

Toni Määttä

Verkkokaupan toteutus ja kehitys

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikka

Insinöörityö

28.4.2015

Tekijä(t) Otsikko	Toni Määttä Verkkokaupan toteutus ja kehitys
Sivumäärä Aika	30 sivua 28.4.2015
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	Ohjelmistotekniikka
Ohjaaja(t)	Yliopettaja Erja Nikunen Projektinjohtaja Reino Ederveen
Insinööritöiden tavoitteena oli kehittää verkkokauppa Fidelix Oy-nimiselle yritykselle. Verkkokauppa oli aluksi tarkoitus tulla käyttöön Fidelixin alihankkijoille ja yhteistyökumppaneille ja sen jälkeen ulkomaanvientiin. Verkkokaupasta toivottiin mahdollisimman yksinkertainen ja helppokäyttöinen tuotteen hakemisesta tilauksen tekemiseen. Verkkokauppa toteutettiin aiemman testauksessa olleen verkkokaupan pohjalta. Pohjana ollut verkkokauppa ei koskaan otettu käyttöön sen keskeneräisyyden takia. Verkkokaupalla on useita eri "tasoisia" asiakkaita, joilla on eri hinnat tuotteille, ja verkkokaupan tärkeimpänä ominaisuutena oli keksiä keino antaa oikealle asiakkaalle oikea tuotteen hinta. Verkkokaupalle myös kehitettiin hallintasivusto, jonka tuli olla mahdollisimman helppokäyttöinen, sillä sivuston hallinnoijat eivät välttämättä ole tietotekniikan ammattilaisia. Verkkokauppa otettiin yhteistyökumppaneilla testaukseen elokuussa. Verkkokauppa kehitetään jatkuvasti aina uusien tarpeiden mukaan.	
Avainsanat	Verkkokauppa, verkkoliiketoiminta

Author(s) Title	Toni Määttä Implementation and improvement of online store
Number of Pages Date	30 pages 28 April 2015
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Software engineering
Instructor(s)	Erja Nikunen, Principal Lecturer Reino Ederveen, Project Manager
The aim of this thesis was to implement a online store to a company called Fidelix Ltd. The online store was first to be used by Fidelix's partners and later by export resellers. The goal was to make the site as simple and easy for the customer to use as possible from the product selection to finalizing the order. The online store was built on an old online store project, which was never finished due to limited resources. The store has multiple different customer "levels", which are used to determine a correct price for a specific customer. The most important part of the project was to figure a way to do this in an efficient way. A administrative-page was also created for the store. It was also to be as simple as possible, because the administrators are likely not computer professionals or do not have understanding of the web technology. The online store was introduced to Fidelix's partners in August for testing. It is being developed continuously to meet the new needs, if any arise.	
Keywords	web store, online store

Sisällys

1	Johdanto	1
2	Projektin lähtökohdat	2
3	Projektin toteutus	3
3.1	Vaatimusmäärittely	3
3.2	Projektisuunnitelma	4
4	Projektin raportointi	5
5	Projektin toteutuksen työvälineet	6
5.1	PHP-ohjelmointikieli	6
5.2	Relaatiotietokanta	8
5.3	Ajax	8
5.4	Ohjelmointi-tiedostoeditori	10
5.5	Muuta	10
6	Järjestelmän toteutus	12
6.1	Rakenne	12
6.2	Kauppa-sivuston toiminnot	14
6.3	Admin-sivusto	22
6.4	Order-sivusto	24
6.5	MakeOrder-sivusto	25
7	Jatkokehitys	26
8	Yhteenveto	27
	Lähteet	28

Lyhenteet

PHP	(PHP: Hypertext Preprocessor) Ohjelmointikieli, lähinnä dynaamisen websisällön tuottamiseen.
SQL	(Structured Query Language) Kyselykieli relaatiotietokantakyselyjä, muutoksia ja lisäyksiä varten.
MySQL	Relaatiotietokantaohjelmisto jolla tietokantoja ylläpidetään ja hallinnoidaan.
Ajax	(Asynchronous JavaScript And XML) Javascriptillä toteutettuja tekniikoita joiden avulla web-sovelluksia voidaan ajaa taustalla (reaaliajassa).
Pear	(PHP Extension and Application Repository) PHP:lle luotu julkaisujärjestelmä joka pitää sisällään uusiokäytettäviä funktioita.

1 Johdanto

Fidelix on suomalainen rakennusautomaatio- ja turvajärjestelmiä kehittävä ja myyvä yritys. Päätuotteina ovat keskusyksiköt ja ala-asemat, joita voidaan käyttää erikokoisissa järjestelmissä. Esimerkiksi Musiikkitalo ja Stockmannin tavaratalo Helsingissä käyttävät Fidelixin laitteita.

Tilausprosessi ja niiden käsittely oli vanhanaikainen ja manuaalinen, jossa asiakkaiden tilaukset otettiin vastaan sähköpostilla, johon vastattiin vastatarjouksella. Tämä prosessi saattoi olla hyvinkin aikaa vievää. Fidelixin myynnin kasvaessa markkinointi- ja hallinto-osastolla todettiin, että tilauksia tulee liikaa käsin käsiteltäviksi. Verkkokaupan avulla nopeutetaan tilausten käsittelyä automatisoimalla tarjousten laskeminen ja sähköpostien lähettäminen. Manuaalisessa prosessissa tapahtuvat lasku- ja kirjoitusvirheet saadaan myös poistettua.

Fidelixillä oli aiemmin yrityksen sisällä aloitettu pohja verkkokaupalle, joka tuli kehittää loppuun. Tavoitteena oli tehdä verkkokaupasta asiakkaan kannalta yksinkertainen ja helppokäyttöinen. Tavoitteena on kehittää sovellus aluksi suomalaisten yhteistyökumppaneiden käyttöön ja myöhemmin laajentaa käyttöä myös ulkomaisille asiakkaille..

Järjestelmän tulee tarjota tuotteiden tilausjärjestelmä yhteistyökumppaneille ja jälleenmyyjille, mikä olisi Fidelixille helposti ylläpidettävissä ja päivitettävissä graafisen käyttöliittymän kautta ilman ohjelmointiosaamista.

Tämä työ käsittelee Fidelix Oy:lle sisäisesti tehtyä verkkokauppaprojektia. Seuraavassa luvussa kuvataan yrityksen aiempi tilausprosessi tarkemmin. Luvuissa 3 ja 4 kuvataan projektin vaatimukset ja niiden hallinta sekä projektin edistymisen seuranta. Luvussa 5 kuvataan käytetyt tekniikat ja luvussa 6 arkkitehtuuri, funktiot ja toiminnot ja projektin toteutus tarkemmin. Lopuksi luvussa 7 käsitellään verkkokaupan jatkokehitystä.

2 Projektin lähtökohdat

Fidelix Oy:llä ei ollut aiempaa automatisoitua tilausjärjestelmää, vaan kaikki tilaukset tehtiin sähköpostilla ja lopulliset hinnat laskettiin itse. Yrityksen myynnin noustessa todettiin, että tilausten käsittelyyn käytetään liikaa aikaa siihen nähden, että sen voisi automatisoida. Vanhan tilausprosessin kulku meni niin, että asiakas lähetti sähköpostilla tilauksen haluamistaan tuotteista, jonka jälkeen Fidelixiltä lähetettiin tarjous. Tarjouksia saatettiin lähetellä edestakaisin useampaan kertaan ennen kauppojen syntymistä. Verkkokaupan kautta hinnoittelu ja tarjousten tekeminen on helpompaa.

Projektin ohjaaja Reino Ederveen oli muiden töiden sivussa työstänyt verkkokaupan Poc (proof of concept) -toteutusta ja siinä oli käytetty PHP:tä ja Sql:ää. Toteutus piti sisällään tuotteiden listauksen, asiakkaiden ja ylläpitäjien käyttöliittymät. Ylläpitäjien käyttöliittymässä oli vain nykyisten tuotteiden hallinta, eli nimen ja hinnan muuttaminen. Verkkokaupasta puuttui kriittisiä toimintoja, jotta verkkokauppa voitaisiin ottaa käyttöön.

Tärkeimpänä ominaisuutena oli tuotteiden lopullisten hintojen laskeminen sivustolla. Hinta määräytyi useamman muuttujan mukaan ja manuaallisesti laskiessa oli mahdollista tehdä virheitä, joten tilausten käsittely haluttiin siirtää kokonaan verkkokauppaan. Ylläpitäjien sivustolle tuli lisätä tarvittavat funktiot, kuten alennusten määrittely ja uusien tuotteiden ja käyttäjien lisääminen.

3 Projektin toteutus

Verkkokaupan toteuttaminen haluttiin aloittaa mahdollisimman pian, joten vaatimusmäärittelyn laatiminen tehtiin pääpiirteittäin. Vaatimusmäärittelyä tarkennettiin työn edistyessä sprinttien välissä.

3.1 Vaatimusmäärittely

Projekti aloitettiin laatimalla vaatimusmäärittely yhdessä projektin ohjaajan kanssa. Vaatimusmäärittelyssä kartoitettiin verkkokauppaan toteutettavat ominaisuudet ja toiminnot. Kartoitetut toiminnot kategorisoitiin projektin kannalta tärkeysjärjestykseen. Tärkeimpänä ominaisuutena oli tuotteiden hintojen laskeminen ja asiakkaiden alennusten määrittelemineen, joista projekti aloitettiin. Projektia toteutettiin Scrum-menetelmän tapaisella työtavalla. Projektin toteutus jaettiin osiin eli ns. "sprintteihin". Yhden sprintin tavoitteena on toteuttaa projektin kannalta jokin oleellinen osa-alue tai toiminto. Jokaisen sprintin lopuksi sivustosta oli valmiina uusi toimiva versio, jota tarkasteltiin muiden työntekijöiden kanssa. [1.]

