ostohetkellä kuin tarjoaman käytön aikana. (Lehtonen & Puhto 2002, 16; Saarikkomäki 2006, 3.)

2.3 Asiakkuuksienhallinnan osa-alueet

Asiakkuuksienhallinta voidaan jakaa kahtia sen perusteella minkälaisia toimintoja asiakkuuksienhallinta tukee. Operatiivinen asiakkuuksienhallinta on asiakasrajapinnassa tapahtuvaa toimintaa. Analyytinen asiakkuuksienhallinta on puolestaan päätöksentekoa tukevaa toimintaa.

2.3.1 Operatiivinen asiakkuuksienhallinta

Operatiivinen asiakkuuksienhallinta kattaa perusteet vuorovaikutus prosessiin asiakkaiden kanssa, prosessin hallinnan sekä hajautettujen toimintojen hallinnoinnin eri kanavissa ja partnerisuhteissa. Operatiiviset järjestelmät ja asiakaskannat toimivat hyvin asiakastiedon keräämisessä ja varastoinnissa, mutta näillä järjestelmillä ei pystytä suoraan tuottamaan kovin merkittävää tietoa pääteksentekoa varten. Operatiivinen asiakkuuksienhallinta pitää sisällään kaikki päivittäisessä asiakastoiminnassa tarvittavat toiminnot. Sen avulla voidaan hoitaa esimerkiksi tilausten käsittelyt ja laskutukset. Operatiivinen asiakkuuksienhallinta toimii asiakasrajapinnassa ja sitä käyttävät hyödyksi muun muassa markkinointi, myynti, asiakaspalvelu ja huolto. (Järvinen 2002, 7-8; TIEKE ry 2007 c.)

2.3.2 Analyttinen asiakkuuksienhallinta

Analyttinen asiakkuuksienhallinta pyrkii tuottamaan merkityksellistä tietoa asiakassuhteista päätöksentekoa varten. Se siis yhdistää asiakasdatan eri kanavista yhteen järjestelmään tarjoten päätöksenteolle alustan. Käytännössä tämä tarkoittaa sitä, että operatiivisessa asiakkuuksienhallinnassa tuotettua dataa käytetään analyttisessä asiakkuuksienhallinnassa hyödyksi. Keskeisenä elementtinä analyttisessä asiakkuuksienhallinnassa on erilaiset tiedonlouhinta menetelmät. Tiedonlouhintaa käsitellään tarkemmin myöhemmässä vaiheessa tässä opinnäytetyössä. (Järvinen 2002, 8.)

Analyttiset asiakkuuksienhallintajärjestelmät tarjoavat niin sanotun 360 –asteen näköalan asiakassuhteisiin. Se auttaa muun muassa ymmärtämään mitä asiakkaat sanovat, keitä he ovat, mitkä ovat heidän tarpeensa ja tärkeimpänä, mitä he tekevät tulevaisuudessa. Tällä järjestelmällä pystytään tarjoamaan tutkittua tietoa päätöksentekoon ”mutu-tuntuman” sijaan. (Järvinen 2002, 8.)

2.4 Tietojärjestelmien suunnittelun lähtökohdat ja vaatimukset

Asiakkuuksienhallintaan tarkoitettujen tietojärjestelmien vaatimukset riippuvat hyvin paljon siitä, kuinka yksityiskohtaista tietoa kerätään ja onko tietoja tarkoitus analysoida tarkemmin. Asiakkuuksienhallinta voi olla vain yksinkertaisesti asiakastietojen keräämistä ja asiakaskontaktien hallintaa. Tällaisessa tilanteessa tietojärjestelmän suunnittelu ei välttämättä ole kovinkaan suuri haaste. Vaikeutta asiaan tosin tulee heti lisää, jos yhdistettävänä on useammista tietojärjestelmistä tulevaa tietoa ja tiedon analysoimista.

2.4.1 Käsitteet, käsittemallit ja tietoyhtenäisyys

Käsite on reaali maailmaan kuuluva asia tai ilmiö, joka voidaan yksilöidä. Esimerkiksi henkilö, projekti, suunnitelma tai vastaava asia. Käsitteet ovatkin ihmisten välisen kommunikoinnin perusta. Tietoteknisessä järjestelmässä käsitteet taas puolestaan ovat sen suurimpia haasteita että niiden ratkaisuja. (Ala-Mutka & Talvela 2004, 138.)

Käsitteiden merkitykset ja sisällöt ovat yleensä organisaatiossa opittuja asioita. Usein käytettävillä käsitteillä on erilaisia merkityksiä ja sisältöjä eri yksiköissä ja eri maissa. Ongelmia saadaan, kun tähän käsiteviidakkoon laitetaan mukaan vielä koko yrityksen yhteinen tietojärjestelmä, joka ymmärtää käsitteet vain yhdellä tavalla. Verkottuneessa toimintatavassa käsitteet eivät olekaan ainoastaan vain organisaation itsensä määrittelemiä, vaan käsitteissä tulee ottaa vielä huomioon yleiset standardit. Kokonaisuuden tekee haastavaksi se, että kaikkia tietoteknisiä ongelmia ei saada ratkaistua yhdellä tietojärjestelmällä vaan on hankittava useampia, ja usein näiden tietojärjestelmien käsitteet poikkeavat toisistaan. Käsite ”projekti” muuttuukin toisessa järjestelmässä ”hankkeeksi”. Ongelmia syntyy myös silloin, kun näitä tietoja pitää siirtää tietojärjestelmästä toiseen. (Ala-Mutka & Talvela 2004, 138-139.)

Tällaista käsiteviidakkoa voidaan välttää, jos eri yksiköt tekevät riittävästi yhteistyötä. Tietenkin käsitteiden ja käsittemallien kokonaisvaltainen hallitseminen estää sen, että jokainen yksikkö ei saa määritellä itse käsitteitään. Se ei kuitenkaan vielä pelkästään riitä, että käsitteet ovat selkeitä ja kaikkialla samalla tavalla ymmärrettyjä. Käsitteet on lopulta muodostettava loogisiksi kokonaisuuksiksi eli käsitemalleiksi. (Ala-Mutka & Talvela 2004, 139-140.)

Tietoyhtenäisyys eli organisaation käsitemalli (~tietomalli) helpottaa tiedon siirtoa, hyödyntämistä ja ymmärtämistä eri tilanteissa. Erityisesti tiedon käytettävyys ja seurattavuus on mahdollista toteuttaa yhtenäisellä käsitteistöllä. Tiedon käytettävyys tarkoittaa siis saman datan hyödyntämistä eri tarkoituksissa, eli esimerkiksi toimitusvarmuutta tehtaittain, asiakkaittain ja alueittain. Tietoa siis näin ollen voidaan käyttää hyväksi eri yksiköissä omiin tarpeisiin. Tiedon seurattavuus tarkoittaa puolestaan sitä, että toimintaketjun alkupäässä luotu ”käsite” ja sen rakenne voidaan tunnistaa ja niitä voidaan seurata toimintaketjun alkupäähän ja päinvastoin. Tiedon seurattavuutta tarvitaan esimerkiksi silloin, jos haluamme tietää jonkin elektronisen komponentin (akku) jostain matkapuhelimesta (tuotantoerä). (Ala-Mutka & Talvela 2004, 141-142.)

2.4.2 Tietotekniset lähtökohdat tietojärjestelmien kehittämiseen

Asiakkuuksienhallintajärjestelmää voidaan lähteä toteuttamaan useammalla eri tavalla. On kuitenkin huomattava, että osa näistä lähestymistavoista sopii paremmin tietynlaisille yrityksille. Asiakkuuksienhallintajärjestelmien tekninen perusta voidaan siis jakaa viiteen erilaiseen lähestymistapaan ja niiden yhdistelmiin. Lähestymistavat ovat seuraavanlaiset:

1. pakettisovellus
2. parametroitava sovellus
3. konfiguroitava sovellus
4. mukautettava sovellus
5. ohjelmoitu sovellus. (Ala-Mutka & Talvela 2004, 143.)

