



TAMPEREEN
AMMATTIKORKEAKOULU

OPINNÄYTETYÖ

VAATEALAN YRITYKSEN KANTA-ASIAKASJÄRJESTELMÄ

Anu Tanneraho

Tietojenkäsittelyn koulutusohjelma
lokakuu 2008
Työn ohjaaja: Petri Heliniemi

TAMPERE 2008



Tekijä(t)	Anu Tanneraho	
Koulutusohjelma(t)	Tietojenkäsittely	
Opinnäytetyön nimi	Vaatealan yrityksen kanta-asiakasjärjestelmä	
Työn valmistumis- kuukausi ja -vuosi	10/2008	
Työn ohjaaja	Petri Heliniemi	Sivumäärä: 39

TIIVISTELMÄ

Tampereen keskustassa avattiin joulukuussa 2007 skandinaaviseen pukeutumistyyliin erikoistunut vaateliike. Yrityksellä oli avajaisista lähtien käytössä kanta-asiakasjärjestelmä mutta kanta-asiakkaiden määrän kasvaessa huimaa vauhtia yritys tarvitsi sovelluksen jonka avulla kanta-asiakkaiden tiedot voisi saada talteen. Kanta-asiakkailta ei myöskään ollut kanta-asiakasnumeroita joten mahdollisten kanta-asiakasalennusten saaminen yrityksen verkkokaupasta ei ollut mahdollista.

Kanta-asiakasjärjestelmän päätavoitteina asiakkaiden tietojen tallentamisen lisäksi olivat saada kanta-asiakkaille kanta-asiakasnumerot joita voisi hyödyntää myös yrityksen verkkokaupan yhteydessä, asiakkaille itselleen hallintamahdollisuus omiin tietoihinsa sekä sovellus jonka avulla ylläpitäjä voisi lähettää kanta-asiakaspostin helpommin kuin toimeksiannon lähötötilanteessa.

Kanta-asiakasjärjestelmän toteutus tapahtui ammattikorkeakoulussa opituilla tiedoilla ja taidoilla. Käytännön sovellus toteutettiin hyödyntäen PHP-ohjelmointikieltä sekä MySQL-tietokantaa.

Toimeksiannon päätuloksina yritys sai kanta-asiakkaiden tiedot talteen tietokantaan ja kanta-asiakkaat saivat itselleen kanta-asiakasnumerot. Yrityksen edustaja sai sovelluksen jonka avulla hän itse voi lisätä tai tarpeen mukaan muokata tai poistaa asiakkaiden tietoja tietokannassa. Kanta-asiakkaiden on kanta-asiakasnumeroiden ansiosta mahdollista saada tuote-etuisuuksia tilatessaan tuotteita yrityksen verkkokaupasta.

Kanta-asiakasjärjestelmää hyödynnetään jatkuvasti yrityksen myynninedistämässä. Toteutumatta jääneet tavoitteet ovat realistisia kehittämissuunnitelmia kanta-asiakasjärjestelmän laajentamiselle.



Author(s)	Anu Tanneraho	
Degree Programme(s)	Business Information Systems	
Title	Regular customer system for a clothing company	
Month and year	October 2008	
Supervisor	Petri Heliniemi	Pages: 39

ABSTRACT

Clothing company specializing in Scandinavian dressing style was opened in central Tampere in December 2007. From the beginning, the company used their own regular customer system. When the number of regular customers increased significantly, the company needed new software which would be able to store information about regular customers. Regular customers didn't have customer ID numbers either, so it was not possible for them to get benefits from the company's webstore.

Main goals of the regular customer system for a clothing company were to store information about regular customers, and to get customer ID numbers for them. These ID numbers could be used at the company's webstore. With this software, regular customers could also control information about themselves. Software could also help the administrator in sending information to regular customers via e-mail easier than in the beginning of the assignment.

The realization of the regular customer system was performed with information and skills learned in the polytechnic. The practical application was carried out utilizing the PHP programming language and MySQL database.

The main achievements of the assignment were a database used to store information about regular customers of the company, and customer ID numbers given to them. The administrator got software which can be used to add, delete or edit information about regular customers in the database. Regular customers can also get benefits from the company's webstore with their customer ID numbers.

This regular customer system is used all the time in the company. Goals yet to be achieved are realistic in the future development of this software.

Sisällysluettelo

1	Johdanto	5
2	Tausta ja tavoitteet	8
3	Toteutuksen kartoitus	10
	3.1 Ohjelmointikielet	10
	3.2 Tietojen varastointi	11
	3.3 Tietojen suojaus	12
	3.4 Asiakastietojen käyttö	13
	3.5 Ulkoasutoteutus	13
4	Suunnitteluprosessi	15
	4.1 Tekniset vaatimukset	15
	4.2 Ulkoasuvaatimukset	16
5	Toteutusprosessi	17
	5.1 Tekniset toteutustavat	17
	5.2 Käytännön toimintatavat	17
	5.3 Sovelluksen toteutus	18
	5.4 Sovelluksen käyttöönotto	23
6	Ratkaisukeinot ongelmatilanteissa	25
	6.1 Päivitykset	25
	6.2 Tekniset ongelmat	25
	6.3 Salasanat	26
7	Lopputulokset	27
	7.1 Onnistumiset	27
	7.2 Ongelmat	28
	7.3 Saavutetut tavoitteet	29
8	Johtopäätökset	33
	Lähteet	35
	Liitteet	37

1 Johdanto

Toteutin vaatealan yritykselle kanta-asiakasjärjestelmän saamani toimeksiannon pohjalta. Tarkastelen tässä opinnäytetyöraportissani toimeksiannon lähtötilanteen lisäksi kanta-asiakasjärjestelmän toteutuksen teoriaa, sen toteutus- ja suunnitteluprosessia sekä lopputulosta. Raportin lopussa käsittelen myös järjestelmän kehitys- ja parannusehdotuksia. Toteutuksen teoriaa sekä omaa käytännön osuuttani järjestelmän aikaansaamiseksi olen tarkastellut suuntautumisvaihtoehtoni hypermedian näkökulmasta.

Toimeksiantajayritys on joulukuussa 2007 avattu, Tampereen keskustassa sijaitseva, vakituisesti kaksi henkilöä työllistävä, skandinaaviseen pukeutumistyyliin keskittynyt vaateliike. Yritys on erikoistunut laadukaiden ja tyylikkäiden vaatteiden sekä muiden asusteiden myyntiin. Yrityksen liiketilat eivät ole kovin suuret, mutta yrityksellä on tietty kohdeyhmä, eli tyylietoiset (nuoret) aikuiset, jotka arvostavat tuotteiden laatua ja tyyliä sekä laadukasta ja henkilökohtaista asiakaspalvelua. Osa vaateliikkeessä myytävistä tuotemerkeistä ei saa muualta Tampereelta.

Päätavoitteena toimeksiannossani oli saada asiakkaiden tiedot talteen sekä antaa jo olemassa oleville sekä tuleville kanta-asiakkaille kanta-asiakasnumerot. Toissijainen, mutta tärkeä tavoite, oli luoda sovellus, jonka kautta asiakkailla olisi mahdollisuus hallinnoida omia tietojaan sekä yrityksen edustajalle sovelluksen kautta mahdollisuus hallita tai tarpeen mukaan poistaa tai muokata kanta-asiakkaiden tietoja. Eräänä tavoitteena oli myös saada sovellus, jonka avulla yrityksen edustaja lähettäisi aiempaa helpommin kanta-asiakkaille kanta-asiakaspostia kanta-asiakkaiden sähköpostiosoitteisiin siten, että lähettäjän ei tarvitsisi esimerkiksi valita sähköpostinsa osoitekirjasta kanta-asiakkaiden tietoja erikseen yksitellen, vaan sekä postin kirjoittaminen että lähettäminen tapahtuisivat sovelluksen, eikä yrityksen edustajan oman sähköpostin välityksellä.

Kanta-asiakasjärjestelmän uudistuttua tarkoituksena oli luoda toiminnot, joiden ansiosta Internetissä kanta-asiakkaaksi liittyvä henkilö saisi sähköpostiinsa tiedon kanta-asiakkaaksi liittymisestä sekä kanta-asiakasnumeron ja oletussalasanan, jonka asiakas voisi käydä vaihtamassa mielekseen sivustolla, jonka kautta kanta-asiakkaat voisivat hallinnoida omia tietojaan.

Lähtötilanteeseen verrattuna valmis, kaikkien tavoitteiden mukaisesti tehty sovellus, antaisi yrityksen henkilöstölle aiempaa enemmän aikaa keskittyä yrityksen varsinaisiin töihin. Toivotusti toteutetun sovelluksen avulla Internetissä kanta-asiakkaaksi liittyvien henkilöiden tiedot mene-

vät automaattisesti tietokantaan, eikä yrityksen edustajan tarvitse syöttää käsin kanta-asiakkaaksi liittyvän henkilön tietoja kuin ainoastaan niiltä henkilöiltä, jotka liittyvät kanta-asiakkaaksi vaateliikkeessä, sillä tietokannassa on tietty määrä varattuja kanta-asiakasnumeroita vaateliikkeessä liittyviä kanta-asiakkaita varten. Näiden kanta-asiakasnumeroiden on tarkoitus olla valmiiksi kirjoitettu liikkeessä olevissa kanta-asiakaskaavakkeissa sekä niiden vastinkappaleissa eli kanta-asiakaskorteissa ennen asiakkaiden liittymistä.

Kanta-asiakasnumeroiden hyöty näkyy käytännössä erityisesti työharjoitteluajanani yritykselle toteuttamani verkkokaupan yhteydessä. Kuka tahansa voi tilata tuotteita verkkokaupasta, mutta ilman kanta-asiakasnumeroita ei verkkokaupassa pysty myymään kanta-asiakkaille suunnattuja tuotteita alennetuilla hinnoilla. Tämän vuoksi mahdollisen kanta-asiakasnumeron kysyminen verkkokaupassa kuului osana toimeksiantoa.