Ensimmäisessä sprintissä tavoitteena oli toteuttaa tuotteen lopullisen hinnan laskeminen. Aloittaessa tuotteella oli vain vakiohinta määriteltynä tietokannassa tuote-tauluun. Alennuksen laskemiseksi tietokantaan asiakkaalle piti lisätä uusi kenttä, jossa määriteltiin asiakkaan alennusprosentti. Sprint 1 jouduttiin käytännössä tekemään uudelleen myöhemmin, kun lopullisen hinnan laskemisen logiikkaa haluttiinkin muuttaa toislaiseksi.

Toisessa sprintissä lisättiin asiakkaalle mahdollisuus lisätä sähköpostiosoitteita, joihin tilausvahvistuksesta ja tilauksen tilan seurannasta lähetetään kopiot. Lähettämisen pohjana toimi Pearista saatu sähköpostinlähetykspaketti.

Kolmannessa sprintissä keskityttiin ylläpitosivuston kehittämiseen. Sivustolle lisättiin mahdollisuus asiakkaiden lisäämiselle joko tyhjistä tai käyttämällä olemassa olevaa asiakasta pohjana. Myös alennusten lisääminen ja muokkaaminen toteutettiin kolmannessa sprintissä.

Neljännessä sprintissä toteutettiin ensimmäinen sprintti uudestaan muuttuneiden vaatimusten mukaisesti. Aiemmassa versiossa asiakkaalla oli alennusprosentti, joka laski kaikkien tuotteiden hintaa. Uudessa versiossa lisättiin mahdollisuus määritellä asiakkaalle tuotekohtainen alennus. Sitä varten luotiin uusi taulu tietokantaan, jossa määriteltiin tuotteentunnus, asiakkaan tunnus ja annettu alennus.

Viidennessä sprintissä paranneltiin ulkoasua ja koodin rakennetta. Funktioita ja kyselyitä siirrettiin omiin tiedostoihin, millä saatiin koodista selkeämpää.

Sprinteissä asiakkaan sivun kehitys oli etusijalla ja yrityksen sisäisessä käytössä olevat sivut ja toiminnot olivat vasta sen jälkeen.

Työn edistyessä ja tärkeimpien ominaisuuksien valmistuessa määrittelyä täydennettiin kyselemällä muilta työntekijöiltä, minkälaisia ominaisuuksia tai parannuksia verkkokauppaan vielä toivottaisiin. Vaatimusmäärittely tehtiin pääpiirteittäin työlistamaiseksi, koska projekti haluttiin aloittaa mahdollisimman pian. Määrittelyä tarkennettiin ja täydennettiin verkkokaupan toteuttamisen ohessa sitä mukaa, kun uusia tarpeita ilmeni. Työlista koostui tehtävistä tärkeysjärjestyksessä. Se oli riittävä ja mahdollisti nopean työskentelyn ja työn edistymisen seurannan.

3.2 Projektisuunnitelma

Projektisuunnitelmassa oli arviot eri ominaisuuksien toteuttamiseen menevästä ajasta, niiden testauksesta ja sprinttien tavoitteet. Tällä oli tarkoitus helpottaa sisäistä tiedotusta. Myös sprinttien valmistumisaikataulujen avulla verkkokaupan käyttäjien oli helpompi varautua uuden version julkaisemiseen ja sen testaamiseen. Projektisuunnitelmassa määriteltiin myös alustavat aikataulut verkkokaupan esittelytilaisuuksille, joissa mietittiin suuremmalla ryhmällä verkkokaupan toimintoja ja lisä-tarpeita.

3.3 Projektin raportointi

Projektin edistymistä seurattiin useammalla eri tavalla. Edistymisestä raportointiin omatoimisesti sähköpostilla projektin ohjaajalle. Projektin ohjaaja kävi paikanpäällä tarkastelemassa projektin edistymistä. Joskus verkkosivu annettiin projektin ohjaajan testattavaksi, jolloin projektin tila myös on nähtävillä konkreettisesti.

Sähköpostilla raportointi oli pääasiassa tiedotustyönedistymisestä, jotta projektin ohjaaja tietää projektin nykyisen tilan. Tapaamisissa katsottiin tarkemmin koodia ja millä menetelmillä ominaisuudet toteutettaisiin tai miten jokin ongelma kannattaisi ratkaista.

Ennen verkkosivun käyttöönottoa testaajia oli vain kaksi, joten virheraportointi oli useimmiten kahdenvälistä sähköpostiviestittelyä löydetyistä ongelmista. Virheet kategorioitiin kriittisyyden mukaan. Vakavat tuli korjata pikimmiten, sillä ne vaikuttavat käytettävyyteen tai estävät käytön kokonaan. Toisen asteen virheet olivat pääasiassa ulkoasuongelmia, joiden prioriteetti oli matala, ja ne korjattiin viimeisenä, jos ne eivät estäneet sivuston käyttöä.

Verkkokauppa otettiin käyttöön 17.10.2014, eikä toistaiseksi virheilmoituksia ole tullut.

3.4 Loppuraportti

Loppuraportti tehtiin Fidelix Oy:lle projektin ohjaajan pyynnöstä. Fidelixille tehty loppuraportti on myös tämän raportin liitteenä. Raportti toimii käyttöoppaana ja sisältää myös lyhyitä esimerkkejä käytetyistä tekniikoista ylläpitoa ja jatkokehitystä varten kuten:

- Ajax-funktiot: selitykset mitä ne tekevät ja miten niitä voidaan käyttää uusien sivujen ja toimintojen toteuttamisessa, jos tarve vaatii.
- Verkkosivu- ja palvelinkoodit eli PHP- ja SQL-syntaksia.

4 Projektin toteutuksen työvälineet

Koska projektina oli verkkokauppa, sen toteutukseen käytettiin PHP-ohjelmointikieltä, SQL-tietokantaa, jossain määrin html-ohjelmointikieltä ja Ajaxia, joka pohjautuu javascriptiin. Ajaxin käyttöä ei ollut suunniteltu etukäteen, vaan se keksittiin työtä tehdessä vastaan tulleiden ongelmien ratkaisuksi. Se todettiin hyväksi työvälineeksi ja myös oman oppimisen kannalta hyödylliseksi, joten sitä pyrittiin käyttämään mahdollisimman paljon. Koska käytettävissä oli jo keskeneräinen verkkokauppa-pohja, emme käyttäneet valmista viitekehystä projektissa.

4.1 PHP-ohjelmointikieli

PHP-ohjelmointikieli on suunniteltu verkko-ohjelmointiin ja on erityisen hyvä palvelinpuolella ajettaviin toimintoihin, eli tavallisesti PHP-koodi ajetaan web-palvelimella. Tämä mahdollistaa verkkosivustojen dynaamisen luonnin ja lopputuloksen näyttämisen käyttäjälle.

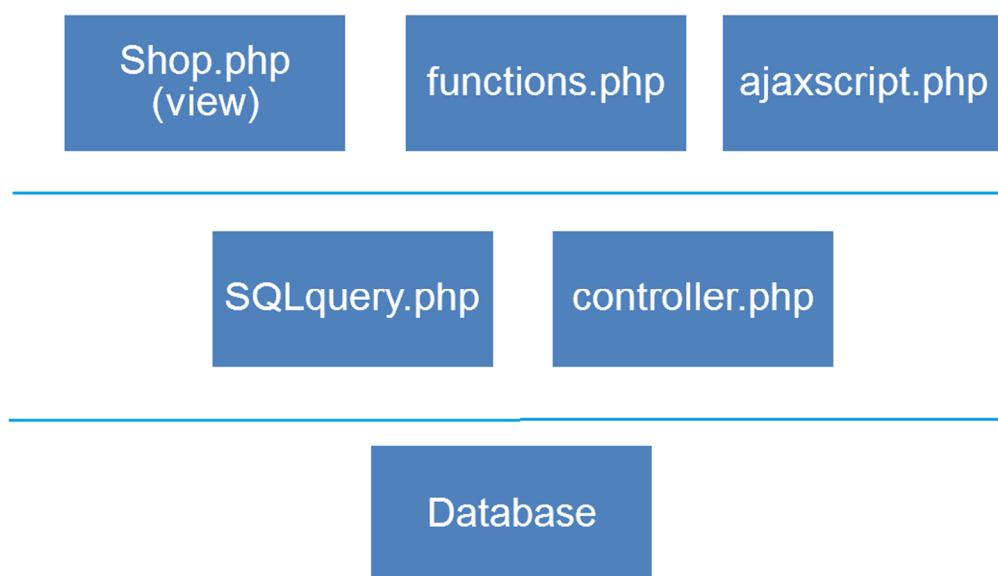
PHP on heikosti tyyppitetty ohjelmointikieli, eli sen muuttujia ei tarvitse tyyppittää, eikä usein tyyppitetäkään, tietyntyyppiseksi merkkijonoksi, olioksi tai kokonaisluvuksi. Tämän takia vertailuja tehdessä täytyy olla tarkka, mitä tietoa muuttujaan on tallennettuna. Esimerkiksi verkkokaupassa tuli vertailtua kokonaislukua ja murtolukua, jonka takia vertailu meni väärin ja lopulliseksi hinnaksi saatiin vääriä lukemia. [2.] Muuttujan tyyppin voi tarkistaa `Php.funktiolla` `gettype()` joka palauttaa muuttujan tyyppin. Toinen tapa tarkistaa muuttujan tyyppi, esimerkiksi dataa syöttäessä, on käyttää jokaiselle tyyppitykselle erikseen tarkoitettua funktiota, joka palauttaa `true` tai `false` arvon. Yksi näistä funktioista on `is_int()`, jolla tarkistetaan onko muuttuja kokonaisluku.