Pakettisovellus sisältää valmiit prosessit ja tietosisällön, jolloin sitä ei mukauteta toiminnallisesti. Käsitteiden ja tietorakenteiden sovittaminen yhteen muiden tietojärjestelmien kanssa on tällaisessa sovelluksessa vaikeaa. Parametroitavassa sovelluksessa voidaan puolestaan muuttaa tiedon sisältöä (kenttiä) ja konfiguroitavassa sovelluksessa voidaan kenttien lisäksi muuttaa myös prosesseja. Mukautettava sovellus on näistä kaikkein joustavin vaihtoehto. Sillä kyetään tekemään parhaiten asiakaskohtaisia sovelluksia sen sisään rakennetun sovelluskehittimen avulla. (Ala-Mutka & Talvela 2004, 144.)

Sovelluskehitin on tässä tapauksessa sellainen ohjelma, jolla voidaan tehdä uusia käsitelmalleja ja prosesseja ilman teknistä räätälöintiä ohjelmaan. Ohjelman mukauttaminen tehdään muuttamalla symboleja ja käsitteitä, loogisia yhteyksiä ja operaatioita ilman, että tehdään varsinaista ohjelmointia. (Ala-Mutka & Talvela 2004, 144.)

Ohjelmoitu sovellus tarkoittaa ohjelmiston arkkitehtuurin suunnittelua ja ohjelmoimista asiakaskohtaisesti aivan alusta loppuun saakka. Yleensä ohjelmoidut sovellukset vastaavat parhaiten tarpeisiin, mutta ovat taas vastaavasti kalliimpia sekä hankintahinnaltaan että omistuskustannuksiltaan korkeiden ylläpitomaksujen ja muutostalustusten takia. Toisaalta tilaustyönä ohjelmoitu sovellus yleensä sopii parhaiten organisaation toimintaan, jolloin säästöä syntyy tehokkaamman prosessin muodossa. (Ala-Mutka & Talvela 2004, 144-145.)

Yhteenvedona voidaan sanoa, että eri lähestymistavat sopivat erilaisiin organisaatioihin, liiketoimintaympäristöihin ja asiakaskantoihin. Järjestelmän valintaan vaikuttavat lähinnä yrityksen koko, liiketoiminnan monimuotoisuus ja strategian ainutlaatuisuus. Taulukkoon 2. on koottu tietoteknisten lähestymistapojen kuvaukset ja soveltuvuus käyttökohteeseen.

Taulukko 2: Tietotekniset lähestymistavat asiakkuudenhallinnanjärjestelmiin ja niiden käyttökohteet. (Ala-Mutka & Talvela 2004, 147.)

Lähestymistapa	Kuvaus	Soveltuvuus
1. Pakettisovellus	Valmiit prosessit, sisältö (kentät) ja näytöt, joita on vaikea muuttaa.	Pienet yritykset (hinta merkitsevä), homogeeninen asiakaskanta, kontaktien operatiivinen hallinta, parhaiden käytäntöjen mukaiset prosessit ja vähän palvelukanavia.
2. Parametroitava sovellus	Valmiit prosessit, joiden sisältöä voidaan muuttaa.	Suuret ja pienet yritykset, homogeeninen asiakaskanta, parhaiden käytäntöjen mukaiset prosessit, standardit toimintamallit ja vain vähän erilaisia palvelukanavia.
3. Konfiguroitava sovellus	Prosesseja voidaan muuttaa ja samoin kenttien sisältöä.	Suuret yritykset, homogeeninen tai joitakin eroja toimintamalleissa. Melko laaja palvelukanavastrategia.
4. Mukautettava sovellus	Prosesseja ja tietosisältöä voidaan muuttaa paljon.	Monimuotoinen liiketoiminta. Laaja kirjo tuoteperheitä ja palveluja. Erilaisia asiakasryhmiä (pienet ja isot organisaatiot sekä kuluttaja-asiakkaat). Laaja palvelukanavastrategia.
5. Ohjelmoitu sovellus	Prosessit ja sisältö sekä näytöt tehdään asiakaskohteisesti.	Erikoiskohteet, jossa toiminnan erityispiirteet vaativat tarpeeseen tehtyjä näyttöjä (esim. karttapohjia), toimintoja ja prosesseja.

2.4.3 Tietoteknologian mukautuminen

Asiakkuuksienhallintaa tukevan teknologian vaatimuksena on mukautua erilaisiin tilanteisiin. Tietojärjestelmiä joudutaan muun muassa mukauttamaan organisaation käsitteisiin ja käsitelmalleihin, organisaation toimintatapoihin, erilaisiin käyttäjäryhmiin ja erilaisiin käyttötilanteisiin sopivaksi. Tärkeää on myös pystyä luomaan erilaisia raportteja eri käyttäjille ja tekemään analyyskejä kerätystä datasta. (Ala-Mutka & Talvela 2004, 148.)

Raportointi ja datan analyysit

Asiakkuuksienhallinnan yksi tärkeä vaatimus on tarpeisiin mukautuva raportointi ja datan analyysit. Yleensä jokaisessa soveluksessa on oma raporttigeneraattori eli ohjelma, joka tuottaa erilaisia raportteja sovelluksesta. Usein nämä raportit on tehty vain yhden toiminnon näkökulmasta, ja siten ne palvelevat hyvin kapeita tarpeita. Asiakkuuksienhallinnassa on puolestaan tarve saada tietoa koko arvoketjusta. Yrityksillä on tarve yhdistää tietoa eri lähteistä ja luoda niistä uusia raportteja ja datan analyyskejä. (Ala-Mutka & Talvela 2004, 150.)

Asiakkuuksienhallinnassa tähän ratkaisuna yleensä on tietovaraston rakentaminen. Tietovarastoon puolestaan voidaan tarvittaessa liittää erilaisia yleisiä raportointityökaluja ja datan analyysityökaluja. Tietovarastoon liitetyillä ohjelmilla voidaan myös tuottaa tavalliset toimintopohjaiset raportit. Tosin pitää muistaa, että organisaation yhteiset käsitteet ja käsitelmallit ovat edellytyksenä tietoyhtenäisyydelle ja tiedon yhteiskäyttöisyydelle. (Ala-Mutka & Talvela 2004, 150.)

Tietojärjestelmien mukauttaminen

Liiketoiminnan muita vaatimuksia ovat tietojen integrointi toimintopohjaisista sovelluksista, esimerkiksi tuotannon, logistiikan ja myynnin käytössä olevista sovelluksista. Integroitavuus tarkoittaa tässä tapauksessa lähinnä avoimien standardien ja rajapintojen käyttöä sekä tietovarastoja. Yleisenä vaatimuksena on tietenkin joustavuus eri käyttötilanteisiin ja käyttäjien tarpeisiin. Tietojärjestelmien mukautuminen liiketoiminnan tarpeisiin on mahdollista toteuttaa teknologioilla, jotka perustuvat käsitteisiin,

käsitelmalleihin ja yleensä siihen, että asioita ei ole ”kovakoodattu” ohjelmaan. Tämä siis tarkoittaa sitä, että tieto ei ole vain ohjelmoitu sovelluksen sisään vaan käsitteitä voidaan tarvittaessa muuttaa. (Ala-Mutka & Talvela 2004, 151-152.)

2.5 Asiakkuustietoon liittyvät tekijät

Asiakkuuksiin liittyvää tietoa syntyy yrityksissä koko ajan lisää, mutta tuskin enää yhdessäkään yrityksessä tiedonhallintaan käytetään manuaalisia menetelmiä, vaan on siirrytty jo tietokoneavusteiseen tiedonhallintaan. Tietoteknologiaa hyväksi käyttämällä, tästä asiakkuuksiin liittyvästä tiedosta voidaan tiedonlouhinta menetelmillä saada poimittua sellaista tietoa, joka auttaa huomattavasti muun muassa yrityksen päätöksenteossa.

2.5.1 Tiedon määrä ja sen lisääminen

Tietokoneavusteiseen tiedonhallintaan siirtyminen tulee luonnolliseksi silloin, kun manuaaliset menetelmät muuttuvat liian työläiksi. Tietokantapohjaiseen työskentelytapaan siirtymiseen olisi myös löydyttävä niin paljon hyötyä, että sen toteuttaminen kannustaa viemään läpi joskus suhteellisen työläänkin siirtymisprojektin. Toisaalta jossain yrityksissä on jo voitu alusta lähtien käyttää tietokantapohjaisia järjestelmiä perustiedonhallintaan. Tiedon määrän lisäämisellä näissä tilanteissa voitaisiin ehkä saavuttaa lisäarvoa myös yritykselle ja asiakkaille kun tiedetään enemmän asiakkaista ja näiden toimitavoista. (Mäntyneva 2000, 76.)