Ajanpuutteestani johtuen en ehtinyt toteuttaa kaikkia toimeksiantoon kuuluneita toiveita. Kanta-asiakaspostin lähettämiseen liittyviin parannusratkaisuihin en ehtinyt perehtyä lainkaan ja asiakkaat eivät saaneet itselleen mahdollisuutta hallinnoida omia tietojaan. Toteutusvaiheessa tähtäsin lopulta vain päätavoitteiden toteuttamiseen.

Kanta-asiakasjärjestelmä tuli sellaisenaan sekä asiakkaiden että ylläpitäjän käyttöön, kuin millaiseksi sen ehdin tehdä. Ylläpitäjä sai sovelluksen, jonka avulla hän voi lisätä, muokata tai poistaa kanta-asiakkaiden tietoja tietokannasta. Tietokantaan lisättiin jo olleiden asiakkaiden tiedot ja tulevaisuudessa uusien kanta-asiakkaiden tietojen lisääminen tietokantaan on helppoa. Kanta-asiakkaat saivat itselleen kanta-asiakasnumerot ja tulevat liikkeessä liittyvät kanta-asiakkaat saavat välittömästi liittyttyään kanta-asiakasnumeron.

Toimeksiantajayrityksen vakituisen kahden työntekijän lisäksi työskentelin suunnittelun alkuvaiheessa myös ulkopuolisen mainostoimiston edustajan kanssa. Mainostoimisto oli ennen minua työskennellyt toimeksiantajayrityksen markkinoinnin parissa. Mainostoimiston edustaja auttoi minua sovelluksen luomiseen ja kehittämiseen liittyvissä ratkaisuissa sekä antoi kallisarvoisia vinkkejä, joilla pääsin suunnitteluvaiheessa järkevään alkuun. Pääasiallisen käytännön toteutuspuolen tein kuitenkin yksin.

Teoreettisen tarkastelun lähteinä tässä opinnäytetyössä olen käyttänyt lähes tuoretta tai melko ajankohtaista ja käyttökelpoista lähdemateriaalia. Internet-alalla uusi tieto päivittyy nopeampaa Internetiin kuin painettuihin teoksiin. Internet on kuitenkin pullollaan periaatteessa kenen tahansa

syöttämää tietoa, joten olen nähnyt suurimman osan kirjallisteistäni luottavampina Internet-lähteisiin verrattuna.

Sovelluksen toteutukseen liittyvissä ratkaisuissa olen tässä raportissa kuitenkin käyttänyt säännöllisesti päivittyviä ja hypermedia-alalla luotettaviksi koettuja verkkolähteitä. World Wide Web Consortium, PHP.net sekä W3Schools päivittävät verkkosivustojaan ajankohtaisiksi standardien sekä muiden uusien toteutusratkaisujen julkistamisen myötä.

2 Tausta ja tavoitteet

Yrityksen alkuperäinen kanta-asiakasjärjestelmä oli käytössä vaateliikkeen avajaisista lähtien. Jo kuukauden kuluttua tästä yrityksen edustajat kuitenkin huomasivat kanta-asiakasjärjestelmän olevan monimutkainen ja käytännössä erittäin hankala, sillä asiakkaita oli liittynyt kanta-asiakkaiksi määrällisesti huomattavasti enemmän kuin alun perin yritys oli osannut odottaakaan. Tämän vuoksi yritys tarvitsi vanhan pohjalle luotavan uuden järjestelmän, jossa olisi erilaisia toimintoja ja joiden avulla kanta-asiakastietojen käsittely ja ylläpito olisivat helpompia kuin lähtötilanteessa.

Lähtötilanne

Toimeksiannon aloitushetkellä yrityksen Internet-sivustoilla kanta-asiakkaiksi liittynyttä henkilöä kiitettiin kanta-asiakkaaksi liittymisestä liittymiskaavakkeen palautesivulla. Kanta-asiakkaiksi liittyneen henkilön tiedot lähtivät palautesivun PHP-koodin ansiosta suoraan yrityksen edustajan sähköpostiin. Muutaman päivän kuluttua liittymisestä verkossa liittynyt kanta-asiakas sai kotiin postitse ensimmäisen kanta-asiakaspostinsa sekä kanta-asiakaskorttinsa.

Vaateliikkeessä kanta-asiakkaiksi liittyvät henkilöt saivat välittömästi liittyttyään kanta-asiakaskortin. Sekä verkossa että vaateliikkeessä kanta-asiakkaiksi liittyneiden henkilöiden kanta-asiakaskorteissa luki ainoastaan asiakkaan nimi, sillä asiakkailta ei järjestelmän puuttumisesta johtuen ollut kanta-asiakasnumeroita.

Kanta-asiakkaiden tiedoille ei myöskään ollut mitään tiettyä ja varmaa tallennuspaikkaa. Kanta-asiakkaiden tiedot olivat tallessa ainoastaan joko liittymiskaavakkeissa niiden asiakkaiden osalta, jotka olivat liittyneet kanta-asiakkaiksi vaateliikkeessä tai yrityksen edustajan sähköpostissa niiden asiakkaiden osalta, jotka olivat liittyneet Internetissä. Yrityksen edustaja lisäsi vaateliikkeessä liittyneiden asiakkaiden sähköpostiosoitteet oman sähköpostinsa osoitekirjaan, sillä yrityksen kanta-asiakasposti lähetettiin kanta-asiakkaille sähköpostitse.

Yrityksen osaaminen

Yrityksen toinen omistaja on ennen yrityksen perustamista opiskellut tietojenkäsittelyalaa, joten toimeksiannon teknisen puolen toteutukseen liittyvät neuvottelut eivät olleet ylivoimaisen vaikeita. Tekninen toteutus toimeksiannossa on kuitenkin toteutukseltaan sellainen, että kuka tahansa henkilö, joka osaa käyttää tietokonetta, osaisi myös helposti käyttää sovellusta. Yrityksen edustajan pohjaosaamisesta on kuitenkin suuri hyöty mahdollisia sovelluksen koodiin liittyvien ongelmakohtien etsimisessä ja ratkaisemisessa. Sovelluksen mahdollinen päivittäminen on näin ollen mahdollista myös yrityksen edustajalle.

Tavoitteet

Kanta-asiakasjärjestelmän pääasiallisena tavoitteena oli siirtää kanta-asiakkaiden tiedot talteen tietokantaan sekä luoda lisäksi asiakkaille itselleen mahdollisuus hallinnoida omia tietojaan. Myös yrityksen edustaja tarvitsi hallintaoikeudet kanta-asiakkaiden tietoihin ja mahdollisiin muokkauksiin tai poistoihin. Eräänä esitettyä toiveena oli myös saada aikaan järjestelmä, jonka avulla kanta-asiakaspostin lähettäminen olisi helpompaa kuin toimeksiannon lähtötilanteessa. Tärkeimpänä tavoitteena oli kuitenkin luoda sekä jo olemassa oleville että tuleville kanta-asiakkaille kanta-asiakasnumerot, jotka jatkossa olisivat käytössä muun muassa yrityksen verkkokaupan yhteydessä.

Tämän opinnäytetyöraportin tarkoituksena on tarkastella kanta-asiakasjärjestelmän toteutuksen teoriaa sekä sovelluksen varsinaista toteutusprosessia. Raportin avulla kanta-asiakasjärjestelmän koodin parissa työskentelevä henkilö ymmärtää sovelluksen toteutusratkaisut sekä pystyy kehittämään koodia paremmin kuin sellainen henkilö, joka ei tätä raporttia lue.

Oman oppimiseni suhteen päätavoitteinani olivat uusien ongelmien ratkaiseminen, koodin kirjoittamisen selkeyden oppiminen, selkeät etukäteissuunnittelut sekä järkevät toteutusratkaisut. Aikataulutuksen suunnittelu sekä aikataulussa pysyminen olivat myös oleellisia opittavia asioita.

3 Toteutuksen kartoitus

Toteutustapojen vertailussa olen tietoisesti jättänyt analysoimatta HTML:n ja XHTML:n/CSS:n eroja. Perehdyn toteutustapojen vertailussa syvemmin vain käytännön teknisiin toteutusratkaisuihin. Itse näen validiuden ja standardien noudattamisen olevan nykypäivänä jo oletus verkkosivustojen luomisen yhteydessä. XHTML:stä, CSS:stä sekä transitional- ja strict-määrittelyksistä opettaa kattavasti muun muassa Jeffrey Zeldman teoksessaan *Designing With Web Standards* (2007).

3.1 Ohjelmointikielet

Palveluntarjoajalta yritykselle tilattu web-hotelli sisältää PHP5-tuen. Tämä tarkoittaa sitä, että palveluntarjoajan palvelimella toimivat PHP5-version toteutusratkaisut.

PHP on tulkattava ohjelmointikieli, eli sivulla oleva koodi suoritetaan eli ajetaan joka kerta, kun WWW-palvelin lähettää sivun selaimelle. Yleisesti käyttäjät pyytävät WWW-palvelimelta tiedostoa, jonka palvelin lähettää selaimen. PHP-koodia sisältävää tiedostoa ei kuitenkaan sellaiseen voi selaimessa näyttää. (Heinisuo & Rauta 2007, 12–13.)

PHP poikkeaa muista tunnetuimmista ohjelmointikielistä siten, että se ei käytä CGI-tekniikkaa. CGI on web-ympäristön tekniikka, jonka avulla selain voi välittää tietoja palvelimella suoritettavalle ohjelmalle (World Wide Web Consortium 2008). CGI-tekniikkaa hyödyntävissä ohjelmointikielissä HTML-merkkaukset tuotetaan ohjelmointikielen komentojen avulla (Heinisuo & Rauta 2007, 13). PHP toimii päinvastoin, sillä PHP-koodi kirjoitetaan HTML-koodin sekaan.

Kanta-asiakasjärjestelmän asiakkaille suunnatun liittymissivun toiminnot sekä verkkokauppa hallintapaneeleineen oli toimeksiannon lähtötilanteessa tehty PHP:llä, joten itselleni oli luontevinta ja kätevintä jatkaa kanta-asiakasjärjestelmän parissa samaa linjaa.