PHP-koodin ajamiseen kuitenkin vaaditaan sivun uudelleen päivitys, jotta palvelin saa tarvittavat tiedot esim. käyttäjän syöttämät tiedot kenttiin ja/tai nappien painallukset. Vaadittavan sivuston uudelleenlataamisen välttämiseksi projektissa käytettiin myös Ajaxia, josta kerrotaan tarkemmin sen omassa luvussa.

PHP valittiin, koska projektin Poc-toteutus oli aloitettu PHP:llä. PHP on muutenkin suhteellisen helppo ja monipuolinen kieli, joten sitä olisi luultavasti käytetty joka tapauk-

sessä. Kirjoitettu PHP-koodi oli yksinkertaista PHP:tä ilman olioita, sillä se on helpommin lähestyttävissä muille Fidelixin työntekijöille. Toteutuksessa ei käytetty mitään valmista viitekehystä, vaan koko sivustoprojekti on Fidelixin sisäisesti tuotettu. Viitekehystä ei käytetty, koska pohja oli aloitettu ilman viitekehystä ja sellaisen käyttöönottoa ei nähty tarpeellisena projektin toteutuksen aikana.

Koodia ei ollut muuten luokiteltu kuin funktiot, ja Ajax-skriptit oli jaettu omiin tiedostoihinsa. Muuten koodi kirjoitettiin noudattaen pohjassa käytettyä järjestystä. Koodin alussa on tietokantayhteyksien määrittelyt ja session alustaminen. Seuraavana on käyttöliittymän luonti SQL-kyselyistä saadun datan avulla. Viimeisenä ovat funktiot ja muut toiminnot.



Kuva 1. Koodin rakenne ja jaottelu

4.2 Relaatietietokanta

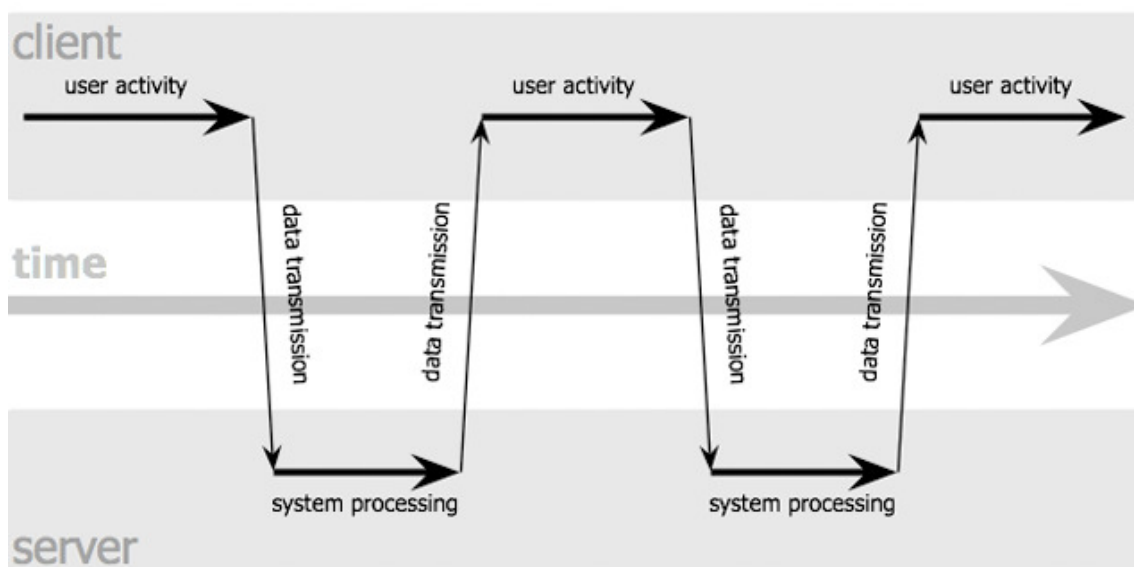
Verkkokaupan sisältämän datan hallintaan päätettiin käyttää MySQL-tietokantaa. MySQL-tietokanta valittiin sen tehokkuuden takia, etenkin pienillä tietomäärillä. Projektin työstämisen aikana tauluja oli noin kymmenen. Yhtenä MySQL:n etuna on myös se, että MySQL-tiluihin on mahdollista lisätä indeksointia, jotka nopeuttavat SQL-kyselyiden suorittamista, mikä näkyy suoraan verkkosivujen toimintanopeudessa. Tämä on tarpeen, jos tietokannan sisältämän datan määrä kasvaa huomattavan suureksi, eli vähintään satoihin tuhansiin riveihin.

MySQL-kantaan tehdään hakuja, muutoksia ja lisäyksiä SQL-kyselykielellä. MySQL-kantaa hallinnoitiin phpMyAdmin-työkalulla. phpMyAdmin on ilmainen avoimen lähdekoodin työkalu, jota käytetään verkkoselaimella. phpMyAdmin-työkalun kautta suoritettiin taulujen luonti-, muokkaus- ja päivityskyselyt. [3.]

MySQL-kanta valittiin sen helppouden ja yleisyyden takia. MySQL-ohjelmisto on saatavilla vapaalla GNU GPL -lisenssillä, joka on huomattava etu, sillä siitä ei tule lisäkuluja.

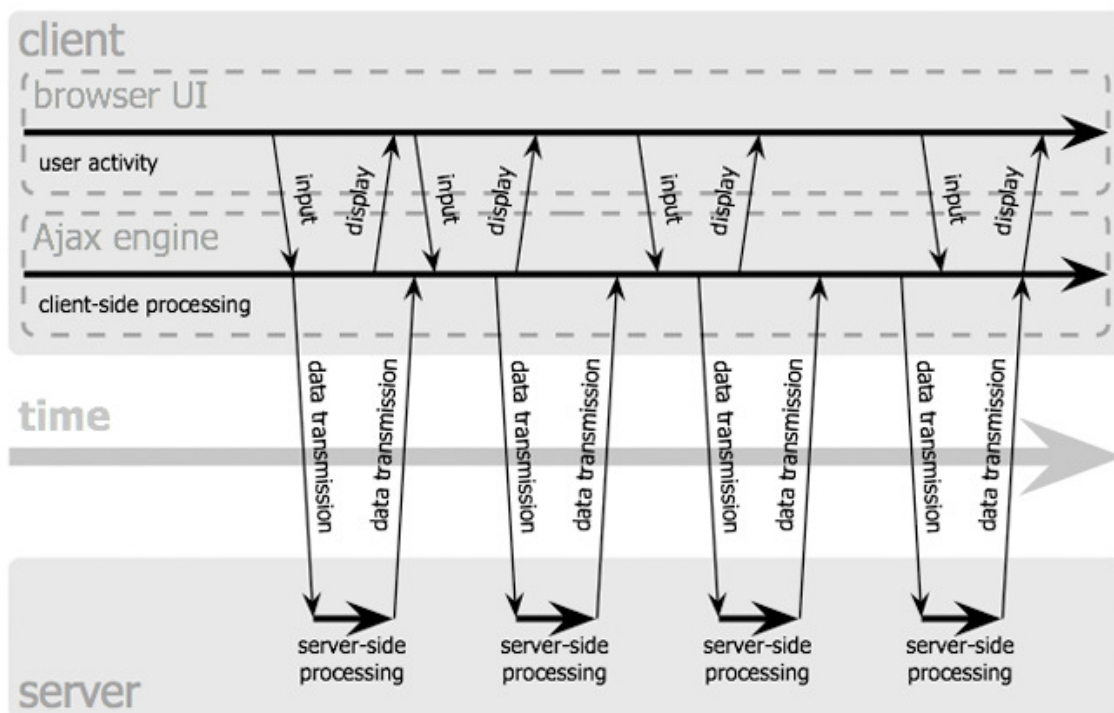
4.3 Ajax

Projektissa sovellettiin myös Ajax-tekniikoita, koska sivusto luotiin PHP:llä, ja PHP-funktioiden kutsuminen vaatii sivun uudelleenlataamisen. Osa toiminnoista haluttiin suoritettaviksi ilman sivun uudelleenlataamista käyttäjäystävällisyyden takia. Ajax (Asynchronous Javascript and xml) mahdollistaa reaaliaikaisen dataliikenteen palvelimen ja selainohjelman välille. Dataliikenne suoritetaan taustalla niin, ettei koko verkkosivua tarvitse ladata uudelleen. Ajax käyttää Javascriptiä asynkronisesti lähettämään ja vastaanottamaan dataa palvelimelta. Saatu data kerätään XML:llä verkkoselaimelle. Ajaxin käyttö nopeuttaa verkkosivujen käyttöä, parantaa käytettävyyttä ja vähentää myös jossain määrin palvelimiin kohdistuvaa raskautta tekemällä dataliikenteestä kevyempää. [4.]



Kuva 2. Perinteinen verkkosivu ilman Ajaxia [6]

Kuvassa 2 näkyy esimerkki perinteisestä verkkosivusta. Käyttäjä lähettää datan palvelimelle käsiteltäväksi, odottaa sen käsittelyä ja vastaanottaa sen. Tämä ilmenee sivun uudelleen lataamisena. Tässä tapauksessa dataliikenne on synkroninen. Kuvassa 3 on kaavio Ajax-käyttöisestä verkkosivusta. Käyttäjä voi lähettää dataa tai suorittaa muita toimintoja verkkosivulla samalla, kun data on käsiteltävänä palvelimella. Verkkosivun käyttöaika ei katkea missään vaiheessa. Verkkosivun toiminta on asynkroninen



Kuva 3. Verkkosivu Ajaxilla [6]

4.4 Ohjelmointi-tiedostoeditori

Ohjelmoinnissa työkaluna käytettiin Adobe Systemsin Adobe Dreamweaveria. Dreamweaverin ominaisuuksia ovat muun muassa syntaksin korostus ja syntaksin automaattinen täyttö jotka nopeuttavat koodin kirjoittamista huomattavasti.