Asiakkuuksiin kohdistuvan informaation tallentaminen ei ole ajatuksena uusi asia. Oletettavasti niin kauan kun on tehty kauppaa on myös kauppoihin liittyvistä transaktioista ja kaupan osapuolista tehty muistiinpanoja. Tietokantojen käyttö mahdollistaa näiden muistiinpanojen hyödyntämisen entistä syvällisemmin ja järjestelmällisemmin. Asiakkuuksienhallinnan näkökulmasta olisi toivottavaa, että kaikki asiakkuuksiin liittyvät muistiinpanot olisivat sopivalla tavalla määrämuotoisia. Informaation määrämuotoon saattaminen lisää tiedon hallittavuutta, vertailtavuutta ja yleensäkin sen käytettävyyttä. Tätä asiaa käsiteltiin tarkemmin jo kappaleessa käsite, käsitemallit ja tietoyhtenäisyys.

Tietokantojen käyttö mahdollistaa paremmin myös asiakkaiden tarpeiden tunnistamisen niin ryhmä- kuin myös yksilötasolla. (Mäntyneva 2000, 76.)

Asiakkaiden ja heidän ostokäyttäytymisen ja tarpeiden parempi tunteminen antaa mahdollisuuden entistä osuvampaan tuotekehitykseen. Tässä yhteydessä informaatio, joka sisältyy toistuviin ostoihin, asiakasdialogiin, reklamaatioihin ja tuotteille myönnettyjen takuiden käsittelyyn, antaa kuvaa siitä, miten asiakkaat todellisuudessa käyttävät tuotteita ja mitä he mahdollisesti haluaisivat niissä parannettavan. Tuotteen parempi vastaavuus asiakkaan tarpeisiin antaa arvoa asiakkaalle ja tietysti yritykselle, koska asiakas pysyy yrityksen asiakkaana. (Mäntyneva 2000, 77.)

Asiakkuuksien tietoja luultavasti käytetään kuitenkin enemmän hyödyksi markkinointitarkoituksissa kuin tuotekehityksessä. Toki on muistettava, että tämä asiahan varmasti riippuu paljolti siitä millä toimialalla toimitaan. Esimerkiksi ohjelmistoalalla voidaan hyvinkin asiakkuuksien tietoja käyttää hyödyksi enemmän tuotekehityksessä kuin markkinoinnissa tai tiedon käytettävyys voi olla kutakuinkin samansuuruista molemmissa asioissa.

2.5.2 Tiedon sisältö ja rakenne

Yrityksen on aluksi selkeytettävä itselleen, mitä tietoja asiakaskannan johtamiseen ja asiakkuuksien hoitamiseen tarvitaan. Millaisia markkinointikampanjoita mahdollisesti halutaan toteuttaa, ja millaista informaatiota on käytettävissä? Tässä yhteydessä on otettava kantaa asiakkuuksienhallintaan kahdesta näkökulmasta: koko asiakaskannan johtamisen ja asiakkuuksien hoitamisen kannalta. Näihin asioihin liittyvät tavoitteet asettavat omat vaatimuksensa tiedon sisällölle ja rakenteelle. (Mäntyneva 2000, 78-79.)

Yleensä ensimmäiseksi kannattaa selvittää, mitä tietoa yrityksellä on jo ennestään käytettävissä sekä mihin ja miten tietoa nykyisin käytetään. Nykytilan selvittämisen jälkeen luodaan näkemys tavoitetilasta eli selvitetään mihin tietoja halutaan käyttää ja mitä tietoja tässä yhteydessä tarvitaan. Nykyisin käytössä olevan asiakastiedon mahdollisuudet ja rajoitteet pitää selvittää. Mikä puolestaan tarkoittaa seuraavien tekijöiden selvittämistä. ”Kuinka paljon asiakkaita on tietokannassa, ja ovatko kaikki asiakkuudet

mukana? Mikä osa tietokannan asiakkaista ei ole aktiivisia asiakkaita? Kuinka virheetön asiakastietokanta on?”. (Mäntyneva 2000, 79.)

Asiakkuuksienhallinnan tarkoituksiin saatetaan kerätä esimerkiksi seuraavanlaisia tietoja: demografiset, psykografiset ja muut asiakkuustekijät, asiakkaan ostokäyttäytymiseen ja ostohistoriaan liittyvät tiedot, asiakkaalle kohdistettu markkinointiviestintä sekä muu asiakkuusinformaatio. Näihin tietoihin voidaan toki yhdistellä myös muita yrityksen ulkoisia tai sisäisiä tietoja pyrkimyksenä asiakkuuksienhallinnan tavoitteiden täyttyminen. Koska tiedoilla on myös paha taipumus vanhentua, tulisi selkeyttää myös seuraavat tekijät. ”Kuinka uusi tai päivitetty tieto viedään tietokantaan? Kenellä on tietokannan päivitysoikeus ja –velvollisuus? Mitkä sisäiset järjestelmät vievät automaattisesti tietoa asiakastietokantaan?” Tärkeää olisi myös täsmentää kuka valvoo prosessia ja sen toimivuutta. (Mäntyneva 2000, 79-80.)

Tietokannan rakennetta ja tietosisältöä ohjaa siis ensisijaisesti tietokannan tarkoitus eli mihin tietoja halutaan käyttää. Sikäli kun tarpeet asiakkuuksienhallinnan osalta liittyvät asiakaskannan johtamiseen ja asiakkuuksien hoitoon, tietotarpeet ovat varsin moninaiset. Käytettävissä olevalla tiedolla tulisi saada selvitettyä asiakkuuksien kannattavuus ja luonne. Tietokannan koon osalta on puolestaan täsmennettävä, että kuinka paljon asiakkaita ja asiakkaista kerättävää tietoa viedään tietokantaan. Potentiaalisten asiakkuuksien liittäminen tietokantaan saattaa kasvattaa tietokannan kokoa merkittävästikin. Tietokannan käyttöaktiiviteetista tulisi myös tehdä päätös, että kuinka usein tietokantaa päivitetäisiin jotta sitä olisi mahdollista käyttää myös markkinoinnin tukena. (Mäntyneva 2000, 80.)

2.5.3 Tietovarastot (Data warehouse)

Tietovarastoja tarvitaan vaatimaan tietojen analysointiin, mallintamiseen ja raportointiin. Operatiivisten tietojärjestelmien soveltuvuus tähän asiaan ei riitä, jolloin on syytä perustaa tietovarastoja. Käytännössä tietovaraston tiedot kuitenkin poimitaan operatiivisten järjestelmien käyttämistä tietokannoista. Keskeisenä tarkoituksena tällä kaikella on se, että voidaan nopeasti tehdä vaativia raportteja ja analyyseja. Yrityksissä syntyy koko ajan merkittävä määrä tietoa, ja kaiken tämän tiedon käyttökelpoiseksi saattaminen vaatii tiedon haltuunottamista. Tietovarastoissa tieto kerätään yhteen paikkaan, jossa yksittäisiä

tietoalkioita on helppo yhdistellä ja analysoida. Tietovarastot antavat mahdollisuuden erilaisten kysymysten tekoon, ja useimpiin niistä saadaan vielä vastaukset. Myös jatkuva asiakkaista kerätyn tiedon käsittely vähentää perinteisten markkinointitutkimusmenetelmien kuten asiakastutkimusten tarvetta (Mäntyneva 2000, 82; Koskinen 2005, 55.)

Asiakkuuksienhallintajärjestelmä perustuu asiakkaista kerätyn tiedon hyödyntämiseen. Tästä johtuen ei ole liioiteltua korostaa tiedon varastoinnin tärkeyttä. Tietovarastoratkaisun tuomia etuja organisaatiolle ovat muun muassa seuraavat asiat:

- Tarkempi ja nopeampi pääsy informaatioon. Tällä helpotetaan vastaamista asiakkaiden esittämiin kysymyksiin.
- Tiedon laatu ja suodattaminen huonon ja kaksinkertaisen tiedon varalta.
- Tiedon nopea erottelu, käsittely ja syväluotaus kannattavuusanalyysijä, asiakasprofilointeja ja asiakaspysyvyyden mallintamista varten.
- Kehittynyt tiedonyhdistely ja datan analysointityökalut korkeamman tason yhteenvetoon ja yksityiskohtaisiin raportteihin.
- Asiakkaan nykyisen kokonaisarvon laskeminen ja tulevan arvon arviointi. (Koskinen 2005, 55.)