3.2 Tietojen varastointi

Esimerkiksi XML:n avulla toteutettu tekstitietokanta olisi parhain tallennusmenetelmä pienelle kanta-asiakasmäärälle. Tiedostoihin tallennetut tietokannat ovat helposti siirrettäviä eri järjestelmien välillä ja tekstimuotoa voi helposti muokata myös tekstieditorilla. Tarvittaessa tekstitietokanta on myös lähes suoraan siirrettävissä relaatiotietokantapalvelimelle. (Peltomäki 2001, 293–294.)

Tietojen etsimisessä ja käsittelyssä relaatiotietokanta on tekstitietokantaan verrattuna huomattavasti käytännöllisempi. Suurten tietomäärien käsittelyssä haku tekstitietokannasta tapahtuu erikseen kirjoitetun sovellusohjelman avulla. Mitä suuremmasta tiedostosta on kysymys, sitä enemmän tehottomuus lisääntyy, sillä ohjelman on periaatteessa käytävä läpi tietokannan jokainen alkio. Relaatiotietokantapalvelin puolestaan tuottaa nopeasti sekä haun että vastauksen relaatiotietokantaan kohdistettuun SQL-kyselyyn. (Peltomäki 2001, 293–295.)

SQL on kehitetty vastaamaan Dr. E.F.Coddin luomaa tietojen relaatiomallia (Stephens ym. 2001, 4). Relaatiotietokannassa tauluja kutsutaan relaatioiksi ja kutakin riviä tietueeksi. Jokaisella rivillä on yhtä monta tietoa eli kenttää ja jokaisella rivillä on oma perusavaimensa. Perusavain on jokaiselle riville yksilöllinen. (Lahtonen 2002, 4–7.) Perusavaimen eli järjestysnumeron käyttö kanta-asiakasnumerona on tämän ansiosta helppoa.

Tekstitietokannan mahdollisina ongelmina olisivat kanta-asiakasnumeroiden käsittely sekä kanta-asiakkaiden tietojen muokkaaminen kanta-asiakasnumeroiden perusteella. Periaatteessa nousevan järjestysnumeron käyttö kanta-asiakasnumerona olisi yhtä mahdollista toteuttaa tekstitietokannan kuin relaatiotietokannankin parissa. Koska jo lähtötilanteessa kanta-asiakkaiden lukumäärä on kuitenkin laskettavissa sadoissa henkilöissä ja asiakkaita liittyy jatkuvasti kanta-asiakkaiksi lisää, on järkevintä valita tietojen varastointimuodoksi relaatiotietokanta.

Kanta-asiakasjärjestelmän yhteydessä käytetään palveluntarjoajalta erillisenä lisäpalveluna tilattua MySQL-tietokantaa. MySQL on relaatiotietokantoja käsittelevä tietokannanhallintajärjestelmä. Käytännössä PHP lähettää SQL-kielisiä kyselyitä MySQL-tietokantapalvelimeen ja MySQL voi komennosta riippuen lähettää PHP-sovellukselle vastauksena tietokantaan tallennettua tietoa. (Heinisuo & Rauta 2007, 37–40.)

Koska MySQL-palvelimeen voidaan ottaa yhteys muun muassa PHP-tekniikan avulla, ja koska MySQL soveltuu niin pieniin kuin suuriinkin WWW-palveluihin, on sen käyttö luontevinta kanta-asiakastietojen varastoinnissa.

3.3 Tietojen suojaus

Ylläpitäjän tiedostokansio on suojattu käyttäen htaccessia. Yrityksen edustaja pääsee ylläpitopaneeliin käsiksi tunnuksen ja salasanan avulla. Juuri tällainen tietyn sivuston tietyn osan suojaaminen salasanan taakse onkin htaccessin yleisin käyttötapa. Htaccessilla voi käytännössä suojata juuri tiettyjä kansioita hakemistorakenteessa, jolloin salasanan takana ovat myös näiden kansioiden alikansiot. (Apache-opas 2008.)

Paneeli olisi mahdollista suojata myös erillisessä tietokantataulussa olevien tunnuksen ja salasanan avulla käyttäen erillistä kirjautumissivua sekä sessionhallintaominaisuutta. Tietokannasta tarkistettavat ja mahdolliselle kirjautumislomakkeelle syötettävät tunnukset ja salasanat täytyy kryptata välittömästi tietojen syöttämisen jälkeen mahdollisten väärinkäytösten estämiseksi. Tähän voisi käyttää esimerkiksi PHP:n valmista md5-tiivistefunktiota (PHP: Hypertext Preprocessor 2008a), joka on sisäänrakennettu MySQL-tietokannan ominaisuuksiin. Täten vertailu syötettyjen tietojen ja tietokannan tunnuksien välillä olisi helppoa.

Htaccessin kätevyys näkyy kuitenkin siinä, että ylläpitäjä voi työskennellä hallintapaneelissa työaikanaan ilman uudelleenkirjautumisia. Realistinen esimerkki htaccessin hyödyistä on esimerkiksi sellainen, jossa ylläpitäjä työskentelee kanta-asiakasjärjestelmän tai verkkokaupan parissa hallintapaneelissa työpaikallaan työaikanaan, ja samaan aikaan asiakas sattuu astumaan liikkeeseen. Mikäli yrityksen edustaja esimerkiksi opastaa asiakasta tiettyjen tuotteiden parissa tietyn ajan, menisi sessionhallintaominaisuuksilla luotu kirjautumisaika umpeen, ja ylläpitäjän täytyisi aloittaa työnsä myöhemmin alusta uudelleenkirjautumisen jälkeen. Htaccessilla työskenneltäessä tällaista tilannetta ei tule.

3.4 Asiakastietojen käyttö

Kanta-asiakkaiden tietojen luovutuksesta ja/tai käytöstä ei kanta-asiakkailla itsellään liene mitään tietoa. Kanta-asiakastietojen lipsuminen väärin käsiin olisi sekä huonoa tietosuojaa että erityisesti huonoa mainosta yritykselle. Kanta-asiakkaiden tietoja ei kuitenkaan anneta yrityksen ulkopuolelle. Tietoja käytetään ainoastaan yrityksen kanta-asiakkaille suunnatun postin lähettämässä eli yrityksen uusien tuotteiden mainonnassa sekä erilaisten tapahtumien mainostamisessa, joissa yritys on jollain tavoin mukana. Asiakkailla on kuitenkin oikeus kieltää omien tietojensa käytön markkinointiin (Hovi ym. 2001, 129).

Vaikka minulla on pääsy kanta-asiakastietokantaan sekä yrityksen muihin sähköisiin tietovarastoihin, en kuitenkaan käsittele kanta-asiakkaiden tietoja, ellei siihen ole erityistä tarvetta esimerkiksi mahdollisissa tietokantahäiriötapahtumissa. Yrityksen edustaja hallitsee tarpeen vaatiessa itsekin tietokannan hallinnoinnin teknisen osaamisensa ansiosta palveluntarjoajan erillisen hallintapaneelin avulla, joten suurta tarvetta minulle kanta-asiakasjärjestelmän valmistuttua ei enää ole.

3.5 Ulkoasutoteutus

Yrityksen sivustojen ulkoasun pyrkimyksenä on viestittää tietty kuva yrityksestä. Kohderyhmän eli potentiaalisten asiakkaiden olisi koettava sivustot kiinnostaviksi, sillä yrityksen verkkosivut toimivat asiakkaille sekä yrityksen näyteikkunana Internetissä että yrityksen käyntikorttina. Verkkosivujen graafisen ilmeen suunnittelu ja toteutus on tärkeää myös siksi, että verkkosivun graafinen ilme on pysyvämpi kuin sivun sisältö. Ulkoasua ei myöskään haluta tai teknisistä syistä voida muokata kovin usein. (Yritys-Suomi 2008.)

Asiakkaille suunnatun sovelluksen tekninen toimivuus on tärkeää ulkoisen tyylikkyyden ohella. Ylläpitäjälle kanta-asiakasjärjestelmän käytön ja hallinnan toimivuus millä tahansa käytettävällä Internet-selaimella on keskeisessä asemassa suunniteltaessa ulkoasutoteutukseen liittyviä toteutustapoja. Tämän vuoksi ylläpitäjälle suunnattu hallintapaneeli, jonka olen luonut jo verkkokaupan yhteydessä, on toteutettu käyttäen XHTML:ää CSS-tyylitiedostoineen transitional-määrittelyin.

Yrityksen verkkosivuilla oleva kanta-asiakassivu pysyy ulkoasultaan samanlaisena kuin toimeksiannon lähtötilanteessa huolimatta siitä, että si-

vuston koodi ei mene läpi W3C:n validaattorista
(<http://validator.w3.org/>).

4 Suunnitteluprosessi

Aloitin toimeksiannon suunnittelun yrityksen edustajan kanssa jo tammi-kuussa 2008. Varsinaisen toteutuksen suunnitteluun pääsimme vaiheittain maaliskuusta alkaen ja pidimme palavereja parhaimmillaan lähes viikoittain. Ensisijaisena tavoitteenani oli kuitenkin saada yrityksen verkkokauppa toimintaan ennen kuin voisin alkaa työstää kanta-asiakasjärjestelmän koodia. Verkkokaupan toteutus kuului osana yritykselle tehtävää työharjoitteluani.

Tiedonhaku tapahtui tarpeen ja tilanteen mukaan. Kanta-asiakasjärjestelmän koodia kuitenkin aloin varsinaisesti työstää vasta siinä vaiheessa, kun realistiset suunnitelmat olivat tarkentuneet ja varmentuneet toimeksiantajan osalta sekä verkkokauppa oli julkistamista vaille valmis.

4.1 Tekniset vaatimukset

Teknisten vaatimusten täyttäminen oli toiminnallisuuden kannalta tärkeää. Tietoturvallisuuden varmuus ja ylläpitäminen olivat ensimmäisistä suunnitteluvaiheista alkaen tärkeimmillä sijoilla teknisiä toteutusratkaisuja suunniteltaessa. Sovelluksen Internet-sivustojen teknisen toteutuksen tarkoitus oli toimia moitteettomasti ja selkeästi eri selaimilla, erikokoisilla näytöillä sekä yleisestikin eri laitteilla.