Jos työstettävässä koodissa on viittauksia/liitteitä toisiin tiedostoihin, lähdekoodeihin tai sivustoihin (include), Dreamweaver hakee ja näyttää ne automaattisesti välilehdissä, mikä nopeuttaa huomattavasti tiedostojen selaamista. Dreamweaverin oma tekstinhalu- ja korvaustoiminto on myös tehokas. Haussa voi määritellä hakuehtoja melko tarkasti, kuten hakeeko koodia nykyiseltä työstettävältä sivulta, koko sivustolta, valitusta polusta tai kansioista. Dreamweaverissa editointi-näkymän voi jakaa useammalle tiedostolle tai pitää lähdekoodi näkymää ja ”design view” -näkymää, joka on niin sanottu WYSIWYG (engl. What you see is what you get eli mitä näet, sitä saat) –editori jossa voi muokata sivustoa graafisessa näkymässä drag and drop -menetelmällä. Design view nopeuttaa ulkoasun suunnittelua ja toimivuuden testausta. Dreamweaverissa on myös sisäänrakennettu verkkoselain nopeaan verkkosivun tarkasteluun, joka on varsin kätevä javascriptin testaamisessa. Sitä käyttäessä ei tarvitse päivittää erillistä verkkoselainta vaan voi suoraan editorissa tarkastella sivua ja muokata koodia nopeasti.

4.5 Muuta

Toistaiseksi verkkokauppaan liittymättömänä sivu-projektina Fidelixin ”ajankohtaista”-sivuille uuden uutisen lisääminen muutettiin semmoiseksi, että uutisen lisääminen ei vaadi ohjelmointitaitoa. Ennen uutiset lisättiin tekemällä muutoksia sivun html-koodiin. Nyt uutiset lisätään käyttämällä CKEditor-lisäosaa. CKEditor on avoimen lähdekoodin WYSIWYG-tekstieditori. CKEditorin tekstinmuokkausnäkyssä nähtävä uutinen vastaa lähes täydellisesti lopullista julkaistua uutista ”ajankohtaista”-sivulla. [5.]

Predate: (optional, startdate for multi-day events)

Date: YYYY-MM-DD (visible on the homepage and in the event title as DD.MM.YYYY)

Location: (optional. In the event title after the date)

Short title: (max. 130 characters; the title that appears on the home page in the news feed)

Long title: (The title for the item in the news page)

Text:

Type your text here ...

Kuva 4. CKEditor-tekstieditori

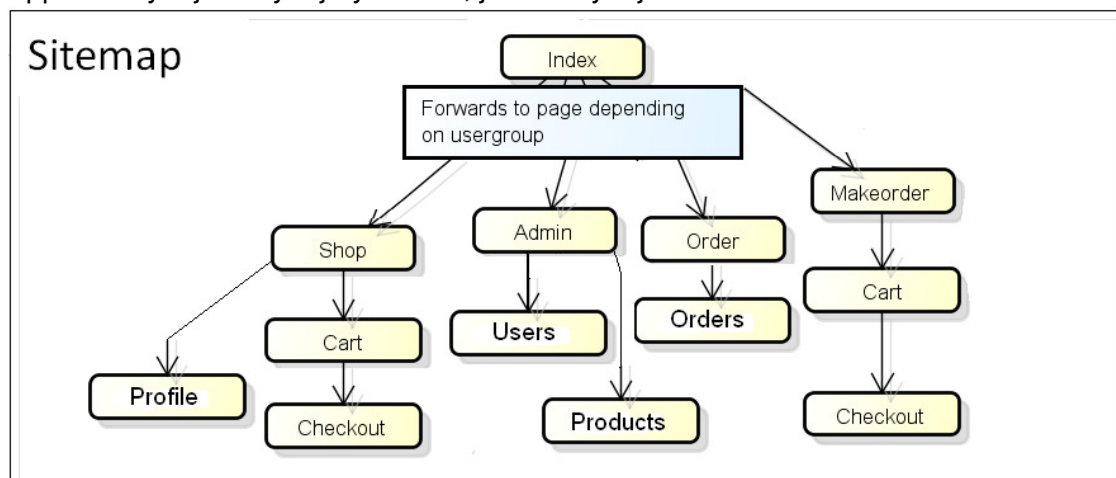
Kyseinen järjestelmä otettiin käyttöön enemmänkin testaukseen uutissivustolle, jotta voitaisiin arvioida, onko sille käyttöä verkkokaupan puolella. Esimerkiksi CKEditorilla olisi helppo lisätä mainoksia uusista tuotteista verkkokauppaan tai muuta sinne sopivaa sisältöä kuten tietoa tuote-esittelyistä tai messuista.

5 Järjestelmän toteutus

Fidelixillä oli keskeneräinen verkkokaupan runko tehtynä, joka oli tarkoitus laajentaa käyttökelpoiseksi. Pohja oli yksinkertaista PHP:tä ilman olioita tai valmista viitekehystä. Osa funktioista kuten sähköpostin lähettäminen oli toteutettu Pearin sähköposti-komponentilla. Pohjan oli tehnyt projektin ohjaaja Reino Ederveen muiden töiden lomassa. Verkkokaupasta puuttui oleellisia ominaisuuksia ja joitain toteutettuja toimintoja piti tehdä uusiksi paremmin. PEAR (PHP Extension and Application Repository) on viitekehys- ja jakelujärjestelmä, jonka tarkoitus on edistää yleisten funktioiden lähdekoodin uudelleenkäyttöä. Pear siis tarjoaa jäsennetyn koodikirjaston uudelleenkäytettäville PHP-komponenteille ja järjestelmän niiden ylläpitoon ja jakeluun. Yksinkertaisimmillaan Pear-paketin lähdekoodi otetaan käyttöön sisällyttämällä (include) se oman PHP-sivun lähdekoodiin.

5.1 Rakenne

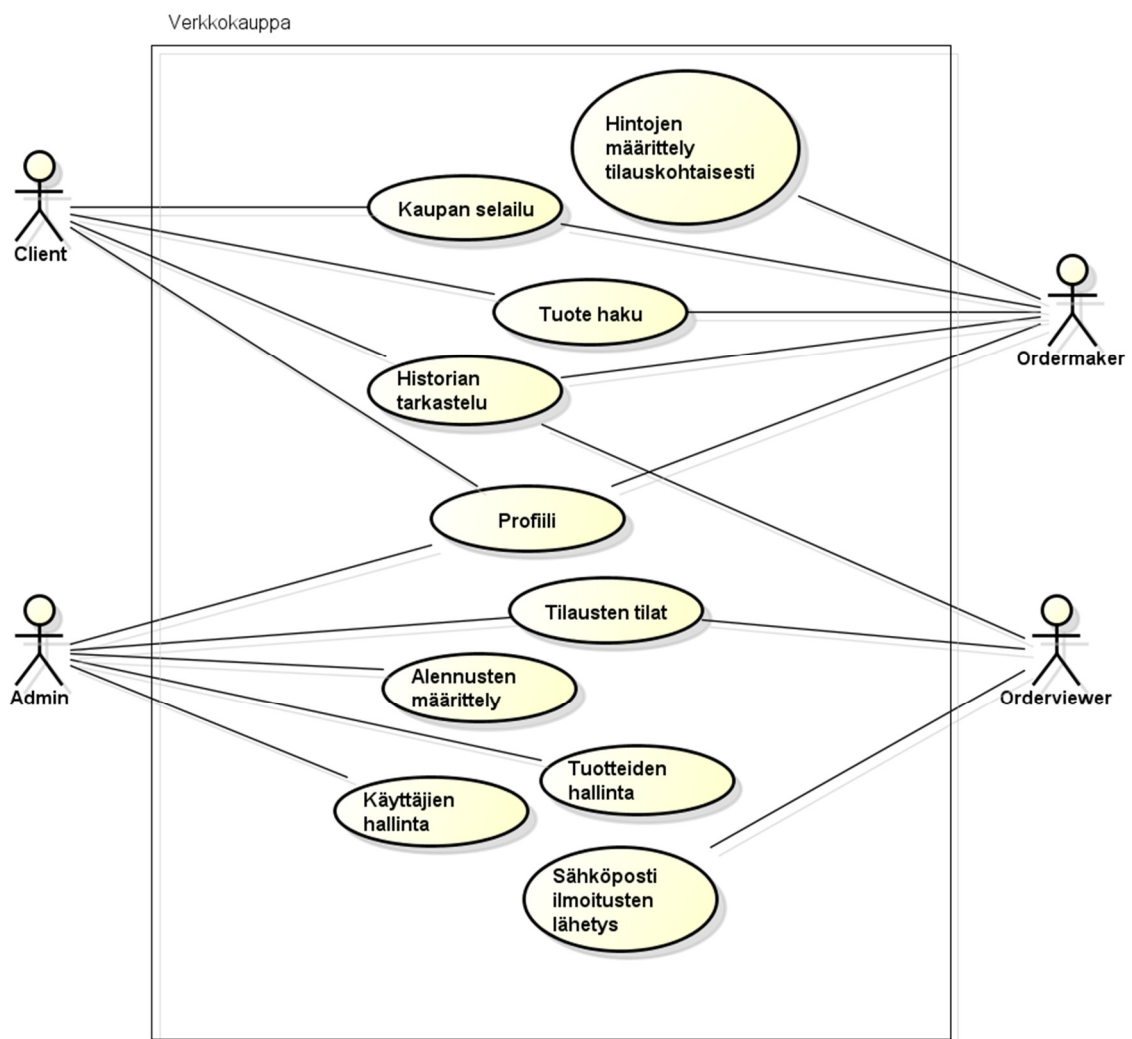
Verkkokauppaprojekti sisältää neljä sivustoa, joihin käyttäjä ohjataan kirjautumisen jälkeen käyttäjäryhmän mukaan, johon käyttäjä kuuluu. Tavalliset asiakkaat ohjataan kauppaan. Fidelixin sisällä on kolme erilaista käyttäjäryhmää. Admin-ryhmään kuuluvat ohjataan ylläpitosivulle. Ordermaker-ryhmä ohjataan Makeorder-sivulle, jossa voidaan tehdä tilauksia toisen käyttäjän puolesta. Orderviewer-ryhmä ohjataan Order-sivulle, jossa voidaan tarkastella tilauksia, niiden tiloja, muokata tiloja ja lähettää asiakkaille sähköposti-ilmoituksia. Kuvassa on sivuston rakenne ja uudelleen ohjauksen logiikka riippuen käyttäjän käyttäjäryhmästä, johon käyttäjä kuuluu.