2.5.4 Paikallisvarasto (Datamart)

Tietovarastot eivät kuitenkaan ole ainoa vaihtoehto tiedon varastoimiseen ja analysointiin. Tiedon paikallisvarastoksi kutsutaan kokonaisuutta, jossa yhden perusjärjestelmän tiedot siirretään omaan tietokantaansa analyysijä ja raportteja varten. Tietovarastoinnin yhteydessä on hyvä ottaa kantaa siihen, että olisiko parempi käyttää tietovarastoa vai paikallisvarastoa. Pienestäkin voi lähteä liikkeelle, mutta tällöin on tärkeätä pitää silmällä mahdollisia jatkokehitystarpeita. Näin vältetään turhien erillisten järjestelmien tekemiseltä, jotka eivät ole yhteydessä toisiinsa. Jos ajatellaan, että tietovarastot on koko yrityksen laajuisia, niin paikallisvarastot kattavat usein vain osastotason tiedot. Periaatteessa tämä usein riittääkin,

sillä osastot eivät aina tarvitse toisten osastojen tietoja. Asiakkuuksienhallinnan kannalta keskeisiä tietoja ovat asiakkuuksiin, tuotteisiin ja asiakaskannattavuuteen liittyvät tiedot. Näistä tiedoista on myös mahdollista tehdä paikallisvarastot, jotka viedään yhteen suureen tietovarastoon tietojen yhdistelemistä ja analysointia varten. (Mäntyneva 2000, 82-83.)

2.5.5 Tiedonlouhinta (Data mining)

Tiedonlouhinta on prosessi, jossa suurista tietomääristä etsitään ennalta tuntemattomia syy-seuraus –suhteita yrityksen päätöksenteon tueksi. Tiedonlouhinta mahdollistaa hyvinkin vaativan ja syvällisen analyysin tekemisen. Suurin haaste tiedonlouhinta työssä on muuttaa data sellaiseen muotoon, jossa algoritmit voivat käyttää sitä tehokkaasti hyödyksi. Tiedonlouhinta on myös mahdollista ilman tietovarasto ratkaisuja, mutta tietovaraston olemassaolo helpottaa huomattavasti tiedonlouhinta. (Järvinen 2002, 17; Koskinen 2005, 54.)

Tiedonlouhinnan pääkohdealueina voidaan pitää liiketoiminnallista ja taloudellista mallinnusta, asiakkuuksien segmentointia ja profilointia, asiakasarvon analyysiä ja mallinnusta sekä markkinointikampanjoiden suunnittelua. Suurin osa kaikista talouden ja liiketoiminnan vaatimuksista pystytään ratkaisemaan seuraavien kuuden tiedonlouhinta perusmenetelmän avulla. (Mäntyneva 2000, 85; Järvinen 2002, 18)

Luokittelu (Classification)

Luokittelu on yleisin tiedonlouhinta menetelmä. Luokittelussa määritellään ensin tarkasti luokka ja sen ominaisuudet sekä luodaan näiden pohjalta luokalle esimerkkitapaukset. Sopivia tekniikoita luokitteluun ovat päätöksentekopuut (decision tree), jäsentäminen (memory-based reasoning) ja suhdeanalyysit (link analysis). (Järvinen 2002, 18)

Arvioiminen (Estimation)

Luokittelussa vastauksena saadaan vaihtoehtoisia vastauksia tyyliin kyllä/ei ja pieni/kohtalainen/suuri. Arvioimisessa saadaan liukuvia arvoja halutuista muuttujista.

Esimerkiksi tulot, luottoraja ja pituus. Arvioimisen avulla voidaan saada järjestykseen suurikin joukko asiakkaita. Sopiva tekniikka arvioimiseen on neuraaliverkot. (Järvinen 2002, 18)

Ennustaminen (Prediction)

Ennustaminen on periaatteessa sama asia kuin luokittelu tai arvioiminen, mutta siinä toiminnot toteutetaan tulevaan ostokäyttäytymiseen tai arvoon pohjautuen. Ennustamisessa käytetään samoja menetelmiä kuin luokittelussa tai arvioimisessa, mutta ennustamisessa pitää olla käytössä muuttujien historiatiedot, jotta niihin pohjautuvat ennustukset voidaan tehdä. Ennustamisella voidaan hakea vastauksia esimerkiksi kysymyksiin mikä asiakasryhmä ottaa seuraavaksi käyttöön uuden teknologian tai missä asiakasryhmissä on odotettavissa myynnin heikkenemistä seuraavan puolenvuoden aikana. Ennustamiseen käytettävä tekniikka riippuu syötettävästä datasta, muuttujan ominaisuuksista ja halutusta ennusteen selittävydestä. (Järvinen 2002, 19)

Kytkeminen (Affinity grouping)

Kytkemisen tarkoituksena on löytää erillisiä tapahtumia tai tekijöitä, jotka esiintyvät yhdessä. Yksinkertaisin esimerkki tästä on varmaan ostoskorianalyysi, jossa tutkitaan mitä tuotteita ostetaan yhtäaikaan. Kauppiaalle tämä antaa arvokasta tietoa esimerkiksi siitä, että miten tuotteet kannattaisi sijoitella toisiinsa nähden. Kytkemistä voidaan hyödyntää myös ristiinmyynnin tai tuotteiden ja palveluiden ”paketoinnin” suunnittelussa. (Järvinen 2002, 19)

Klusterointi (Clustering)

Klusterointi on puolestaan heterogeenisen ryhmän jakamista homogeenisiin alaryhmiin tai klustereihin. Luokitteluun tässä on erona se, että siinä luokat määriteltiin etukäteen ja käytettiin esimerkkitapauksia luokan ilmentyminä. Klusteroinnissa data jaetaan samanlaisten arvojen tai ominaisuuksien mukaan, joiden löydetään olevan määräävinä

tekijöinä. Taulukossa 3. on esimerkkinä matkatoimiston asiakaskannan klusteroinnista markkinointiviestintää varten. (Järvinen 2002, 20)

Taulukko 3: Esimerkki matkatoimiston asiakaskannan klusteroinnista markkinointiviestintää varten. (Järvinen 2002, 20)

Klusteri	Määräävä tekijä	Viesti
Kaukomatkailijat	Tehdyt kaukomatkat	Elämymatkat
Euroopan matkailijat	Tehdyt aurinkolomat ja euroopanmatkat	Kaukomatkat
Kotimaanmatkailijat	Tehdyt kotimaan matkat ja risteilyt	Aurinkolomat ja Euroopanmatkat

Selittäminen (Description)

Tiedonlouhinnalla on joskus tarkoituksena yksinkertaisesti selittää mitä tietovarastossa olevalle tiedolle tapahtuu. Selittämisellä on siinä mielessä suuri merkitys, että raakadataa ei ole mitenkään mahdollista tutkia tehokkaasti ilman apuvälineitä. Hyvä kuvaus informaation käyttäytymisestä antaa hyvin usein myös selityksen havaituista muutoksista tai kehityssuunnista. Muuttujien ja niiden kehityksen sekä niiden suhteita toisiinsa nähden tutkimalla voidaan löytää piileviä riippuvuuksia, jotka auttavat ymmärtämään havaitut ongelmat. Selittämiseen voidaan käyttää muun muassa jo aiemminkin mainittuja ostoskori- ja neuraaliverkkotekniikoita. (Järvinen 2002, 20)

Tehokkaaseen analysointiin pääsemiseen suurissa tietomäärissä tarvitaan ehdottomasti esimerkkitapausten luomista useimmissa eri menetelmissä. Näillä menetelmillä varmistetaan tulosten oikeellisuutta, validiteettiä ja luotettavuutta. (Järvinen 2002, 20)

3 LUOVANET ASIAKKUUKSIENHALLINTAOHJELMISTO

Luovanet asiakkuuksienhallintaohjelmisto on Internet-selaimella toimiva asiakkuuksienhallintaan tarkoitettu sovellus. Sovelluksen tarkoituksena on helpottaa Luovanet Oy:n ja Luovaliike Oy:n asiakkuustietojen hallitsemista. Sovelluksesta ei ollut olemassa aiempaa versiota, joten projektin toteuttaminen aloitettiin pitämällä palaveri, jossa määriteltiin sovellukseen tulevat ominaisuudet. Ohjelmiston virallisena nimenä ei kuitenkaan ole ”Luovanet asiakkuuksienhallintaohjelmisto”, vaan se on ainakin toistaiseksi toiminut ohjelmiston projektinimenä.