Tärkein koodiin kirjoitettava toiminto jokaiselle sellaiselle sivulle, jossa joko asiakas tai ylläpitäjä syöttää tietoa, on HTML-merkkien poiskarsiminen. HTML-merkkien karsimisen ansiosta kukaan ei pääse tahallaan tai vahingossa tyhjentämään tietokannassa olevia asiakastietoja.

Kaikista syötettävistä tiedoista karsitaan HTML-merkit pois käyttäen PHP:n `mysql_real_escape_string`-funktiota (PHP: Hypertext Preprocessor 2008b) ennen tietojen lisäämistä tietokantatauluun. Varsinaisessa sovelluksessa tämä HTML-merkkien karsiminen kuuluu oletuksena jokaiseen sellaiseen sivuun, jossa syötetään tietoja. Tämän vuoksi aiheesta ei ole erikseen mainittu sovelluksen toteutukseen liittyvässä osuudessa (luvussa 5.2 Sovelluksen toteutus).

Juoksevan järjestysnumeron käyttö kanta-asiakasnumerona oli sovelluksen selkeimpänä teknisenä vaatimuksena. Samaa numeroa ei saisi olla

monella eri asiakkaalla, ja yhdellä asiakkaalla ei saisi olla kuin yksi kanta-asiakasnumero.

4.2 Ulkoasuvaatimukset

Kanta-asiakassovelluksen ulkoasulla oli suunnitteluvaiheen alusta lähtien tärkeä merkitys niin yritykselle kuin myös tuleville asiakkaille. Sovelluksen ulkoasun tarkoitus oli olla yhtenäinen muiden yrityksen sivustojen eli kotisivujen ja verkkokaupan kanssa, mutta sen tuli kuitenkin olla oma itsenäinen kokonaisuutensa. Kanta-asiakkaiden oma mahdollisuus kirjautumiseen ja omien tietojen muokkaamiseen eivät sinällään ole asiakkaiden houkuttimia tai tuotteiden mainoskanavia, mutta silmiä miellyttävän ulkoasun tärkeyttä ei kuitenkaan pitänyt unohtaa.

Ulkoasutoteutuksen suunnittelun tärkeys oli sekä minulla että yrityksen edustajan mielessä alusta lähtien. Näimme kuitenkin tärkeämpänä sovelluksen teknisen toimivuuden, ja päätimme miettiä ulkoasun toteutusta tarpeen mukaan vasta järjestelmän toiminnallisuuksien valmistuttua.

5 Toteutusprosessi

Kanta-asiakasjärjestelmän toteutus alkoi käytännössä vasta siinä vaiheessa, kun työharjoitteluuni kuuluva verkkokauppa oli julkistamista vaille valmis. Kanta-asiakasjärjestelmän perustavoitteet olivat itselleni jo selvät, joten kävin vain suoraan töihin.

5.1 Tekniset toteutustavat

PHP oli itselleni jo ennakolta tuttu ohjelmointikieli. PHP:n perusfunktiot ovat melko yksinkertaista, mutta kattavia, joten kanta-asiakasjärjestelmän moitteeton toteuttaminen oli minulle helpointa juuri tällä ohjelmointikielillä.

Palveluntarjoajalta yritykselle tilattu web-hotelli sisältää kotisivutilan ja oman domain-nimen lisäksi sähköpostipalvelut, PHP5-tuen, FTP-yhteyden sekä kävijälaskurin. MySQL-tietokanta on tilattu palveluntarjoajalta erillisenä lisäpalveluna. MySQL-tietokantataulun luomiseen sekä tietokannassa olevien taulujen muokkaamiseen käytin palveluntarjoajan asiakkailleen tarjoamaa graafista MySQL-tietokantojen käsittelyyn tarkoitettua phpMyAdminia.

5.2 Käytännön toimintatavat

Aikataulullisesti yritys antoi minulle hyvin vapaat kädet työstää toimeksiantoa. Varsinaista käytännön toteutuksen aloittamishetkeä en edes ehtinyt kertomaan yrityksen edustajalle, sillä niihin aikoihin hioin vielä satunnaisesti verkkokaupasta erinäisiä virheitä pois. Vasta kun kanta-asiakasjärjestelmä oli lähes julkaisuvalmis, aloin käydä tiiviitä kehityskeskusteluja yrityksen edustajan kanssa kanta-asiakasjärjestelmän mahdollisista virheistä ja puutteista. Työskentelin pääasiassa kotoa käsin. Kävin ajoittain yrityksessä esittelemässä tuotoksiani tai lähetin sähköisiä kanavia myöten (esimerkiksi sähköpostitse) lyhyitä raportteja yrityksen edustajalle toimeksiantoni edistymisestä.

Tietoteknisten rajoitteiden sekä hankalien ja monimutkaisten toteutusratkaisujen selittäminen yrityksen edustajalle ei ollut vaikeaa, sillä yrityksen edustajan tietotekninen pohjaosaaminen oli hyvin lähellä omaani. Oma

aiempaa parempi lähtötasoni olisi toki taannut nykyistä paremman järjestelmän sekä selkeämmät koodiratkaisut.

Koska sovelluksen käytännön toteutuksen aloittaminen venyi aikataulullisesti pidemmälle kuin alun perin oli tarkoitus, en ehtinyt opetella uusia tietoteknisiä asioita ennen tuotoksen toteuttamisen aloittamista, enkä myöskään sen aikana, vaan järjestelmä täytyi toteuttaa niillä tiedoilla ja taidoilla, jotka minulla jo valmiiksi olivat.

5.3 Sovelluksen toteutus

Aloitin kanta-asiakasjärjestelmän toteuttamisen tietokantataulun suunnittelusta ja luomisesta. Taulun suunnittelu ei ollut vaikeaa, sillä kanta-asiakasjärjestelmän uudistamisen alkuvaiheessa yrityksen verkkosivuilla kysyttiin kanta-asiakkaaksi liittyvältä henkilöltä tietyt asiat (nimi, katuosoite, postinumero, postitoimipaikka, syntymäaika sekä sähköpostiosoite), joille kullekin tein oman kentän tietokannan kanta-asiakastauluun. Jokainen taulun tietue sai myös järjestysnumeron, joka toimii siinä kanta-asiakasnumerona. Näin estetään tilanne, jossa eri kanta-asiakkailla olisi sama kanta-asiakasnumero.

Verkkokaupan toteuttamisen puolella olin toteuttanut ylläpitäjälle yleisen hallintapaneelin, johon oli luontevinta lisätä myös kanta-asiakkaiden hallintamahdollisuudet. Tässä vaiheessa toimeksiannon toteutusta eli heinäkuun 2008 loppuvaiheilla huomasin, että verkkokaupan toteutukseen oli kulunut odotettua enemmän aikaa, joten oli selvää, että yritys ei tule ainakaan syksyn aikana saamaan toivottua aiemmasta tavasta helpompaa kanta-asiakaspostin lähetyismahdollisuutta. Tiesin kuitenkin, että kanta-asiakkaiden tietojen saaminen tietokantaan, kanta-asiakkaiden numerointi ja ylläpitäjän kanta-asiakastietojen hallinnointimahdollisuus olivat tärkeimmät toiveet, joiden toteutumiseen tähtäsin.

Luotuaani kanta-asiakkaiden tietokantataulun kopioin jo olemassa olevan liittymiskaavakkeen sekä palautesivun PHP-koodit yrityksen sivustoilta hallintapaneeliin muokkaus- ja testauskäyttöön. Muokkasin kopioitua liittymiskaavakkeen koodia aluksi vain siten, että hallintapaneelin kautta syötetyt kanta-asiakkaaksi liittymisen tiedot ohjautuivat PHP:n mail-funktion (PHP: Hypertext Preprocessor 2008c) avulla omaani, eivätkä yrityksen edustajan sähköpostiin, jonne ne ohjautuvat silloin, kun oikea kanta-asiakkaaksi haluava henkilö liittyy kanta-asiakkaaksi yrityksen verkkosivujen kautta. Oma yritykseltä saatu sähköpostiosoitteeni toimi parhaimpana sähköpostiosoitteena, jota pystyin käyttämään jatkuvasti

testatessani uudistettavan kanta-asiakasjärjestelmän toimintoa lähettää kanta-asiakkaan tiedot yrityksen edustajan sähköpostiin.

Hallintapaneeliin testauskäyttöön kopioidun liittymissivun lisäksi loin paneeliin toisen hyvin samantapaisen sivun kuin asiakkaille suunnattu liittymissivukin. Kyseessä oleva toinen sivu toimii ylläpitäjällä suunnattuna asiakkaiden tietojen lisäyssivuna. Näiden kummankin sivun PHP-koodeissa ovat toiminnot, joilla otetaan yhteys tietokantaan (Liite 1) sekä tarkistuksesta läpipääsyn jälkeen lisätään tiedot tietokantatauluun hyödyntäen SQL:n INSERT-komentoa (W3Schools Online Web Tutorials 2008a).

Tarkistajat ovat itse luotuja funktioita, joissa tarkistetaan kullakin sellaisella sivulla, joilla tietoja syötetään tietokantaan, että tietoja on syötetty tarvittava määrä (Liite 2). Asiakkaan täytyy täyttää kaikki kaavakkeessa olevat kohdat, mutta ylläpitäjän tarvitsee syöttää vain vähintään yksi kohta, joka käytännössä on satunnainen keksitty tieto tai yksittäinen merkki. Näin siksi, että täten ylläpitäjän on helpoin lisätä tietokantatauluun varattuja kanta-asiakasnumeroita niitä asiakkaita varten, jotka liittyvät kanta-asiakkaiksi vaateliikkeessä. Ylläpitäjä kirjoittaa nämä varatut kanta-asiakasnumerot kanta-asiakaskaavakkeisiin sekä niiden vastinkappaleisiin eli kanta-asiakaskortteihin.