Kuva 5. Sivuston rakenne

Sivuston eri toiminnot ovat käytössä eri käyttäjäryhmille kuvan osoittamalla tavalla. Jotkin toiminnot muuttuvat hieman käyttäjäryhmästä riippuen. Admin- ja ordermaker-ryhmät voivat katsoa kaikkien profileja, kuin client näkee vain omansa. Samoin ordermaker näkee kaikkien historiat.

Admin-ryhmässä olevat ovat ylläpitäjiä ja Fidelixin tietohallinnon työntekijöitä ja IT-tuki-henkilöitä. Admin voi muokata tuotteita, käyttäjäryhmiä ja käyttäjiä. Ordermaker-ryhmässä on pääasiassa Fidelixin markkinointiosaston työntekijöitä. Ordermaker voi tehdä tilauksia asiakkaan puolesta ja määrittellä erikoishintoja tilauskohtaisesti. Orderviewer-ryhmään kuuluu tavallisesti varasto- ja logistiikka-osastoilla työskenteleviä. Orderviewerin tehtävä on päivittää tilausten ja tuotteiden tilat (esim. lähetetty, loppu, jne) ja lähettää niistä ilmoitusviestit asiakkaalle.



Kuva 6. Use-case diagrammi

5.2 Kauppa-sivuston toiminnot

Hintojen laskeminen

Tuotteiden hintojen laskeminen oli tärkein ja monimutkaisin ominaisuus. Tuotteen hinnalle on kolme eri alennusprosenttia tallennettuna MySQL-kannassa oleviin tauluihin. Eri tauluihin syötetyt alennukset omaavat eri prioriteetit. Käyttäjä-taulussa oleva alennus pätee kaikkiin tuotteisiin. Tätä kutsutaan koodissa "baseDiscount"-nimellä. Tuote-taulussa oleva alennus pätee kyseiseen tuotteeseen kaikille asiakkaille. Alennus-taulussa määritellään asiakkaan alennus tiettyyn tuotteeseen, eli Asiakas-Tuote-relaatio. Jos käyttäjä-taulussa on määritelty alennus, ja se on suurempi kuin Tuote-taulussa oleva alennus, niin silloin käytetään Käyttäjä-taulussa olevaa alennuskertoimta. Jos Alennus-taulussa on määritelty tuotteelle alennus, niin se korvaa kaikki muut alennukset siitä huolimatta, onko se suurempi tai pienempi kuin muut.

Asiakkaan kaikkiin tuotteisiin kohdistuva alennus on tallennettu sessioon sisäänkirjautuessa, ja muut alennukset haetaan tietokannasta. Alennusten arvoja vertaillaan yksinkertaisella else-if-rakenteella. Alennusten tarkistuksen lopussa vielä tarkistetaan, ettei alennus ole "100". 100 prosentin alennusta käytetään tuotteiden piilottamiseksi asiakailta, esim. jotain tuotteita ei myydä tiettyihin maihin.

```

mysql_connect($server_address,$username,$password);
@mysql_select_db($database) or
die( "Unable to select database");
$sql = mysql_query("SET CHARACTER SET 'utf8'") ;

//Connecting to database and check the individual discount%
$uusi=mysql_query(" SELECT Discount FROM testdiscount
WHERE AsiakasId=$_SESSION[userid] AND TuoteId=$prodID ")
or die(mysql_error()) ;
$uusitesti=mysql_fetch_array($uusi);

//IF-check does item specific discount exist,
if true use, else use $_SESSION['baseDiscount']

if ($uusitesti[Discount]==null)
{
    if ($_SESSION[baseDiscount]<$prodPRICE2)
    {
        $uusifinal=$prodPRICE2;
    }
    if ($_SESSION[baseDiscount]>$prodPRICE2)
    {
        $uusifinal=$_SESSION[baseDiscount];
    }
}
else {
    {
        $uusifinal=$uusitesti[Discount];
    }
}
//Adding the price reduction
$prodPRICE=$prodPRICE*((100-$uusifinal)/100);
$prodPRICE=round($prodPRICE,2);
$prodTYPE=mysql_result($result,$i,"prodTYPE");
$prodSUBCAT=mysql_result($result,$i,"prodSUBTYPE");
$temp = $_POST['counter'];
if ($uusifinal!=100 && $prodPRICE2!=100)
{

```

Koodiesimerkki 1. Alennuksen laskemisen logiikka

Sähköposti-ilmoitukset

Asiakkaalla on tietokannassa määritelty yhteyssähköposti, johon tilauksesta lähetetään automaattinen vahvistusviesti. Fidelixillä todettiin, että asiakkaalla pitäisi olla mahdollisuus lisätä vastaanottajia, joille vahvistuksesta lähetettäisiin kopio. Toimintoon tarvittiin sähköpostin oikeellisuuden tarkistus. Käyttäjäystävällisyyden vuoksi toiminto tuli toteut-

taa niin, ettei sivustoa tarvitsisi ladata uudestaan. Loogisin ja toimivin keino toiminnon toteuttamiseen oli Ajax. Tekstikenttään kirjoittaessa kutsutaan Ajax-skriptiä, joka lähettää tekstin http:n POST-pyyntöä käyttäen toiselle check.php-nimiselle PHP-sivulle tarkistettavaksi. Koodissa hyödynnetään XMLHttpRequestia, jolla data lähetetään palvelimelle uudelleen lataamatta kokoverkkosivua.

```

this.checkAjax = function(elm) {
if(elm.value != '') {
    var xmlhttp = (window.ActiveXObject) ? new ActiveXObject("Microsoft.XMLHTTP")
    : new XMLHttpRequest();
var  datatosend = elm.name +'='+ elm.value;

xmlhttp.open("POST", phpcheck, true);

xmlhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
xmlhttp.send(datatosend);

    xmlhttp.onreadystatechange = function() {
        if (xmlhttp.readyState == 4) setelm(elm, xmlhttp.responseText);
    }
    else setelm(elm, '');
}
}
var chkF = new checkFormElm();
document.getElementById('extrarecipients').onblur = function()
{ chkF.checkAjax(this); }

```

Koodiesimerkki 2. Lyhyt otos kyseisestä funktiosta. Viimeisellä rivillä näkyy että funktiota kutsutaan kun tekstikentästä poistutaan (onblur).

Check.phpssa sähköpostit otetaan POSTista ja tarkistetaan preg_match-funktiolla. Koska sähköpostiosoitteita voi olla useita, ne erotetaan toisistaan ;-merkillä. Jos ei sallittuja merkkejä löydy, näytetään virheilmoitus.

```

if (isset($_POST['extrarecipients']))
{
    $extramails2 = explode(";", $_POST[extrarecipients]);

    for ($loop2 = 0; $loop2 < count($extramails2); $loop2++) {
        $extramail = $extramails2[$loop2];

        if (!preg_match
            ('/^[_a-z0-9-]+(\.[_a-z0-9-]+)*@[a-z0-9-]+(\.[a-z0-9-]+)*(\.([a-z]{2,4}))$/i',
            $extramail))
        {
            $reout = $err['extrarecipients'];

```

Koodiesimerkki 3. Sähköpostin tarkistus

Sähköpostin sisältö koostuu objekteista, joihin määritellään sähköpostin sisältö, esimerkiksi taulukko, jossa on tilauksen tuotteet, määrät ja hinnat. Foreach-silmukassa käydään ostoskorin sisältö läpi, jonka mukaan sähköpostissa oleva taulukko täytetään.

```
ob_start();
echo "<table border='1' bordercolor='\"#CCCCCC\"'
style='\"border-collapse:collapse\"' width='\"100%\"'>";
foreach($order as $key => $value)
{
    echo "<tr><td><no>\".$mail_processed_product_header_txt.\": \"
    .$value['description'].
    \"</no></td><td><no>\". $shop_details_quantity_txt.
    \": \".$value['quantity'].\"</no></td><td>\"
    .$mail_processed_piece_price_header_txt.
    \": \".$value['piece price'].\"</td>
    <td><no>\".$checkout_confirm_subtotal_header_txt.\":
    \".$value['total amount price'].\" &euro;\";
    if (!($value['comments']=="-")) echo "</no></td><td>\"
    .$mail_processed_comments_header_txt.
    \": \".$value['comments'];else echo "</td><td>&nbsp;\";
    echo("</td></tr>");
}
echo "<tr height=20px><td colspan='5'>&nbsp;</td></tr>";
echo "<tr><td colspan='3'>&nbsp;</td><td><no>\"
    .$checkout_confirm_total_header_txt.
    \": \".$grand_total_price.\" &euro;</no></td><td><no>\";
    if (!($remarks=="NULL")) echo $mail_processed_general_remarks_header_txt.
    \": \".$remarks.\"</no></td></tr>\"; else echo "&nbsp; </no></td></tr>\";
    echo "<tr><td colspan='3'>&nbsp;</td><td><no>\"
    .$mail_processed_fidelix_reference_header_txt.
    \": \".$orderId.\"</no></td><td><no>\" ;
    if (!($userREF=="NULL")) echo $mail_processed_customer_reference_header_txt.
    \": \".$userREF ; else echo "&nbsp;\" ;
    echo "</no></td></tr></table>\" ;
    $orderlisting = ob_get_contents();
ob_end_clean();
```

Koodiesimerkki 4. Sähköpostin sisällön rakentaminen.