Ohjelmiston toteuttamisen suhteen oli kohtuullisen vapaat kädet kokeilla omia ratkaisuja. Ohjelmistolle käytännössä määriteltiin yrityksen puolelta toiminnalliset vaatimukset ja ohjelmistossa huomioitavat asiat. Näiden tietojen pohjalta aloitettiin suunnittelemaan ohjelmistoa ja kirjoittamaan vaatimusraporttia. Vaatimusraportti piti kuitenkin hyväksyttää, ennen kuin varsinaisen ohjelmiston ohjelmoiminen voitiin aloittaa.

3.1 Taustaa

Luovanet asiakkuuksienhallintaohjelmisto toteutettiin Luovanet Oy:n tiloissa. Projektin toteuttaminen aloitettiin kesäkuussa 2006 ja projekti saatiin valmiiksi syyskuun 2006 alkuun mennessä. Projektin toteuttamiseen ei kuitenkaan käytetty kaikkea tuona välisenä olevaa aikaa, vaan ohjelmiston toteuttamisessa oli pieniä taukoja.

3.2 Ohjelmointikielien ja työkalut

Projekti toteutettiin käyttämällä Visual Studio 2005 ohjelmointiympäristöä. Ohjelmointikielenä oli C# ja ohjelmointialustana toimi ASP.NET 2.0. Tietojen tallentamiseen käytettiin SQL-Server 2005 tietokantaohjelmistoa. Osa ohjelmassa käytettävästä tiedosta oli XML-muotoista, jota voitiin tallentaa SQL-Server –tietokannasta löytyviin kenttiin XML-rakenteen muotoisena. Ohjelmiston määrittelyraportin tekemiseen käytettiin Microsoft Word ohjelmaa ja vuokaavion tekeminen suoritettiin Microsoft Visio ohjelmalla. Ohjelman testaamiseen käytettiin luonnollisesti Internet-selainta, millä valmista ohjelmaakin käytetään.

3.3 Vaatimusmäärittely

Projektin toteuttaminen alkoi palaverista, jossa määriteltiin ohjelmistoon tulevat vaatimukset. Palaverissa käsiteltiin muutenkin ohjelmiston toimintaperiaatteita sekä miten ohjelmistoa lähdetäisiin toteuttamaan. Palaverissa ilmeni myös se, että ohjelmisto toteutetaan valmiin ohjelmistopohjan päälle. Keskeisimpinä toiminnallisina vaatimuksina Luovanet asiakkuuksienhallintaohjelmistolle oli seuraavanlaiset asiat:

- Ohjelmistossa tulee olla asiakkaiden yhteystietojen, yhteyshenkilöiden ja asiakaskohtaisen laskutusjärjestelmän ylläpitoon tarvittavat mekanismit.
- Ohjelmiston tulee pitää sisällään myös laskutusjärjestelmän hallinta, josta voidaan muun muassa syöttää uusia laskuja järjestelmään.

Ohjelmiston vaatimusmäärittelyssä oli alussa enemmänkin kohtia kuin nämä aiemmin mainitut, mutta opinnäytetyöhön käytettävän ajan rajallisuuden takia osa vaatimuksista päätettiin poistaa ohjelmistosta.

Ohjelmistolle ei asetettu kovin tarkasti kaikkia vaatimuksia, vaan lähinnä määriteltiin kehys jonka sisällä toimitaan. Näin ollen ohjelmiston suunnittelun ja toteuttamisen suhteen oli

mahdollista kokeilla omia ratkaisuja. Hyötynä tässä menetelmässä on se, että tekemistään ratkaisuista oppii joka tapauksessa vaikka ne ei eivät toimisi hyvin käytännössä. Haittapuolena ohjelmiston tekemiseen käytettävä aika lisääntyy, koska kokemattomana ohjelmoijana voi joutua ohjelmistoa muuttamaan montakin kertaa toimimattomien ratkaisujen takia.

3.3.1 Tietokanta

Tietokantaan suoranaisesti kohdistuvia vaatimuksia ei paljoa ollut. Tärkeimpänä asiana oli ottaa huomioon, että osa ohjelmistossa käsiteltävistä tiedoista tulee suunnitella dynaamisiksi. Dynaamisuudella tarkoitetaan sitä, että esimerkiksi asiakkaalla on joitain tietoja joiden määrää ei voida ennalta tietää. Käytännössä tämä ohjelman kannalta tarkoittaa sitä, että joitain tietoja pitää pystyä lisäämään asiakkaalle ennalta määräämätön määrä. Kyseisen kaltaisia tietoja asiakkaalla oli muun muassa osoitteet ja yhteyshenkilöt.

Dynaamisten tietojen määrittelyminen vaikuttaa huomattavasti ohjelmistossa käytettävien tietokantojen suunnitteluun. Tietokantojen rakenteen tulisi olla oikein suunnitellut ennen kuin ohjelmointia kannattaa aloittaa. Myöhemmin ohjelmaan tehtävät tietokantojen rakenteen muutokset voivat aiheuttaa huomattavan paljon lisätyötä uudelleen ohjelmoimisen muodossa. Luovanet asiakkuuksienhallintaohjelmiston kaltaisessa sovelluksessa ohjelmiston sydämenä voidaan hyvinkin pitää tietokantoja, joten vaatimusmäärittelyssä tulisi ilmetä mahdollisimman tarkasti tietokantoihin vaikuttavat tekijät. Tosin on muistettava, että suunnitteluvaiheessa huomattava osa tietokantoihin vaikuttavista tekijöistä tulee vielä ilmi.

3.3.2 Käyttöliittymä

Käyttöliittymän määrittelyn osalta suurin tekijä oli se, että ohjelmisto toteutetaan valmiina olevan ohjelmistopohjan päälle. Ohjelmistopohjan käyttäminen yhdennäköistää tehtävän ohjelmiston paremmin yrityksen muihin sovelluksiin nähden. Valmiin ohjelmistopohjan käyttäminen asettaa myös selvät raamit toteutettavalle käyttöliittymälle. Käyttöliittymälle ei

kuitenkaan asetettu sen tarkemmin mitään muita vaatimuksia. Tietenkin yleisesti sen tulee olla mahdollisimman hyvin käytettävä. Ohjelmistossa käytettäviä kontrollejakaan ei määritelty sen tarkemmin. Näin ollen ohjelmiston käyttöliittymän toteuttamisen suhteen voidaan kokeilla käyttää omia ratkaisuja. Huomioitavaa tietenkin on se, että huonosti toimivat ratkaisut pitää muuttaa toimiviksi.

3.4 Työprosessi

Projektin toteuttaminen koostui kolmesta selkeästi isommasta työprosessista. Näitä prosesseja olivat suunnittelu, ohjelmoiminen sekä testaus. Suunnittelun tärkeyttä ei voi yhtään vähätellä kun ollaan toteuttamassa isompaa ohjelmistoa. Ilman suunnittelua ohjelmiston valmiiksi saaminen on käytännössä lähes tulkoon mahdotonta tai ainakin ohjelmiston toteuttamiseen käytettävä aika on huomattavasti pitempi.

Suunnitelman oltua valmis voitiin ohjelmointia lähteä toteuttamaan tehokkaasti. Ohjelmoinnin edetessä huomattiin, että suunnitelmaan määriteltyjä tietokantatauluja on sama korvata XML-tauluilla. Näin ollen ohjelma ei täysin vastannut suunnitelmassa olleita asioita, mutta suunnitelman tarkoituksenahan onkin antaa tehtävälle asialla pohja miten asiaa lähdetään toteuttamaan.

Valmiiseen ohjelmaan tehtiin viimeisenä käyttäjäsyyötteiden testaaminen. Tämä testaamismenetelmä paljastikin ohjelmasta selviä virheitä. Virheistä johtuen ohjelmaa jouduttiin vielä korjaamaan, että ohjelmassa ilmenneet virheet saatiin poistettua.