Seuraavaksi tein lähes lisäyssivujen tapaisilla koodipohjilla hallintapaneeliin asiakkaiden tietojen muokkaus- että poistosivut. Muokkaussivulla tarkistajafunktio sallii uusien tietojen lisäämisen tietokantaan samoin kuin ylläpitäjän asiakkaiden tietojen lisäyssivu, eli silloin kun ylläpitäjä on syöttänyt kaavakkeeseen vähintään yhden kohdan. Osittain tällainen toteutusratkaisu on tarkoituksellista, mutta perusteltua (ks. lisää luvusta 8.2 Ongelmat). Käytännössä ylläpitäjä kirjoittaa muokkaussivulle kanta-asiakastiedot liikkeessä täytetystä paperisesta kanta-asiakaskaavakkeesta sekä näiden lisäksi syöttää kaavakkeen alussa olevaan kenttään paperissa olevan kanta-asiakasnumeron. Tämän kanta-asiakasnumeron perusteella ylläpitäjän syöttämät tiedot korvaavat aiemmat keksityt tiedot (kuva 1). Tiedot päivittyvät vanhoista uusiin SQL:n UPDATE-komennon ansiosta (W3Schools Online Web Tutorials 2008b).

HUOMI Täytä kaikki kohdat - tyhjiksi jätetyt kentät kumoavat mahd. olemassaolevan tiedon ko. asiakkaan kohdalla.

Syötä kantisID jonka tietoja muutat

Kuva 1 Tietojen muokkaus kanta-asiakasnumeron perusteella

Muokkauksen lisäksi myös asiakkaiden poisto tapahtuu järjestys- eli kanta-asiakasnumeron perusteella. Tiedot poistuvat tietokantataulusta SQL:n DELETE-komennolla (W3Schools Online Web Tutorials 2008c). Asiakkaiden poisto tapahtuu kuitenkin muita sivuja yksinkertaisimmin ilman tarkistajafunktioita. Hallintapaneelissa ylläpitäjä voi valita alasve-tovalikosta haluamansa järjestys- eli kanta-asiakasnumeron, jonka perusteella tietyllä kanta-asiakasnumerolla olevan kanta-asiakkaan tiedot poistuvat tietokantataulusta (kuva 2).

Kanta-asiakasnumero	
303	
Nimi	
Terttu Testiasiakas	
Syntymäaika	
1.3.1980	
Katuosoite	
Opinnäytetyönkatu 1	
Postinumero	Paikkakunta
33580	Tampere
Puhelinnumero	
050 123 4567	
Sähköpostiosoite	
terttu.testiasiakas@opiskelupaikka.com	
Poista kanta-asiakas	
303 ↓	
Poista tämän kantisnumeron mukainen kanta-asiakas	

Kuva 2 Asiakastietojen poisto kanta-asiakasnumeron perusteella

Sekä muokkaus-, poisto- että ylläpitäjän lisäyssivuilla näkyvät ylläpitäjälle jatkuvasti kaikkien kanta-asiakkaiden kaikki tietokannassa olevat kanta-asiakastiedot (kuva 3). Tällöin ylläpitäjän on yksinkertaista jatkuvasti nähdä sekä tietokantataulun sisältö että mahdolliset uudet onnistuneesti syötetyt lisätyt tai muokatut tiedot.

Kanta-asiakasnumero	
303	
Nimi	
Terttu Testiasiakas	
Syntymäaika	
1.3.1980	
Katuosoite	
Opinnäytetyönkatu 1	
Postinumero	Paikkakunta
33580	Tampere
Puhelinnumero	
050 123 4567	
Sähköpostiosoite	
terttu.testiasiakas@opiskelupaikka.com	

Kuva 3 Kaikkien kanta-asiakkaiden tiedot näkyvillä ylläpitäjälle

Tämän jälkeen aloin muokata testauskäytössä olevaa tuleville kanta-asiakkaille suunnattua liittymissivua hallintapaneelissa. Aluksi muokkasin koodin sellaiseksi, että onnistuneen kanta-asiakaan liittymisen ja tietokantaan lisäämisen jälkeen asiakas näkee palautesivulla omat kanta-asiakastietonsa. Tämän tein tosin vain testaamisen vuoksi, koska täten opin, miten kanta-asiakasnumeron sekä muut syötetyt tiedot voi saada juuri liittyneen asiakkaan kohdalla selaimen näkyviin.

Testasin toimintoa, jossa asiakas saisi omaan sähköpostiinsa kaikki syöttämänsä tiedot oman kanta-asiakasnumeronsa lisäksi. Mahdollisissa väärinkäyttötapauksissa tämä olisi kuitenkin ollut tietoturvariski, sillä kanta-asiakkaaksi liittyneen henkilön tiedot olisivat mahdollisesti lähteneet väärään eli väärän henkilön sähköpostiosoitteeseen.

Skandinaavisten merkkien kohdalla vastaan tuli suuria ongelmia. Liitetyäni PHP:n mail-funktion (PHP: Hypertext Preprocessor 2008c) yhteyteen header-tietoja, joiden avulla asiakas olisi saanut omaan sähköpostiinsa mahdolliset sähköpostiinsa tulevat skandinaaviset merkit oikeanäköisiksi, ei liittymiskaavakesivu toiminut enää lainkaan. Muokkasin header-tietoja useaan kertaan, mutta lopulta tein liittymiskaavakkeen sekä palautesivujen koodit sellaisiksi, että asiakkaan liittymisen jälkeen hänen syöttämänsä sähköpostiosoitteeseen lähetetään automaattisesti ainoastaan hänen kanta-asiakasnumerosa. En jatkanut tämän ongelman parissa taistelua tämän kauempaa, sillä tiesin, että voin toteuttaa kyseisen toiminnon yrityksen edustajan toiveiden mukaiseksi sovelluksen lopullisessa käyttöönottovaiheessa.

Liittymiskaavakkeen koodi tarkisti lähtötilanteessa sen, että asiakkaaksi liittyvä henkilö oli syöttänyt tietoa kaikkiin kysytyihin kohtiin. Pidin tämän osan koodista koskemattomana, mutta lisäsin kuitenkin koodiin tietokantayhteyden ansiosta tarkistustoiminnon. Tarkistustoiminto tarkisti ennen syötettyjen tietojen lisäämistä tietokantaan, että syötettyjä tietoja ei jo ole tietokannassa. Tämän tarkoitus oli estää saman asiakkaan liittymisen useaan kertaan sekä varmistaa se, että samalla asiakkaalla ei ole kuin yksi kanta-asiakasnumero.

Muokkasin tarkistustoiminnon vuoksi hieman palautesivun sisältöä. Mikäli kaikki annetut tiedot ovat jo kanta-asiakastaulussa sellaisenaan, ilmoitetaan palautesivulla asiakkaalle tämän olevan jo kanta-asiakas. Niin tällaisessa tapauksessa, kuin onnistuneessa kanta-asiakkaaksi liittymisessä, palautesivulla ilmoitetaan asiakkaalle, että mikäli asiakas ei ole saanut sähköpostiinsa kanta-asiakasnumeroaan, tulee hänen ilmoittaa asiasta yrityksen tiettyyn sähköpostiosoitteeseen.

Vaikka Internetissä liittyneiden kanta-asiakkaiden tiedot menevätkin jatkossa suoraan tietokantaan, lähtevät tiedot kaiken varalta aiemman tavan mukaisesti yrityksen edustajan sähköpostiosoitteeseen. Henkilön tietojen lisäksi liitin kyseisen asiakkaan kanta-asiakasnumeron lähtemään yrityksen edustajan sähköpostiin. Vaateliikkeessä kanta-asiakkaiksi liittyneiden henkilöiden tiedot säilyvät tallessa täytetyillä paperisilla kanta-asiakaskaavakkeilla papereihin käsinkirjoitetun kanta-asiakasnumeron lisäksi. Näin kaikkien kanta-asiakkaiden tiedot säilyvät tallessa, mikäli kanta-asiakkaiden tietokantataulun käsittelyssä tapahtuisi ongelmia.

Verkkokauppa ei minusta johtumattomista syistä ollut kanta-asiakasjärjestelmän toteutusvaiheessa vielääkään julkistettu, joten pystyin integroimaan kanta-asiakasjärjestelmän verkkokaupan yhteyteen jo ennen verkkokaupan virallista avaamista. Verkkokaupan luominen oli erillinen työharjoitteluun kuuluva kokonaisuutensa, mutta kanta-asiakasnumeron

kysyminen tilauksen yhteydessä, kanta-asiakasnumeron oikeellisuuden tarkistus sekä sen lähetys tilauksen yhteydessä tilaustietokantatauluun sekä yrityksen edustajan sähköpostiin kuuluivat kuitenkin osaksi kanta-asiakasjärjestelmän toimeksiantoa.

Verkkokaupan olemassa olevalle tilaussivulle lisäsin yhden täytettävän kentän lisää, johon tilaaja voi kirjoittaa mahdollisen kanta-asiakasnumeron. Numero ei kuitenkaan ole pakollinen syötettävä tieto, mutta mikäli tieto on annettu, ja mikäli tieto on oikeaa muotoa eli numeerista, näkee asiakas syöttämänsä kanta-asiakasnumeron tilausvahvistussivulla. Asiakkaan täytyy hyväksyä tilausvahvistus ennen tilauksen lähettämistä, jolloin tilaustiedot, asiakkaan omat tiedot sekä asiakkaan syöttämä kanta-asiakasnumero lähtevät sekä tilaustietokantatauluun että yrityksen edustajan sähköpostiin.

Testasin jokaista tekemääni toimintoa hallintapaneelissa useaan kertaan. Lisäsin ja poistin keksimiäni asiakastietoja sekä testasin myös tietojen muokkaamista. Mitä enemmän lisäsin asiakkaita, sitä suuremmaksi sain kasvamaan sen kanta-asiakasnumeron, joka järjestelmän käyttöönotto-vaiheessa tulisi olemaan ensimmäinen käytettävissä oleva kanta-asiakasnumero oikealle asiakkaalle. Testausta varten syötetyt keksityt kanta-asiakastiedot poistettiin tietokantataulusta ennen järjestelmän käyttöönottoa.