Sähköposti lähetetään Pearin Mail/mime.phpta hyödyntäen. Mime.php sisällytettiin (include) sähköpostin lähetykseen koodiin palvelin- ja lähetyksasetusten lisäksi. Koodissa määritellään palvelinasetukset, viestin koodaus/muoto ja vastaanottajat.

```

require_once "Mail.php"; require_once "Mail/mime.php";

$from = "$fidelix_sender_name <verkkokauppa@fidelix.fi>";
$to = "?utf-8?B?".base64_encode($fidelix_recipient_name)."?="
. " <$fidelix_recipient_email_address>" ;
$bcc = "reino.ederveen@fidelix.fi"; //$recipients = $to; --> this is the old way
$recipients = $to;
$subject = $fidelix_subject ;

$mime = new Mail_mime(); $mime->setHTMLBody($fidelix_mail_body);

$host = "smtp.nbinetworks.fi"; $SMTP_username = "verkkokauppa@fidelix.fi";
$SMTP_password = "xxxx"; $SMTP_port = "25";

$mimeparams=array(); $mimeparams['text_encoding']='7bit'; $mimeparams['text_charset']='UTF-8';
$mimeparams['html_charset']='UTF-8'; $mimeparams['head_charset']='UTF-8';

$fidelix_mail_body = $mime->get($mimeparams);

$headers=array(); $hdrs['From']=$from; $hdrs['To']=$to; $hdrs['Subject']=$subject;
$headers = $mime->headers($hdrs);

$smtp = Mail::factory ('smtp', array('host'=>$host, 'port'=>$SMTP_port, 'auth'=>false));

$mail = $smtp->send($recipients, $headers, $fidelix_mail_body);

```

Koodiesimerkki 5. Sähköpostin lähettäminen.

Historia

Kauppaan lisättiin asiakkaalle mahdollisuus nähdä tilaushistoria ja toimituksen tila. Toiminto antaa listan tilatuista tuotteista, toimituksen/tuotteen tilan ja yleiset tiedot, kuten toimitusosoitteen. Klikkaamalla tuotetta pääsee kauppaan suoraan tuotteen kohdalle. Historian haku toteutettiin yksinkertaisella mysql-kyselyllä, joka hakee kaikki sisäänkirjautuneen käyttäjän tilaukset tietokannasta käyttäen Sessioon tallennettua käyttäjätunnusta (id). Kyselyn tulokset käydään läpi while-silmukassa ja niistä tehdään taulukko.

```

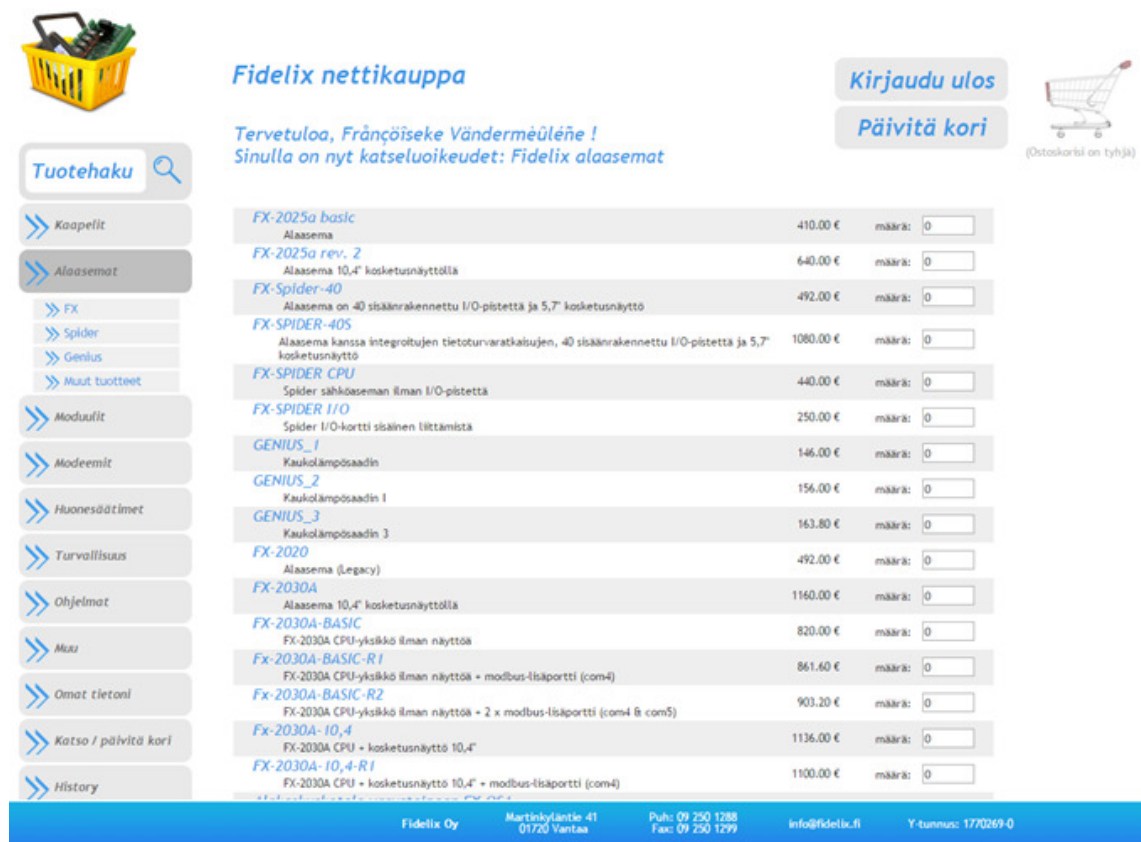
$sql="SELECT fx_orders.orderID, fx_orders.orderSTATUS, fx_orderitem.prodID,
fx_orderitem.quantity, fx_orderitem.remarks, product.prodNAME, fx_order_status.'
FROM fx_orders
LEFT JOIN fx_orderitem ON fx_orders.orderID = fx_orderitem.orderID
LEFT JOIN product on fx_orderitem.prodID = product.prodID
LEFT JOIN fx_order_status on fx_orders.orderSTATUS = fx_order_status.statusID
WHERE fx_orders.orderUSERid = ".$_SESSION['userid'];
//run the query against the mysql query function
$result=mysql_query($sql);
//create while loop and loop through result set
?>
|
<?php
while($row=mysql_fetch_array($result)){
    $nimi=$row['prodNAME'];
    $quantity=$row['quantity'];
    $status=$row['statusTXT_'.$lang];
    $orderid=$row['orderID'];

```

Koodiesimerkki 6. Historian haku.

Tuotteiden valinta

Kuvassa 7 sivupalkissa näkyvät tuotekategoriat, ja jos kategorialla on ala-kategorioita ne, tulevat painaessa näkyviin. Kategorialistaus luodaan sql-kyselyllä, jonka tulokset listataan menuun toistosilmukalla. Valittu kategoria tallennetaan posttiin, jonka jälkeen sitä käytetään valitun kategorian ala-kategorioiden hakemiseen, jos sellaisia on. Lopuksi tuotelista luodaan valitun kategorian ja mahdollisen ala-kategorian mukaan tehdystä sql-kyselystä.



Fidelix nettikauppa

Kirjaudu ulos Päivitä kori

Tervetuloa, Frånçöiseke Vändermeüléne !
Sinulla on nyt katseluoikeudet: Fidelix alaasemat

Tuotehaku

Kaapelit

Alaasemat

- FX
- Spider
- Genius
- Muut tuotteet

Moduulit

Modeemit

Huonesäätimet

Turvallisuus

Ohjelmat

Muu

Omat tietoni

Katso / päivitä kori

History

FX-2025a basic Alaasema	410.00 €	määrä: 0
FX-2025a rev. 2 Alaasema 10,4" kosketusnäyttöllä	640.00 €	määrä: 0
FX-Spider-40 Alaasema on 40 sisäänrakennettu I/O-pistettä ja 5,7" kosketusnäyttö	492.00 €	määrä: 0
FX-SPIDER-40S Alaasema kanssa integroitujen tietoturvaratkaisujen, 40 sisäänrakennettu I/O-pistettä ja 5,7" kosketusnäyttö	1080.00 €	määrä: 0
FX-SPIDER CPU Spider sähköaseman ilman I/O-pistettä	440.00 €	määrä: 0
FX-SPIDER I/O Spider I/O-kortti sisäinen liittäminen	250.00 €	määrä: 0
GENIUS_1 Kaukolämpöasemien	146.00 €	määrä: 0
GENIUS_2 Kaukolämpöasemien 1	156.00 €	määrä: 0
GENIUS_3 Kaukolämpöasemien 3	163.80 €	määrä: 0
FX-2020 Alaasema (Legacy)	492.00 €	määrä: 0
FX-2030A Alaasema 10,4" kosketusnäyttöllä	1160.00 €	määrä: 0
FX-2030A-BASIC FX-2030A CPU-yksikkö ilman näyttöä	820.00 €	määrä: 0
FX-2030A-BASIC-R1 FX-2030A CPU-yksikkö ilman näyttöä + modbus-lisäportti (com4)	861.60 €	määrä: 0
FX-2030A-BASIC-R2 FX-2030A CPU-yksikkö ilman näyttöä + 2 x modbus-lisäportti (com4 & com5)	903.20 €	määrä: 0
FX-2030A-10,4 FX-2030A CPU + kosketusnäyttö 10,4"	1136.00 €	määrä: 0
FX-2030A-10,4-R1 FX-2030A CPU + kosketusnäyttö 10,4" + modbus-lisäportti (com4)	1100.00 €	määrä: 0