3.4.1 Suunnittelu

Ohjelmistoon vaadittujen ominaisuuksien pohjalta alkoi määrittelyraportin kirjoittaminen. Raporttiin selvennettiin mitä tietoja tallennetaan aina jotain kokonaisuutta varten. Esimerkiksi mitä tietoja tarvitaan ja pitää ottaa huomioon, että saadaan vaikka asiakkaan yhteystiedot tallennettua. Kyseisessä tapauksessa oli huomioitava se, että esimerkiksi postiosoitteiden tulee olla dynaamisia. Tallennettavien tietojen selvityksessä alkoi myös

tietokantataulujen suunnittelu. Tietokantataulujen määrä ohjelmistossa väheni kuitenkin ohjelmoinnin edetessä, koska tiedon tallentaminen ja käsitteleminen XML-muotoisena osoitti omat voimavaransa verrattuna useammalla tietokantataululla toimimiseen. Ohjelmaan tulevista XML-taulukkorakenteista tehtiin erilliset kuvaukset määrittelyraporttiin.

Esimerkki XML-taulukkorakenteen määrittelystä:

```
Tietokannanimi.Taulunnimi
<Tuotteet>
  <Tuote>
    <Nimi></Nimi>
    <Hinta></Hinta>
  </Tuote>
</Tuotteet>
```

Kuvio 1. Yksinkertainen esimerkki XML-taulukkorakenteen määrittelystä.

Määrittelyraporttiin kirjattiin myös ylös, että millaisia sivuja nettisivustolle tarvitaan ja mitä toimintoja näillä sivuilla on mahdollista tehdä. Sovelluksessa tarvittavien sivustojen tiedostonimet määriteltiin myös ennakkoon ennen varsinaisen ohjelmoimisen aloittamista. Toteutettavan ohjelman käyttöliittymästä ei kuitenkaan tehty tarkkoja suunnitelmia. Näin ollen käyttöliittymää kehitettiin ohjelmoimisen edetessä. Tarvittavien määrittelyiden jälkeen ohjelmistosta tehtiin vuokaavio Microsoft Visio ohjelmaa apuna käyttäen.

3.4.2 Ohjelmointi

Varsinaisen sovelluksen ohjelmoiminen aloitettiin vasta, kun suunnitteluvaihe oli ohi. Suunnitteluvaiheen edetessä kuitenkin tutustuttiin jo käytettävään ohjelmistopohjaan jonka päälle sovellus toteutetaan. Ohjelmistopohjan rakenteen ohjelmoiminen paremmin sovellukselle sopivaksi hoidettiin suunnitteluvaiheen kanssa rinnakkain.

Ohjelmiston ohjelmoiminen pyrittiin toteuttamaan asiakokonaisuus kerrallaan vuokaavioon piirrettyjen kokonaisuuksien mukaan. Suurin osa ohjelmoinnin toteutuksesta oli tässä projektissa tietojenkäsittelyyn liittyvää. Tähän asiaan liittyvää on muun muassa tietojen

hakeminen tietokannasta, tietojen näyttäminen sivuilla ja tietojen tallentaminen tietokantaan. Ohjelmoinnissa suurimpina haasteina oli opetella XML-tietojen käsittelyä koodissa. Haastavaa tämä on sen takia, koska ei ole aiempaa kokemusta XML-tietojen käsittelystä.

Ohjelmoimisen edetessä käytetystä tietokannasta vähennettiin tauluja ja korvattiin niitä SQL-Server 2005 tietokantaohjelmistosta löytyvillä XML-kentillä. Päälimmäisenä syynä tähän oli, että kerran XML-tietojen käsittelyn opittua oli sama vähentää taulujen määrää, jotta saataisiin käytetystä tietokannasta yksinkertaisempi. Yksinkertaisemman tämä tekee siitä siinä mielessä, että tietokannassa ei ole enää niin paljon tauluja ja näiden välisiä yhteyksiä. Sovelluksen mahdollinen jatkokehittäminenkin on helpompaa, kun tietokannasta saadaan tehtyä mahdollisimman yksinkertainen rakenteeltaan.

3.4.3 Testaus

Sovelluksen testaamisesta ei tehty mitään testausraporttia tai mitään tarkempaa suunnitelmaa, että miten testaaminen suoritettaisiin. Suurin osa testaamisesta tapahtui jo ohjelmaa tehdessä, koska aina uuden toiminnon tultua ohjelmistoon sitä piti testata, että toimiihan se oikein. Tämän tapainen testaaminen ei kuitenkaan ole kovin varmaa, koska ajatus kuitenkin on varsinaisen ohjelman tekemisessä eteenpäin ja näin ollen jotain testattavia asioita voi helposti unohtua. Tällä tavalla kuitenkin suurin osa ohjelman virheistä karsiintuu pois ennen valmiin ohjelman testausvaihetta. Toisaalta ohjelmaan voi jäädä todella kriittisiä virheitä joiden korjaaminen vaatii paljon aikaa.

Ohjelman valmistuessa viimeisenä vaiheena suoritettiin vielä käyttäjänsyötteiden testaaminen. Kyseisen testin tarkoituksena on, että ohjelma toimii oikein käyttäjän antamiin syötteisiin nähden. Tällä testillä ohjelmasta löytyikin kriittinen virhe, jota ei ohjelmoinnin yhteydessä tapahtuneessa testaamisessa löytynyt. Tietynlaisten erikoismerkkien käyttö XML-tyylisten tietorakenteiden kanssa sotki XML-rakenteen täysin. Samainen testi paljasti myös erikoismerkkien näkymisen väärin tietyissä tilanteissa nettisivulla.

Testin osoittamien virheiden korjaamiseen meni aikaa lähemmäs kaksi työpäivää. Virheen huomaaminen ohjelmiston aiemmassa vaiheessa olisi vähentänyt huomattavasti tarvittavan työn määrää mikä korjaamiseen tuhlaantui. Aiemmin ohjelmiston kehityksessä huomattua

virhettä ei olisi tarvinnut korjata niin moneen paikkaan mitä nyt ohjelmiston valmiissa vaiheessa löydettyä. Kaiken korjaamisen jälkeen ohjelmistoa pitää testata vielä kertaalleen, että eihän johonkin kohtaan virhe vielä jäänyt, kun korjaamista jouduttiin suorittamaan niin moneen paikkaan.

Jälkikäteen ajateltuna kyseisenlainen testaaminen olisi kannattanut jo toteuttaa aiemmassa vaiheessa ohjelmaa, koska ohjelmiston toiminta perustuu suurimmalta osalta juuri saman tyyllisen tiedon käsittelemiseen eri muodoissaan. Näin ollen virhe olisi löytynyt jo huomattavasti aiemmin ja korjaamiseen käytettävä työmäärä sekä uusien mahdollisten virheiden syntyminen ohjelmaan olisi vähäisempää.

3.5 Toteutunut ohjelmisto

Toteutuneen ohjelmiston toiminnalliset vaatimukset vastaavat vaatimusmäärittelyn vastaavia asioita. Suunnitteluraporttiin määriteltyihin asioihin nähden toteutunut sovellus ei kuitenkaan ihan täysin vastaa määrittelyä. Syynä tähän on se, että ohjelmistosta saatiin parempi, kun suunnitteluraporttia ei noudatettu orjallisesti. Suurin muutos määrittelyyn nähden koski toteutuneen ohjelmiston tietokantarakennetta.

Käyttöliittymän ulkoasua ei määritelty sen tarkemmin mihinkään määrittelyraportteihin. Näin ollen toteutuneen ohjelmiston käyttöliittymä kehittyi omaan muotoonsa ohjelmoimisen edetessä. Käyttöliittymän puuttuminen määrittelystä lisäsi hieman tehtävän työn määrää, koska käyttöliittymää piti muokkaila, jos jokin asia ei toiminut riittävän hyvin. Oppimisen kannalta tämä kyseinen menetelmä kuitenkin oli hyvä vaihtoehto. Toteutettavalle ohjelmistolle ei asetettu kovin tarkasti mitään muita määritelmiä, vaan lähinnä kehykset minkä päälle ohjelmistoa toteutetaan.

3.5.1 Käyttöliittymä

Toteutuneen ohjelman käyttöliittymään vaikutti huomattavasti käytettävä ohjelmistopohja. Valmiin ohjelmistopohjan käytöllä saatiin käyttöön muun muassa välilehdet, listat, poiston

vahvistus ja sivustorakenne ohjelmalle. Toteutuneessa ohjelmassa on periaatteessa kaksi päänäköä. Nämä päänäkömät ovat Asiakkaat ja Tuotteet välilehdet. Kuviossa 2. on kuva asiakkaat näkymästä. Kuvaa on muokattu keskimmäisen komponentin kohdalta, että kuva saataisiin paremmin näkymään opinnäytetyössä.