Testasin useaan kertaan myös kanta-asiakkaiden tietojen tarkistamista tietokannasta asiakkaan liittymisvaiheessa sekä kanta-asiakasnumeron lähettämistoiminnon kanta-asiakkaaksi liittyneelle asiakkaalle.

5.4 Sovelluksen käyttöönotto

Alustavassa käyttöönottovaiheessa yrityksen edustaja huomasi itse unohtaneensa kysyä Internetissä liittyviltä kanta-asiakkailta asiakkaan puhelinnumeroa. Internet-kaavakkeeseen tämän kohdan lisääminen kävi helpposti, mutta lisäsin sen myös kaikkiin kohtiin, joissa joko kysyttiin tietoja (hallintapaneelissa lisäys- ja muokkaussivu sekä asiakkaille suunnattu liittymissivu) tai tulostettiin tietokannassa olevia tietoja (hallintapaneelin lisäys-, muokkaus ja poistosivut). Tärkeimpänä muutoksena lisäsin puhelinnumerosarakkeen kanta-asiakastietokantatauluun (kuva 4).

Sarake	Tyyppi	Aakkosjärjestys	Attribuutit	Tyhjä	Oletusarvo	Lisätiedot
idkantisro	int(10)		UNSIGNED	Ei		auto_increment
nimi	varchar(70)	latin1_swedish_ci		Kyllä	NULL	
katu	varchar(50)	latin1_swedish_ci		Kyllä	NULL	
postinumero	varchar(20)	latin1_swedish_ci		Kyllä	NULL	
paikkakunta	varchar(30)	latin1_swedish_ci		Kyllä	NULL	
puhelinnumero	varchar(30)	latin1_swedish_ci		Kyllä	NULL	
email	varchar(70)	latin1_swedish_ci		Kyllä	NULL	
syntaika	varchar(15)	latin1_swedish_ci		Kyllä	NULL	

Kuva 4 Kanta-asiakastietokantataulu

Lopullisessa sovelluksen käyttöönottovaiheessa havaitsin suuria ongelmia. Kopioin hallintapaneelissa toimineet ja moitteettomasti testauskäytössä toimineet koodit asiakkaille suunnatulle yrityksen Internet-sivujen kanta-asiakkaiden liittymis- sekä palautesivuille. Tunteamattomasta syystä syötettyjen tietojen tarkistus tietokantataulusta ei kuitenkaan toiminut, joten otin kyseisen kohdan koodista kokonaan pois ja muokkasin myös asiakkaille suunnatun palautesivun sisältöä. Tämän tarkistustoiminnon puuttuminen tosin mahdollistaa asiakkaan tietojen lisäämisen useaan kertaan niin tietokantaan kuin jokaisella kerralla niiden lähettämisen yrityksen edustajan sähköpostiin. Yrityksen edustaja voi kuitenkin poistaa sähköpostistaan sekä hallintapaneelin kautta tietokantataulusta ylimääräiset ja turhat liittymistiedot.

Edellä mainitun ongelman vuoksi sekä toimeksiantajan pyynnöstä otin pois käytöstä myös toiminnon, jossa kanta-asiakkaaksi liittynyt henkilö olisi saman tien liittyttyään saanut sähköpostiinsa kanta-asiakasnumeron. Olisi nimittäin ollut mahdollista, että asiakas olisi saanut sähköpostiinsa useita turhia sähköposteja, joissa olisi lukenut vain numerosarja. Toisin sanoen asiakkaan ei ole käytännössä mahdollista liittyä kanta-asiakkaaksi saadakseen saman tien kanta-asiakasalennuksia verkkokaupasta, sillä Internetissä kanta-asiakkaaksi liittynyt henkilö saa jatkossa kanta-asiakasnumeron sähköpostitse vasta myöhemmin, eli silloin kun yrityksen edustaja sen hänelle lähettää. Tämä tapahtuu joko ensimmäisen sähköisen kanta-asiakaspostin yhteydessä tai silloin kun ylläpitäjä lähettää kanta-asiakaskortin kanta-asiakkaalle postitse.

6 Ratkaisukeinot ongelmatilanteissa

Olen tarkoituksellisesti jättänyt tarkastelematta mahdollisia palvelinongelmia. Käsittelen tässä ainoastaan omaan toteutukseeni liittyviä mahdollisia tulevaisuudessa esiintyviä häiriöitä.

6.1 Päivitykset

Kanta-asiakassovelluksen päivitys tai koodin muokkaus olisi tekniseltä toteutukseltaan mahdollista yrityksen edustajalle hänen tietoteknisen osaamisensa ansiosta. Yrityksen omien toimien vuoksi hänellä tuskin kuitenkaan on aikaa päivittämiseen, joten jatkopäivitystä tai sovelluksen luomista uudelleenlaiseksi olisi pyydettävä ulkopuoliselta taholta.

Olen itse lupautunut pienten ja helposti ratkaistavien ongelmien korjajaksi vuoden 2008 aikana. Tulevaisuuden työharjoittelijoista yrityksellä ei ole visiota, mutta tarpeen mukaan sellaisenkin yritys voi ottaa tietotekniselle puolelle luomaan uusia teknisiä asioita ja/tai kehittämään luomiani sovelluksia.

Päivityksen osaisi teknisesti helpoimmin hoitaa mainostoimisto, jonka edustajalta itekin sain apua toimeksiannon aloitusvaiheessa. Kirjoittamani koodin päivittäminen merkitsisi kuitenkin omiin toteutustapoihini perehtymistä, minkä vuoksi mainostoimiston edustajalle olisi helpointa luoda kokonaan uusi sovellus selkeämmillä toteutusratkaisuilla, kuin mikä itse olen kanta-asiakasjärjestelmän luonut.

6.2 Tekniset ongelmat

Eräs ennakoitava yrityksestä riippumaton ongelma on yrityksestä lähetettyjen sähköpostien automaattinen ohjautuminen tiettyjen tahojen sähköpostipalvelimien roskapostisuodattimiin. Tällaiselle ongelmalle yritys tosin ei voi mitään ja asiasta on tiedote yrityksen verkkosivuilla kanta-asiakaskaavakkeen yhteydessä sekä verkkokaupan toimitusehdoissa.

Toinen sähköposteihin liittyvä ongelma on skandinaavisten merkkien käytön ongelmallisuus. Osasin itse ennakoida tämän ongelman ennen paneutumistani kanta-asiakasjärjestelmän uudistamistyöhön. Omalle koh-

dalleni tätä ongelmaa suuressa laajuudessa ei kuitenkaan tullut, sillä en olisi ehtinyt tehdä kanta-asiakaspostin lähettämissovellusta toivotussa aikataulussa. Jatkokehitystä ajatellen sähköpostien skandinaavisten merkkien käyttöön liittyvä ongelmallisuus on varmasti ratkaistavissa joillakin keinoilla. Yrityksen omat sähköpostiosoitteet ottavat kuitenkin täysin moitteettomasti vastaan yrityksen omilta sivustoilta lähetettyjä tietoja, joten on oletettavissa, että ongelma skandinaavisten merkkien toiminnasta ulkopuolisten sähköpostiosoitteissa on korjattavissa.

Vaikka kaikkien kanta-asiakkaiden tiedot säilyvät tallessa joko yrityksen edustajan sähköpostissa tai paperisilla kanta-asiakaskaavakkeilla, mikäli kanta-asiakkaiden tietokantataulu esimerkiksi jostain syystä tyhjentäisi, eivät ennen uuden kanta-asiakasjärjestelmän käyttöönottoa liittyneiden henkilöiden kanta-asiakasnumerot ole missään tallessa. Nämä täytyisi kirjoittaa esimerkiksi erikseen käsin muistiin tai ottaa tietokantataulusta kopio. Ensimmäinen vaihtoehto on lähes mahdoton, koska kanta-asiakkaiksi liittyneitä henkilöitä oli noin kaksisataa ennen uuden järjestelmän käyttöönottoa.

6.3 Salasanat

Ylläpitäjän hallintapaneeli on suojattu htaccessilla, johon tunnukset ja salasanat ovat vain yrityksen edustajalla sekä minulla. Mikäli kuitenkin olisi syytä epäillä tunnusten väärinkäyttöä ja/tai tunnusten joutumista ulkopuolisten käsiin voisin helposti vaihtaa uudet tunnukset ja salasanat. Tunnusten vaihtaminen olisi toki mahdollista kenelle tahansa muullekin henkilölle, joka kyseessä olevan toimen osaa, mutta koska yrityksen web-hotelli on virtuaalisessa palvelinympäristössä, vaativat oletetut absoluuttiset polut muokkaamista. Tämän muokkaamisen oppii tosin kuka tahansa, joka asian haluaa oppia, mutta tarpeen vaatiessa pystyisin muita nopeammin vaihtamaan tunnukset, salasanat ja polut.

7 Lopputulos

Ensimmäisten suunnittelujen ja toteutuksen aloituksen välissä työstin yritykselle verkkokaupan, johon oli kulunut odotettua enemmän aikaa. Ajan loppumisen vuoksi jouduin lopullisesta toimeksiannostani jättämään pois tiettyjä toivottuja asioita. Selkeiden suunnitelmien ansiosta pystyin kuitenkin aloittamaan päämäärätietoisesti kanta-asiakasjärjestelmän tärkeimmät uudistamistoimenpiteet.

Verkkokauppaa ei minusta riippumattomista syistä voitu julkistaa odotetussa aikataulussa, joten kanta-asiakasjärjestelmän integrointi verkkokaupan yhteyteen oli mahdollista jo ennen verkkokaupan julkistamista. Koodiin lisättävät asiat tapahtuivat yllättävän helposti, sillä olin ottanut verkkokaupan toteutusvaiheessa odotettua paremmin huomioon tulevan kanta-asiakasjärjestelmän.