Fidelix Oy Martinkyläntie 41 01720 Vantaa Puh: 09 250 1288 Fax: 09 250 1299 info@fidelix.fi Y-tunnus: 1770269-0

Kuva 7. Kuvankaappaus tuotenäkymästä

Oikea sql-kysely valitaan if-ehdoilla sen mukaan, onko kategorialla ala-kategorioita. Otoksessa myös näkyy tuotteiden haku valitun kategorian mukaan.

```
<?php $typeselectd = $prodTYPE; // "prodTYPE" from the URL
      $userselected = $_SESSION['usergroup'];
      // a value from the "usergroup" table in the DB

mysql_connect($server_address,$username,$password);
@mysql_select_db($database) or die( "Unable to select database");
$sql = mysql_query("SET CHARACTER SET 'utf8'") ;

$usergroupquery=mysql_query(" SELECT groupNUMBER FROM usergroup WHERE
groupNAME='$userselected' ") or die(mysql_error());
$usergroupnumber=mysql_fetch_array($usergroupquery);

if ($prodSUBCAT==NULL) $result=mysql_query(" SELECT * FROM product WHERE
prodTYPE='$typeselectd' ") ;
elseif ($prodSUBCAT=='nonSUBCATED') $result=mysql_query
(" SELECT * FROM product
WHERE prodTYPE='$typeselectd' AND prodSUBTYPE IS NULL ") ;
else $result=mysql_query(" SELECT * FROM product
WHERE prodTYPE='$typeselectd' AND
prodSUBTYPE='$prodSUBCAT' ") ;
$num=mysql_numrows($result) or die(mysql_error());
```

Koodiesimerkki 7. Tuotteiden listauksen koodi

Valitut tuotteet tallennetaan ostoskoriin, joka on moniulotteinen-tila taulukko sessiossa. Seuraava näkymä on ostoskori, josta nähdään valitut tuotteet ja niiden määrät. Määriä voi muokata tässä näkymässä.



Fidelix nettikauppa

Kirjaudu ulos | Tyhjä kori

Tervetuloa, Frånçöiseke Vändermèülène !
Ostoskorisi sisältö:

Tuote	Hinta	Määrä	Yhteensä	Muuta määrää
FX-2025a basic	410.00 €	1	410.00 €	1 päivitä poista
FX-2025a rev. 2	640.00 €	1	640.00 €	1 päivitä poista

Kokonaishinta: 1050 €

Kuva 8. Kuvankaappaus ostoskorista.

Cart-taulukko käydään läpi. \$Key sisältää tuote Id:n ja \$value määrän, kuinka monta kappaletta koriin on lisätty.

```
foreach($_SESSION['cart'] as $key => $value)
    $sql = mysql_query("SET CHARACTER SET 'utf8'") ;
    $usergroupquery=mysql_query(" SELECT groupNUMBER FROM usergroup
    WHERE groupName='$userselected' ")

or die(mysql_error()) ;
$usergroupnumber=mysql_fetch_array($usergroupquery) or die(mysql_error()) ;
$result=mysql_query(" SELECT * FROM product WHERE prodID='$key' ") or die(mysql_error());

$prodNAME=mysql_result($result,0,"prodNAME");
```

Koodiesimerkki 8. Ostoskorin sisällön haun logiikka.

Ostotapahtuman lopuksi on tilauksen lähettäminen- ja viimeistely-sivu. Tällä sivulla voi lisätä tilausvahvistussähköpostille vastaanottajia ja lisätä huomautuksia tilaukseen liittyen. Asiakkaan lisäämät kommentit tallennetaan Mysql-kantaan ja lähetetään sähköpostilla Fidelixille.

Tämä on sisältö ostoskoriisi.
Tarkista tilaus ja tee lisämerkintöjä.

FX-2025a basic	1	410.00 €	kommentteja tähän tuotteeseen...
FX-2025a rev. 2	1	640.00 €	kommentteja tähän tuotteeseen...
Kokonaishinta:		1050 €	huomautuksia koko tilaukseen...

Toimitusosoite:

Laskutusosoite:
(Voit lisätä osoitteita "Omat tietoni")

Vitteesi tähän tilaukseen:
(Saat meiltä tilausnumeron tilausvahvistuksen yhteydessä)

Sähköposti kopioiden vastaanottajat
(Anna lisä-sähköpostiosoitteet joihin myös lähetetään tilauksesta sähköposti, erota osoitteet puolipisteellä ;
Ilman tyhjää välejä osoitteiden välissä)
Invalid email address.

Tehdessäsi tilauksen, saat vahvistuksen sähköpostitse sähköpostiosoitteeseen "Omat tietoni"

Kuva 9. Kuvankaappaus checkout-sivusta. Kuvassa annettu virheellinen sähköposti jolloin se korostetaan punaisena.

5.3 Admin-sivusto

Admin-sivusto kehitettiin korvaamaan Phpmyadmin-sql-ylläpitojärjestelmä ja mahdollistamaan sivuston ylläpito ilman manuaalista SQL-kyselyiden kirjoittamista. Admin-sivustolla ei ole SSL- tai muuta erityistä suojausta, mutta sisäänkirjautuminen/sivustolle ohjaus onnistuu vain Fidelixin sisäisestä verkosta. Parempaa suojausta ja yhteyden muodostamista internetin kautta on suunniteltu, ja ne luultavasti lisätään myöhempiin versioihin, jos tarve vaatii.

Käyttäjät-välilehdellä voi luoda, kopioida ja muokata käyttäjiä. Käyttäjille annetaan alennukset myös tällä sivulla.

The screenshot shows the 'Käyttäjät' (Users) section of the 'Fidelix nettikauppa - muokkaustilassa' (Fidelix online store - edit mode) interface. On the left is a sidebar with navigation links: 'Tuotteet', 'Käyttäjät' (selected), 'Tuoteryhmät', 'Alaluokat', 'Tilauksien tilanne', 'Tuotteiden tila:', and 'Omat tietoni'. The main content area has the title 'Fidelix nettikauppa - muokkaustilassa' and a subtitle 'Muokkaa nyt: Käyttäjät'. A 'Kirjaudu' (Login) button is in the top right. Below the title, a blue banner reads 'Valitse käyttäjä jonka alennukset kopioidaan, new user luo tyhjän käyttäjän:'. A dropdown menu shows 'New user' and a 'Valitse' button. Below this, two user entries are listed. The first entry is for 'Antti Koskinen' with email 'the.antti.koskinen@fideli', username 'antti', and password 'Fidelix'. The second entry is for 'Frånçöiseke Vändermööle' with email 'reino.ederveen@fidelix.fi', username 'reino', and password 'Fidelix'. Both entries have a '(käyttäjätunnus)' label and a 'FI' or 'EN' language dropdown. To the right of each entry are buttons for 'Ota muutokset käyttöön' (Apply changes), 'Poista' (Delete), and 'Alennukset' (Discounts).

Kuva 10. Kuvankaappaus asiakkaiden hallinnasta

Tuotteen lisäämiseksi tuote valitaan tiputusvalikoista ja painetaan valitse-painiketta. Valitun tuotteen nimi ilmestyy alempaan "Lisää uusi alennus"- kenttään ja tuotteen id piilotettuun id-kenttään. Id-kenttä on piilotettu väärin nimi-id-parien muodostumisen estämiseksi. Tiputusvalikot ja alennuslaskut on toteutettu myös Ajaxilla, koska sivun uudelleen lataamisen jälkeen kentät joutuisi täyttämään uudelleen. Lopuksi tallenna-painikkeella alennukset tallennetaan Mysql-kantaan.

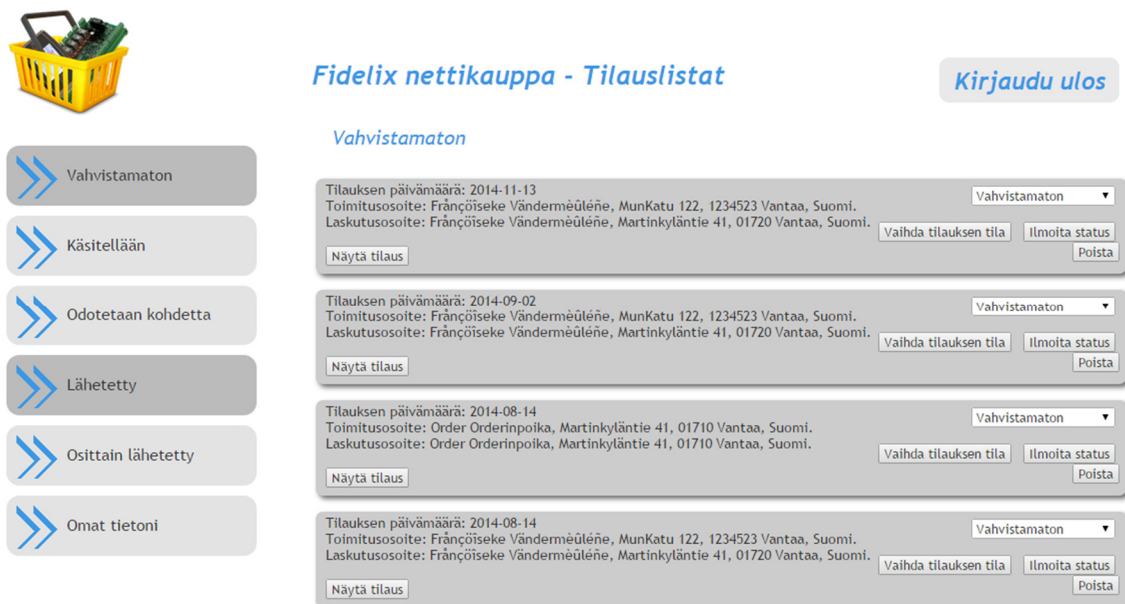
Kuva 11. Kuvan 20 "alennukset"-painikkeen painamisen jälkeen saadaan kuvan 15 näkymä.