Kuvio 2. Kuva asiakkaat näkymästä.

Asiakkaat välilehden tulisi olla kuvassa valittuna, mutta kuva on otettu paikallisesti koneella suoritettuna, jolloin tyylitiedosto ei toimi oikein. Ohjelman vasemmassa reunassa olevassa listarakenteessa näkyy ohjelmaan syötetyt asiakkaat. Oikean puoleisessa komponentissa on puolestaan aina kyseisellä sivustolla tehtävät toiminnot. Keskimmäisessä komponentissa ei tässä tilanteessa ole mitään tietoa. Kyseiseen komponenttiin voisi vaikka tehdä tälle sivulle yhteenvedon, että kuinka paljon asiakkaita on ohjelmaan syötetty.

Tuotteet näkymän rakenne on täsmälleen samanlainen kuin asiakkaat näkymänkin. Tuotteet näkymässä listarakenteessa näkyy puolestaan ohjelmistoon syötetyt tuotteet. Ohjelmistoon syötettävät tuotteet ovat siis Luovanetin ja Luovaliiikkeen omia tuotteita joita ne myyvät asiakkailleen. Asiakkaalle kytkettävien tuotteiden ja projektien laittaminen ei siis tapahdu tuotteet näkymästä.

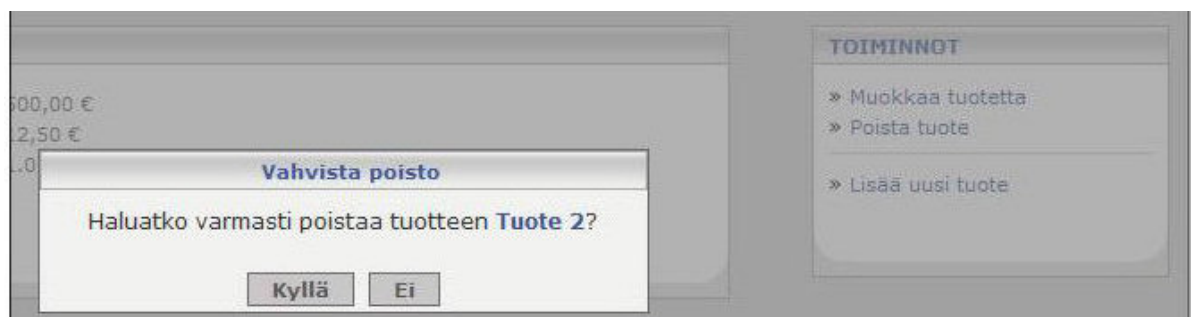
Ohjelman vasemmalla reunalla oleva listarakenne on aina sivustolla näkyvissä. Listarakenteen sisältö riippuu luonnollisesti siitä kummalla välilehdellä liikutaan. Kuviossa 3. on kuva tuotteet näkymästä, kun on valittu joku tuote listarakenteesta. Valittuna oleva tuote näkyy niin listarakenteessa kuin keskimmäisen komponentin otsikkorakenteessa.

Keskimmäisessä komponentissa näkyy tuotteen tarkemmat tiedot. Asiakkaat näkymässä valittu asiakas näkyy siis samalla lailla kuin tuote näkymässä valittu tuotekin. Ohjelmassa oleva keskimäinen komponentti voi jossain tilanteissa yltää oikeaan reunaan asti ja näin ollen toiminnot komponentti ei ole silloin näkyvillä.



Kuvio 3. Kuva tuotteet näkymässä valitusta tuotteesta.

Valmiin ohjelmistopohjan käytöllä saatiin ohjelmistoon käytettäväksi poiston vahvistukseen tarkoitettu kontrolli. Kyseistä kontrollia käytetään ohjelmassa aina vahvistusta vaativien tietojen poistossa. Kontrolli niin sanotusti lukitsee kaikki muut toiminnot Internet sivulta, joten käyttäjä voi tässä tilanteessa painaa vain kyllä ja ei nappeja. Kaikki sivun taustalla olevat asiat näkyvät tummemman harmaalla ja näihin ei voida vaikuttaa ennen kuin valinta on kontrollista suoritettu. Kuviossa 4. on kuva poiston vahvistukseen tarkoitettua kontrollista käytännössä.



Kuvio 4. Kuva poiston vahvistukseen tarkoitettua kontrollista.

Uutta asiakasta ohjelmistoon lisättäessä on käytetty apuna niin sanottua velhoa (wizard). Kuviossa 5. on kuva asiakkaan tietojen lisäämisestä velhoa käyttäen. Velhon vasemmalla reunassa olevassa listassa näkyy tummennettuna mitä tietoja ollaan asiakkaalle parhaillaan syöttämässä. Seuraava painiketta painamalla siirrytään syöttämään listassa seuraavaksi tulevia tietoja. On kuitenkin huomattava, että osa tiedoista voi olla pakollisia ja näin ollen ne tulee syöttää ennen kuin ohjelma antaa vaihtaa seuraavaa sivua velhosta. Ohjelma ilmoittaa pakollisista tiedoista, jos niitä ei ole täytetty ja painetaan ”seuraava” painiketta. Velho on yksi niistä toiminnoista joka venyttää keskimmäisen komponentin oikeaan reunaan asti. Kuvasta on muokattu vasemmalla oleva asiakkaat lista pois.

Kuvio 5. Kuva asiakkaan lisäämiseen tarkoitetusta velhosta.

3.5.2 Tietokanta

Ohjelmistoon toteutunut tietokantarakenne muuttui suunnitelmassa aiemmin olleesta määrittelystä. Tietokantataulujen määrä väheni tietokannasta, kun näitä korvattiin SQL-Server 2005 tietokantaohjelmasta löytyvillä XML-kentillä. Suunnitelmassa oli alussa määriteltynä noin kuusi kappaletta tietokantatauluja, mutta valmiissa ohjelmassa niitä oli enää vain kaksi kappaletta. Toteutuneeseen ohjelmaan näin ollen tuli tietokantataulut asiakkaan tiedoista ja yrityksen tuotteista. Toteutuneet tietokantataulut siis vastaavat suoraan ohjelmassa olevien näkymien asiakkaat ja tuotteet kohtia. Toisaalta ovathan nämä

kaksi asiaa selvästi omia kokonaisuuksiaan, joten niistä tulee tietokantaan omat tietokantataulunsa.

Asiakkaat tauluun kohdistui suurin muutos toteutuneessa tietokantarakenteessa. Kyseiseen tauluun XML-kenttiä lisäämällä saatiin vähennettyä tietokantataulujen määrää ohjelmassa. Ilman tätä toimenpidettä asiakkaat taulusta olisi ollut useampien taulujen välisiä yhteyksiä ja näin ollen tietokannasta olisi tullut sekavampi. Toteutuneeseen ohjelmaan tuli nyt erittäin selkeä tietokantarakenne, ja tästä seurauksena ohjelman jatkokehittäminen on nyt paljon helpompaa. Taulukossa 4. on esitetty toteutuneen ohjelmiston asiakkaat tietokantataulu.

Taulukko 4: Toteutuneen ohjelmiston asiakkaat tietokantataulu.

Customer_Company		
Kentän nimi	Tietotyyppi	Selite
Id	int	ident
Status	int	Tila: 0= normaali, -1= poistettu
Name	nvarchar	Nimi
Addresses	xml	Osoitteet
Homepage	nvarchar	Kotisivu
YCode	nvarchar	Y-tunnus
Comments	xml	Kommentit
Created	smalldatetime / default getDate()	Luotu
Modified	smalldatetime	Muokattu
Deleted	smalldatetime	Poistettu
Products	xml	Tuotteet ja projektit
Contacts	xml	Yhteyshenkilöt

Asiakkaat tietokantataulun ja tuotteet taulun välillä ei ole suoranaista yhteyttä. Tällä siis tarkoitetaan sitä, että jos tuotteet taulusta mennään muuttamaan vaikka myytävän tuotteen versionumeroa ja hintaa, niin asiakkaan tiedoista tulee löytyä todellisen myydyn tuotteen tiedot. Asiakkaan tiedot vääristyisivät, jos taulujen välillä olisi suora yhteys, koska eihän asiakkaalle myyty uutta versiota uudella hinnalla, vaan sen hetkinen versio mikä oli tarjolla. Kyseisen asian välttäminen on toteutettu lisäämällä asiakkaat tauluun ”Products” XML-kenttä. Kyseiseen kenttään kopioidaan asiakkaalle kytkettyjen tuotteiden tiedot XML-

rakenteen muotoiseksi tuotteet taulusta. Samaiseen kenttään myös lisätään tiedot projektiluontoisista tilauksista.