7.1 Onnistumiset

Pääasialliset tarpeet ja toiveet, joita yritys esitti toimeksiantoon, tulivat tehtyä ajoissa. Asiakkaat saivat kanta-asiakasnumerot, joita kysytään myös verkkokaupan tilauksen yhteydessä. Asiakkaiden tiedot ovat tietokannassa, johon yrityksen edustajalla on pääsy sekä hallinnointimahdollisuudet. Eri asiakkaiden ei myöskään ole mahdollista saada samaa kanta-asiakasnumeroa kuin jollakulla toisella asiakkaalla on. Internetissä liittyvien asiakkaiden tiedot menevät suoraan tietokantaan ja liikkeessä kanta-asiakkaiksi liittyvät henkilöt saavat välittömästi liityttyään kanta-asiakas-korttinsa, jossa lukee valmiina asiakkaan kanta-asiakasnumero.

Niin asiakkaan kuin ylläpitäjän minkä tahansa tietojen syöttövaiheessa koodi karsii syötetyistä tiedoista kaikki mahdolliset HTML-merkit pois. Tämä on oleellinen toiminto siksi, että kukaan ei pääse tahallaan tai vahingossa esimerkiksi tyhjentämään tietokantaa, mikä on mahdollista tällaisissa tilanteissa, jossa syötetyt tiedot lisätään automaattisesti tietokantaan.

Asiakkaille yrityksen verkkosivuilla näkyvä sovellus (kuva 5) on puhelinnumero-kenttää lukuun ottamatta samanlainen kuin aloitusvaiheessa-kin. Kanta-asiakasjärjestelmän suurimmat sekä tekniset että ulkoasulliset uudistukset tapahtuivat koodin lisäksi ylläpitäjälle suunnatulla hallintapaneelipuolella, joka on suojattu htaccessilla. Myös yrityksen verkko-

kaupassa tilauksen yhteydessä kysytään mahdollista kanta-asiakasnumeroa.



Nimi:

Katuosoite:

Postinumero:

Paikkakunta:

Puhelinnumero:

Sähköpostiosoite:

Syntymäaika:

Kuva 5 Sovelluksen ulkoasu asiakkaille

7.2 Ongelmat

Kaikista syötetyistä tiedoista, niin asiakkaan täyttämästä liittymiskaavakkeesta, kuin myös ylläpitäjän syöttämistä tiedoista hallintapaneelissa, karsitaan kaikki mahdolliset HTML-merkit pois. Koodi ei kuitenkaan tarkista tietojen muotoa muualla kuin kanta-asiakasnumeron muodon verkkokaupan tilaussivulla. Asiakkaan syöttämät liittymistiedot voivat siis olla mitä vain ja esimerkiksi sähköpostiosoite voi olla pelkästään yksittäinen merkki. Tällaisten yksinkertaisten virheiden jättäminen koodiin kertoo todellisesti ajanpuutteesta sekä kiireestäni saada kanta-asiakasjärjestelmä vain jotenkuten toimimaan. Toisaalta kanta-asiakkaiden toimivat ja oikeelliset sähköpostiosoitteet ovat edellytys kanta-asiakaspostin vastaanottamiselle sekä verkkokaupan kohdalla yksi noudatettavista ja hyväksyttävistä toimitusehdoista.

Vaikka verkkokaupassa tilauksen yhteydessä kanta-asiakasmuodon oikeellisuus tarkistetaankin, ei syötetyn kanta-asiakasnumeron olemassaoloa tarkisteta tietokannasta. Tämä tarkoittaa sitä, että mikäli syötetty

numero on oikeaa muotoa eli numeerista, lähtee asiakkaan tuotetilauksen yhteydessä syötetty mahdollinen keksittykin kanta-asiakasnumero yrityksen edustajalle. Ylläpitäjän täytyy tarkistaa tilauksesta käsin sekä se, onko kyseistä kanta-asiakasnumeroa yleensä olemassa, mutta myös se, onko kanta-asiakasnumero yhteydessä kyseiseen asiakkaaseen.

Ylläpitäjälle tällaiset toimenpiteet tuovat vain ylimääräistä työtä. Hänelle lähetettävässä verkkokauppatilauksessa lukevat kuitenkin sekä kaikkien yksittäisten tuotteiden, kuin myös kokonaistilauksen hinnat postikuluneen yleisesti sekä kanta-asiakkaille huolimatta siitä, onko tilattavissa tuotteissa lainkaan kanta-asiakasalennuksilla myytäviä tuotteita. On kuitenkin mahdollista, että asiakas haluaisi tilata tuotteita kaverinsa kanta-asiakasnumerolla alennuksien toivossa, mutta tällaisten tapausten selvittäminen ja tarkistaminen on tapauskohtaista ja hankalaa.

Hallintapaneelin kanta-asiakkaiden tietojen muokkaussivu on osittain ongelmallinen. Muokkaussivulla täytettävästä kaavakkeesta ylläpitäjän on pakollista täyttää vain vähintään yhden sarakkeen tiedot. Lisäsin ylläpitäjälle kyseisen muokkaussivun alkuun muistutuksen ongelmasta, mutta jätin koodin tarkoituksella muokkaamatta, vaikka huomasin ongelman jo testausvaiheessa. Ongelma ilmenee silloin, jos tietyltä asiakkaalta on tietokantaan annettu kaikki pyydytyt tiedot. Mikäli ylläpitäjä muokkaa jotain yksittäistä tietoa, mutta jättääkin muut muokkauskaavakkeen kohdat tyhjäksi, kumoavat tyhjät tiedot muissa sarakkeissa mahdolliset jo olemassa olevat tiedot. Muokkaussivun koodin korjaaminen sellaiseksi, että kaikki kysytyt kohdat olisivat pakollisia annettavia tietoja, ei olisi vaikeaa. Liikkeessä kanta-asiakkaaksi liittyvä henkilö ei välttämättä kuitenkaan täytä paperikaavakkeeseen kaikkia kysytyjä kohtia, jolloin ylläpitäjän olisi täytettävä muokkaussivun kaavakkeeseen keksittyjä tietoja tai ylimääräisiä turhia merkkejä, jotta asiakkaan tiedot pystyisi tallentamaan tietokantaan.

7.3 Saavutetut tavoitteet

Toimeksiannon tärkeimmät tavoitteet eli kanta-asiakkaiden numeroiminen sekä kanta-asiakastietokanta toteutuivat onnistuneesti. Internetissä kanta-asiakkaiksi liittyvien asiakkaiden tiedot lisätään automaattisesti kanta-asiakastietokantatauluun. Kanta-asiakasnumerona toimii tietokantataulussa nouseva järjestysnumero, minkä ansiosta eri asiakkailta ei voi järjestelmävirheen takia olla samaa kanta-asiakasnumeroa.

Yrityksessä ylläpitäjällä on hallinnointimahdollisuus kanta-asiakkaiden tietoihin. Hallintapaneelin kautta tietokantaan on mahdollista lisätä varattuja asiakastietoja, joista saatava järjestysnumero kirjoitetaan valmiiksi liittymiskaavakkeeseen sekä sen vastakappaleeseen eli kanta-asiakas-korttiin. Kun uusi asiakas liittyy kanta-asiakkaaksi vaateliikkeessä, saa hän välittömästi kanta-asiakaskortin, jossa on valmiina hänen kanta-asiakasnumerosa. Ylläpitäjä lisää tiedot liittymiskaavakkeesta kanta-asiakastietoihin järjestys- eli kanta-asiakasnumeron perusteella vaihtamalla niin kutsutut tyhjät tiedot uusiin.

Ylläpitäjä lisää tietokantaan käsin siis ainoastaan niiden asiakkaiden tiedot, jotka ovat liittyneet kanta-asiakkaiksi vaateliikkeessä sekä kaikki ennen järjestelmän käyttöönottoa kanta-asiakkaiksi liittyneiden tiedot (Liite 3).

Kanta-asiakasjärjestelmän integroiminen yrityksen verkkokaupan yhteyteen kävi vaivattomasti. Verkkokaupassa tilauksen yhteydessä asiakkaalta kysytään mahdollista kanta-asiakasnumeroa joka lähetetään muiden tilaustietojen yhteydessä yrityksen edustajalle (kuva 6).



The image shows a vertical list of input fields for an online order form. The fields are labeled in Finnish: Etunimi (First name), Sukunimi (Last name), Katuosoite (Street address), Postinumero (Postal code), Postitoimipaikka (Post office location), Puhelinnumero (Phone number), Sähköposti (Email), Kanta-asiakasnumero (jos on) (Loyalty customer number (if any)), and Maksu (Payment). Each label is positioned above a corresponding white rectangular input box with a thin black border. The entire form is set against a light gray background.

Kuva 6 Yrityksen verkkokaupan tilauskaavake

Olen opettanut yrityksen edustajalle hallinnointipaneelin käytön sekä ominaisuudet. Näin ollen ylläpitäjä on päässyt työskentelemään itsenäisesti kanta-asiakasjärjestelmän parissa, ja oma osuuteni järjestelmän toteutuksen ja käyttöönoton osalta on toivotusti ohi.

Oman oppimiseni suhteen opin ratkaisemaan uusia vastaantulevia ongelmia sekä koodinkirjoittamisen selkeyttä. Selkeä etukäteissuunnittelu oli avainasemassa alkaessani toteuttaa toimeksiantoa, joten käsitykseni selkeiden suunnitelmien tärkeydestä ennen toteutuksen aloittamista vahvistui entisestään. Aikataulutuksen suunnittelu lähti vinoutumaan heti aikaisessa vaiheessa, sillä verkkokaupan toteutuksen ja julkistamisen kanssa tapahtui suuria viivästyksiä. Suhteutettuna verkkokaupan julkistamisaikatauluun pysyin kuitenkin hyvässä aikataulussa.

Ymmärsin lopulta myös sen, että koodin kirjoittamista ja eri asioiden onnistunutta toteutusta ei opi koulussa oppitunneilla yhtäkkisessä hetkessä, vaan asiaan täytyy perehtyä ja keskittyä yksin sekä kokeilla erilaisia toteutustapoja ja löytää oikeat ratkaisut yritysten ja erehdysten kautta. Asiaan täytyy todellakin olla suuri kiinnostus, jotta keskittyminen ja mielenkiinto toteutettavaa asiaa kohtaan säilyvät.