Tuotteet-välilehdellä hallinnoidaan tuotteita. Tuotteita voi lisätä, poistaa, muokata ja tuotekohtaisia alennuksia voi myös tarkastella ja muokata tässä näkymässä. Jos tuotteen alennus on 100 %, sitä ei näytetä verkkokaupassa.

Kuva 12. Kuvankaappaus tuotteiden hallinnasta. Ylemmässä tuotteessa on painettu "Annetut alennukset" -painiketta.

5.4 Order-sivusto

Order-sivustolla listataan kaikki saadut tilaukset lajiteltuina nykyisen tilan mukaan. Tilauksen tilaa voi muuttaa ja tilanmuutoksesta voidaan lähettää asiakkaalle sähköposti-ilmoitus. Tilauksen tuotteille on myös mahdollista määritellä erikseen tilat, kuten ”loppunut”, ”tilattu” jne. Ilmoitus sähköposti luodaan jälleen Pearin sähköposti-funktiolla, kuten tilaustakin tehdessä.



Fidelix nettikauppa - Tilauslistat Kirjaudu ulos

Vahvistamaton

Tilauksen päivämäärä: 2014-11-13
 Toimitusosoite: Frånçöiseke Vändermëülëne, MunKatu 122, 1234523 Vantaa, Suomi.
 Laskutusosoite: Frånçöiseke Vändermëülëne, Martinkyläntie 41, 01720 Vantaa, Suomi.

Tilauksen päivämäärä: 2014-09-02
 Toimitusosoite: Frånçöiseke Vändermëülëne, MunKatu 122, 1234523 Vantaa, Suomi.
 Laskutusosoite: Frånçöiseke Vändermëülëne, Martinkyläntie 41, 01720 Vantaa, Suomi.

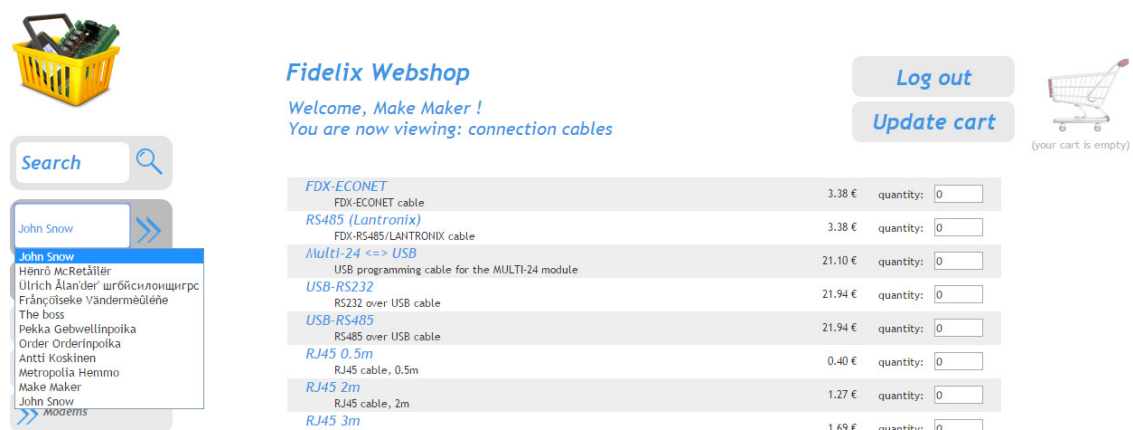
Tilauksen päivämäärä: 2014-08-14
 Toimitusosoite: Order Orderinpoika, Martinkyläntie 41, 01710 Vantaa, Suomi.
 Laskutusosoite: Order Orderinpoika, Martinkyläntie 41, 01710 Vantaa, Suomi.

Tilauksen päivämäärä: 2014-08-14
 Toimitusosoite: Frånçöiseke Vändermëülëne, MunKatu 122, 1234523 Vantaa, Suomi.
 Laskutusosoite: Frånçöiseke Vändermëülëne, Martinkyläntie 41, 01720 Vantaa, Suomi.

Kuva 13. Kuvankaappaus tilaukset-näkymästä.

5.5 MakeOrder-sivusto

Makeorder-sivusto on Fidelixin työntekijöille tehty järjestelmä, jonka kautta voidaan tehdä tilauksia asiakkaiden puolesta. Sivuston toiminnallisuus on muuten sama kuin julkisessa kauppa-sivustossa, asiakkaan valintaa lukuunottamatta. Valittu asiakas tallennetaan sessioon. Tilausta tallennettaessa tietokantaan tilauksesta tallennetaan erikseen asiakas-id ja tilauksen tekijän id. Normaalilla kauppa-sivustolla asiakas- ja tilauksen tekijä -id:t ovat samat.



Fidelix Webshop
Welcome, Make Maker!
You are now viewing: connection cables

[Log out](#)
[Update cart](#)

(your cart is empty)

FDX-ECONET			
FDX-ECONET cable	3.38 €	quantity:	0
RS485 (Lantronix)			
FDX-RS485/LANTRONIX cable	3.38 €	quantity:	0
Multi-24 <=> USB			
USB programming cable for the MULTI-24 module	21.10 €	quantity:	0
USB-RS232			
RS232 over USB cable	21.94 €	quantity:	0
USB-RS485			
RS485 over USB cable	21.94 €	quantity:	0
RJ45 0.5m			
RJ45 cable, 0.5m	0.40 €	quantity:	0
RJ45 2m			
RJ45 cable, 2m	1.27 €	quantity:	0
RJ45 3m			
RJ45 cable, 3m	1.69 €	quantity:	0

Kuva 14. Kuvankaappaus tilauksenluonnista. Vasemmalla lista asiakastileistä.

6 Jatkokehitys

Raportin kirjoitushetkellä 13.4.2015 verkkokaupan käyttöönotto on aloitettu ja ensimmäiset demo-kokeilut on tehty.

Käyttäjiltä ei ole tullut toiminnollisuuden kannalta toiveita. Kun verkkokaupan käyttöä lisätään ja se otetaan käyttöön yrityksen ulkopuolella, esimerkiksi vientiin, niin sql-kyselyiden turvallisuutta tullaan parantamaan kirjoittamalla ne uudestaan prepared statement -menetelmällä.

Myös tilauksen tekijöille lisätään mahdollisuus tehdä valmiita tuotepaketteja, jotka tallentuvat tietokantaan ja tuotelistaan tuotepaketit-kategoriaan, uudelleen käytettäväksi.

7 Yhteenveto

Mielestäni projekti oli erittäin opettavainen, sillä siinä pääsi tutustumaan eri tekniikoihin. PHP ja SQL olivat entuudestaan tuttuja, mutta Ajax oli uutta ja sen opettelu olikin hyvin mielenkiintoista. Projektia toteuttaessa näki myös, miten eri tekniikat toimivat yhteen, esimerkiksi Ajaxista muuttujien tietoja php:lle ja toisinpäin.

Verkkokauppa otettiin testaukseen vuoden 2015 huhtikuussa. Ensimmäisen perehdytyksen aikana havaittiin, että verkkokaupassa ylläpidon eikä asiakkaan sivustoilla funktiot ja sql-kyselyt toimineet. Myöhemmin tutkimisen jälkeen selvisi, että jokin päivitys oli muuttanut hakemistorakennetta ja palvelimen kirjoitus- ja lukuoikeuksia. Ongelman korjaamisen jälkeen verkkokauppa toimi moitteetta. Toistaiseksi käyttäjät ovat olleet tyytyväisiä.

Verkkokauppa todennäköisesti tulee tarvitsemaan vielä jatkokehitystä, joten se tulee työllistämään jatkossakin. Sillä suunnitteilla on esimerkiksi mahdollisuus maksamiseen suoraan verkkokaupan kautta.

Lähteet

- 1 Scrum.org, [verkkosivu], <https://www.scrum.org/resources/what-is-scrum> [5 Huhtikuu 2015].
- 2 Php.net dokumentaatio [verkkosivu] <http://php.net/> [19 Huhtikuu 2015].
- 3 Oracle Sql dokumentaatio ja manuaalit [verkkosivu] <http://www.oracle.com/technetwork/database/database-technologies/sql/overview/index.html> [19 Huhtikuu 2015].
- 4 w3schools manuaali [verkkosivu] http://www.w3schools.com/Ajax/ajax_intro.asp
jQuery dokumentaatio [verkkosivu] <http://api.jquery.com/jquery.ajax/>.
- 5 CKeditor. <http://ckeditor.com/>.
- 6 Ajax kaavio howes-it-going.com