Toteutuneen ohjelmiston tuotteet tietokantataulu on rakenteeltaan yksinkertainen. Tuotteet taulussa ei ole yhtään kohtaa joka vaatisi esimerkiksi XML-kenttien käyttämisen taulussa. Taulukossa 5. on esitetty toteutuneen ohjelmiston tuotteet tietokantataulu.

Taulukko 5: Toteutuneen ohjelmiston tuotteet tietokantataulu.

Customer_Product		
Kentän nimi	Tietotyyppi	Selite
Id	int	ident
Status	int	Tila: 0= normaali, -1= poistettu
Name	nvarchar	Nimi
LicensePrice	money	Lisenssihintaa
MonthPrice	money	Kuukausihinta
Version	nvarchar	Versionumero
Created	smalldatetime / default getDate()	Luotu
Modified	smalldatetime	Muokattu
Deleted	smalldatetime	Poistettu
Description	ntext	Kuvaus

Ohjelmistoon toteutunut tietokanta on erittäin yksinkertainen. Nykyisellään tietokanta pitää sisällään vain kaksi tietokantataulua ja näiden välillä ei ole suoranaista yhteyttä toisiinsa. Ilman XML-kenttien käyttämistä tietokantatauluissa olisi tietokantatauluja tullut huomattavasti enemmän ohjelmistoon ja näiden välisiä yhteyksiä olisi ollut paljon.

POHDINTA

Opinnäyte oli erittäin mielenkiintoinen työ. Työ eteni kohtuullisen tasaisesti eteenpäin, eikä toteuttamisessa näin ollen ollut kovin paljoa ongelmatilanteita joiden ratkaisemiseen olisi mennyt kauan aikaa. Osaltaan työn etenemiseen vaikutti työskentely Luovanetin tiloissa, koska sain tarvittaessa apua ohjelman toteuttamisessa ilmenneisiin ongelmiin. Yhteistyö toimeksiantajan kanssa oli erittäin hyvää.

Ohjelmiston toteuttamisessa minulla oli kohtuullisen vapaat kädet kokeilla omia ratkaisuja sovellukseen. Toimeksiantaja määritteli lähinnä ohjelmiston toiminnalliset vaatimukset ja ohjelmistossa huomioitavat asiat, joiden pohjalta toteutin määrittelyraportin ja hyväksytin sen ennen ohjelmoimisen aloittamista. Tällainen vapaampi ohjelmiston toteuttaminen oli hyvä asia oman oppimisen suhteen, koska jos jokin tekemistäni ratkaisuista ei toiminut sain kuulla tähän asiaan perustelut miksi asiaa ei kannata toteuttaa kyseisellä tavalla.

Opinnäytetyössä haastavin osuus oli XML-tietojenkäsittelyn opetteleminen. Minulla ei ollut aikaisempaa kokemusta kyseisestä asiasta ennen tätä projektia. Opinnäytetyön myötä huomasin kuinka kätevää tietojen käsitteleminen on XML-muotoisena ohjelmistossa. Lisäksi huomasin kuinka ohjelmiston tietokantarakennetta saatiin yksinkertaistettua korvaamalla tietokantatauluja SQL Server tietokantaohjelmasta löytyvillä XML-kentillä.

Toteutuneessa ohjelmistossa kaikki osa-alueet onnistuivat mielestäni hyvin. Toisaalta ohjelmiston selkeämmät puutteet ja toimimattomimmat ratkaisut huomataan vasta pitempi aikaisessa käytössä. Ohjelmistossa ei myöskään mielestäni mikään onnistunut huonosti, koska ohjelmistoa muutettiin paremmaksi jos jokin asia ei ollut hyvin tehty. Projektin

kannalta ohjelmiston testaaminen olisi kannattanut tehdä harkitummin, kuin miten se nyt toteutettiin.

Teoriaosuudessa asiakkuuksienhallinnan käsitteleminen sai minut todellisuudessa tajuamaan millaista asiakkuuksienhallinta voi olla. Toteutettu ohjelmisto ei siis ole lähellekään niin laajaa asiakkuuksienhallitsemista mitä teoriaosassa käsiteltiin. On tietenkin muistettava, että asiakkuuksienhallinta voi olla erilaista eri yrityksissä ja kaikkialla samat ratkaisut eivät ole niin perusteltuja. Toteutunut ohjelmisto on vielä alkutekijöissään, koska osa ohjelmistossa olleista vaatimuksista poistettiin opinnäytetyöhön käytettävän ajan rajallisuuden takia. Ohjelmiston jatkokehittäminen on vielä täysin mahdollista ja tätä ajatusta tukee vahvasti tietoteknisen rakenteen yksinkertaistuminen.

Opinnäytetyön tekemisen myötä opin paljon asiakkuuksienhallinnasta ja siitä miksi sitä nykyaikana on perusteltua toteuttaa. Teoriaosan käsitellessä asiakkuuksienhallintaa se tuki vahvasti toteutunutta ohjelmistoa ja näin ollen antoi laajempaa näkemystä asiakkuuksienhallinnan erilaisiin toteutuksiin. Käytännönsä myötä sain taas paljon lisää kokemusta ohjelmoimisesta ja ohjelmiston suunnittelemisesta. Samalla myös opin XML-tietojenkäsittelyn ja näin kuinka kätevää tämä voi parhaimmillaan olla.

LÄHTEET

Ala-Mutka, J & Talvela, E. 2004. Tee asiakassuhteista tuottavia. Helsinki: Talentum

Helia 2007. <http://myy.helia.fi/~vanvu/xmlkurssi/mikaxml.html> (Luettu 14.5.2007)

Järvinen, T. 2002. Asiakastiedon analysointi ja käyttö asiakkuuksien johtamisessa. Jyväskylän yliopisto. Tietojenkäsittelytieteiden laitos. Kandidaatintutkielma. www.slk.fi/h/ruipi/pdf/tomjarvi_acrm.pdf

Koskinen, K. 2005. Asiakkuus ja sen hallinta CRM-ohjelmiston avulla -Case mySAP. Tampereen yliopisto. Johtamistieteiden laitos. Pro gradu -tutkielma. <http://tutkielmat.uta.fi/pdf/gradu00739.pdf>

Lehtonen, T & Puhto, J. 2002. Asiakkuudenhallinta toimitilapalveluympäristössä. Teknillinen korkeakoulu. Teknillisen korkeakoulun rakentamistalouden laboratorion raportti 209. www.rta.tkk.fi/Julkaisut/pdf_raportit/Raportti%20209.pdf

Luovanet Oy 2007. <http://www.luovanet.fi/> (Luettu 11.4.2007)

Mäntyneva, M. 2000. Asiakkuudenhallinta. Helsinki: WSOY

Saarikkomäki, K. 2006. Asiakkuuksien hallinta.

Hämeen ammattikorkeakoulu. Opinnäytetyö.

<http://www4.hamk.fi/julkaisut/lataa.php?id=NDYxfDIwMDcwNDI0>

TIEKE ry 2007 a.

http://www.tieke.fi/verkkokaveri/teemat/asiakkuuden_hallinta/usein_kysytyt_kysymykset/

(Luettu 9.4.2007)

TIEKE ry 2007 b.

http://www.tieke.fi/verkkokaveri/teemat/asiakkuuden_hallinta/asiakkuudenhallinta_ja_sen_merki/

(Luettu 9.4.2007)

TIEKE ry 2007 c.

http://www.tieke.fi/verkkokaveri/teemat/asiakkuuden_hallinta/crm-sovellusratkaisun_rakenne/

(Luettu 24.4.2007)

Vuopio, V-P. 2002. Asiakkuuden hallinta suunnittelu & toteutus - urakointimallissa.

Teknillinen korkeakoulu. Teknillisen korkeakoulun rakentamistalouden laboratorion selvitys 45.

www.rta.tkk.fi/Julkaisut/pdf_selvitykset/Selvitys%2045.pdf