Tämän opinnäytetyöraportin tarkoitus tarkastella kanta-asiakasjärjestelmän toteutuksen teoriaa sekä sovelluksen varsinaista toteutusprosessia onnistui odotettua huonommin. Opinnäytetyöraportin avulla lukija toki ymmärtää, miten kanta-asiakasjärjestelmä on toteutettu, mutta kanta-asiakasjärjestelmän kehittämiskäytäntöihin lähtevä henkilö pystyy muokkaamaan kanta-asiakassovelluksen koodia nykyistä paremmaksi, vaikka ei tätä raporttia olisi lukenutkaan.

Pidimme järjestelmän käyttöönoton jälkeen palaverin yrityksessä, jolloin keskustelin ylläpitäjän kanssa sovellukseen jääneiden virheiden korjauksesta. Teknisesti hän osaa itse korjata räikeimmät virheet koodiin, kuten esimerkiksi lisätä sähköpostiosoitteiden oikeellisuuden tarkistustoiminnon. Samassa yhteydessä asensin yrityksen tietokoneelle tiedostonsiirto-ohjelman sekä opetin ylläpitäjälle sen käytön. Tämän ansiosta yrityksen edustaja korjaa oman aikataulunsa mukaisesti kanta-asiakasjärjestelmään jääneet selkeät virheet, ja itse toimin tarvittaessa sovelluksen laajemman koodipuolen korjaajana.

Järjestelmästä puuttumaan jääneet osiot, kuten kanta-asiakaspostin helppo lähettäminen, siirtyvät tehtäväksi tulevaisuuteen mahdollisille työharjoittelijoille. Yritys ei kuitenkaan koe kanta-asiakaspostin helppoa lähettämistä enää akuutiksi ongelmaksi.

8 Johtopäätökset

Sovelluksen toiveiden, tavoitteiden sekä teknisten ratkaisujen selkeä etukäteissuunnittelu olivat tärkeimmät asiat ennen sovelluksen toteutuksen aloittamista. Etukäteissuunnittelun ansiosta tiesin, mitä minulta odotetaan, kenelle ja minkä vuoksi teen kanta-asiakasjärjestelmää, millä aikataululla järjestelmän odotetaan olevan valmis, sekä millaisia teknisiä toimintoja järjestelmään kuuluu luoda joko tyhjästä tai jonkin vanhan pohjalta.

Palveluntarjoajan rajoitteet olisi ollut hyvä ottaa etukäteen selville. Vaikka tiesinkin skandinaavisten merkkien ongelmallisuuden, en kuitenkaan tiedostanut sitä tarpeeksi ajoissa. Toimeksiannon eräänä tavoitteena ollut kanta-asiakaspostin aiempaa helpompi lähettäminen jäi kuitenkin tekemättä johtuen kuitenkin pääasiassa ajanpuutteestani. Myös ajanpuutteesta johtuen kanta-asiakkailta itsellään ei edelleenkään ole omaa mahdollisuutta muokata omia tietojaan.

Kanta-asiakasjärjestelmä tuli kuitenkin sellaisenaan sekä asiakkaiden että yrityksen käyttöön, kuin millaiseksi sen sain tehtyä, vaikka en ollut pystynyt aloittamaan sovelluksen toteutuspuolta suunnitellun aikataulun mukaisesti. Opin, että olisi ollut kuitenkin järkevää suunnitella toteutuspuoli aiempaakin selkeämmin kuin ihmetellä vasta lähes julkaisuvaiheessa, miksi jokin asia ei toimikaan kuten pitäisi.

Eräänä oleellisena suunnitteluvirheenä huomasin tietokantataulun, jossa olen yhdistänyt sekä etunimi- että sukunimikentät alkuperäisen kanta-asiakaskaavakkeen mukaisesti yhdeksi tietokantataulun soluksi. Näiden kenttien eriyttäminen olisi ollut nykyistä parempi vaihtoehto, ja mikäli kanta-asiakasnumeron oikeellisuuden tarkistaminen tietokannasta olisi toiminut, olisi ollut helppoa pystyä tarkistamaan myös kanta-asiakasnumeron yhdistäminen tiettyyn henkilöön esimerkiksi etu- ja sukunimen perusteella.

Raportoinnin tärkeys jokaisen toteutetun vaiheen jälkeen olisi ollut erittäin oleellista. Tiesin jatkuvan raportoinnin jokaisen yksittäisen työvaiheen jälkeen tärkeäksi tulevaisuuden työelämääkin ajatellen, mutta käytännön koodaaminen veti kuitenkin liikaa puoleensa. Lopullinen muistiinpanojeni tekeminen varsinaisesta toteutuksestani alkoi vasta kanta-asiakasjärjestelmän valmistuttua. Siinä vaiheessa oli vaikeaa enää muistella, mitä olin tehnyt milloinkin ja miksi. Näin pienen projektin ollessa kyseessä raportointi onnistuneesti jälkikäteen kuitenkin onnistui yllättävän kiitettävästi.

Jatkuva testaus eri työvaiheiden aikana oli eräs parhaimmista oivalluksista. Olkoonkin, että kaikki testialustalla toimineet asiat eivät enää julkaisuvaiheessa toimineetkaan, joten ne poistettiin lopullisesta sovelluksesta, mutta oman oppimisen sekä sovelluksen kehittämisen kannalta jatkuva testaus työstämisvaiheessa oli edellytys toimivalle kanta-asiakasjärjestelmälle.

Lähteet

Painetut

Heinisuo, Rami & Rauta, Ilkka 2007. PHP ja MySQL:tietokantapohjaiset verkkopalvelut. Helsinki: Talentum.

Hovi, Ari & Koistinen, Heikki & Ylinen, Jari 2001. Tietovarastot liiketoiminnan tukena. Helsinki: Satku.

Lahtonen, Tommi 2002. SQL. Jyväskylä: Docendo.

Stephens, Ryan K. & Plew, Ronald R. & Morgan, Bryan & Perkins, Jeff (kääntäjä: Arola, Jussi) 2001. SQL tietokantaohjelmointi. Helsinki: Edita.

Peltomäki, Juha 2001. Perl. Jyväskylä: Docendo.

Internet

Apache-opas 2008. Htaccess. [online] [viitattu 11.9.2008]
http://www.2kmediat.com/apache/apache_konfiguraatio2.asp

PHP: Hypertext Preprocessor 2008a. Md5-tiivistefunktio. [online] [viitattu 10.9.2008]
<http://fi2.php.net/manual/en/function.md5.php>

PHP: Hypertext Preprocessor 2008b. Mysql-real-escape-string. [online] [viitattu 25.9.2008]
<http://fi.php.net/manual/en/function.mysql-real-escape-string.php>

PHP: Hypertext Preprocessor 2008c. Mail-funktio. [online] [viitattu 26.8.2008]
<http://fi.php.net/manual/en/function.mail.php>

World Wide Web Consortium 2008. CGI. [online] [viitattu 17.9.2008]
<http://www.w3.org/CGI/>

W3Schools Online Web Tutorials 2008a. INSERT INTO. [online] [viitattu 25.9.2008]
http://www.w3schools.com/SQL/sql_insert.asp

W3Schools Online Web Tutorials 2008b. UPDATE. [online] [viitattu 25.9.2008]
http://www.w3schools.com/SQL/sql_update.asp

W3Schools Online Web Tutorials 2008c. DELETE. [online] [viitattu 25.9.2008]
http://www.w3schools.com/SQL/sql_delete.asp

Yritys-Suomi 2008. Www-sivujen graafisen ulkoasun suunnittelu. [online] [viitattu 15.9.2008]
<http://www.yrityssuomi.fi/default.aspx?nodeid=16217>

Liitteet

Liite 1 Yhteyden ottaminen tietokantaan

```
<?php
```

```
//PHP-koodissa yhteys tietokantaan otetaan seuraavalla tavalla
```

```
$db_user = "käyttäjätunnus";  
$db_passwd = "salasana";  
$db_server = "localhost";  
$db_db = "tietokannan_nimi";  
$tbl_customs = "tietokantataulun_nimi";
```

```
//avataan yhteys tietokantaan
```

```
$lnk = mysql_connect($db_server,$db_user,$db_passwd) or die ("Yhteys  
tietokantaan epäonnistui");
```

```
mysql_select_db($db_db, $lnk) or die ("Tietokannan valitseminen epäon-  
nistui");
```

```
// yhteyden muodostuttua tietokantaan tapahtuvat koodiin seuraavaksi  
kirjoitetut toiminnot
```

Liite 2 Tarkistajafunktio

```
<?php
```

```
/* Tarkistaja on itse tehty funktio, jossa tarkistetaan, onko kaikki vaaditut tiedot annettu. Seuraavassa on tarkistajafunktion koodi sovelluksen kohdasta, missä vaaditaan kaikkien tietojen syöttämistä. */
```

```
function tarkistaSyöte ($nimi, $katu, $postinumero, $paikkakunta, $email, $syntaika, $puhelinumero) {  
    if ($nimi!="" && $katu!="" && $postinumero!="" && $paikkakunta!="" && $email!="" && $syntaika!="" && $puhelinumero!="") {  
        return true;  
    }  
    else return false;  
}
```

```
/* If-else -rakenteella tarkistetaan syöte, ja mikäli kaikki tiedot on annettu, suoritetaan iffin sisällä olevat asiat. */
```

```
if (tarkistaSyöte($nimi, $katu, $postinumero, $paikkakunta, $email, $syntaika, $puhelinumero)) {
```

```
/* Otetaan yhteys tietokantaan ja suoritetaan koodissa olevat asiat, esim. lisätään syötetyt tiedot tietokantaan. */
```

```
}
```

```
/* Mikäli kaikkia tietoja ei ole annettu, palauttaa tarkistajafunktio käyttäjän else-rakenteen sisään. */
```

```
else {  
    echo "Kaavakkeesta puuttuu tietoja, ole hyvä ja palaa takaisin."  
}
```

```
?>
```

Liite 3 Kanta-asiakasjärjestelmän kokonaistoiminta

