

INTERNET-MUSIIKKIKAUPAN HALLINNOINTISIVUSTO

Arttu Huotari
2011
Oulun seudun ammattikorkeakoulu

INTERNET-MUSIIKKIKAUPAN HALLINNOINTISIVUSTO

Arttu Huotari
Opinnäytetyö
28.2.2011
Tietotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

Koulutusohjelma	Opinnäytetyö	Sivuja	+	Liitteitä
Tietotekniikan koulutusohjelma	Insinööriyö	39	+	0
Suuntautumisvaihtoehto	Aika			
Sulautetut järjestelmät	2011			
Työn tilaaja	Työn tekijä			
Hitmedia Oy	Arttu Huotari			
Työn nimi				
Internet-musiikkikaupan hallinnointisivusto				
Avainsanat				
HTML, PHP, SQL, WWW-kauppa				

HitCaster on oululaisen Hitmedian elokuussa 2009 julkistama WWW-musiikkikauppa. Sivuston laajetessa kävi ilmeiseksi, että tarvitaan työkalu helpottamaan kaupan ylläpitoa ja hallintaa.

Insinööriyössä suunniteltiin ja toteutettiin nettikaupalle WWW-pohjainen hallintasivusto. Tarkoituksena oli kehittää helppokäyttöinen ja monipuolinen hallintaohjelmisto, jonka avulla ylläpitäjät pystyvät nopeasti ja tehokkaasti valvomaan ja hallitsemaan nettikaupan toimintaa.

Työ käynnistettiin suunnitteluvaiheella, jonka aikana luotiin sivustolle vaatimusmäärittely, hahmoteltiin ulkoasua sekä luotiin alustava valikkorakenne. Tätä seurasi kehitysvaihe, jossa luotiin sivuston ulkoasu HTML:n ja CSS:n avulla, sekä toiminnallisuus PHP:tä ja PostgreSQL-tietokantapalvelinta käyttäen. Lopuksi sivusto siirrettiin testipalvelimelle koekäyttöön, jonka aikana korjattiin ilmenneet virheet.

Sivuston suunnittelussa otettiin huomioon myös laajennettavuus, sillä oli tiedossa että nettikauppaa tulnaisiin jatkossa laajentamaan. Loppuvuodesta 2010 kauppaan lisättiin myyntiin karaoke sekä äänikirjat. Kehitetyn sivuston todettiin vastaavan vaatimusmäärittelyä, ja se luovutettiin käyttöön syksyllä 2010.

Degree programme	Thesis	Number of pages	+	Appendices
Information Technology	Bachelor's thesis	39	+	0
Line	Date			
Embedded systems	2011			
Commissioned by	Author			
Hitmedia Oy	Arttu Huotari			
Thesis title				
Web store administration site				
Keywords				
HTML, PHP, SQL, Web store				

HitCaster is a music web store launched on August 2009 by the Oulu-based mobile software company Hitmedia. As the web store grew in features, it became apparent that the store required an efficient administration tool to ease the upkeep and monitoring of the site.

The purpose of this thesis work was to design and implement a web-based administration site to the web store for Hitmedia. The goal was to develop an easy-to-use and comprehensive tool with which the administrators of the web store could quickly and efficiently monitor and maintain the operation of the site.

The work began with a planning phase, during which a requirement specification, a draft layout of the site and a preliminary menu structure were created. This was followed by a development phase, during which the layout and functionality of the site was implemented with HTML, CSS, PHP and SQL using the PostgreSQL database server. Finally the site had a trial run in a test environment, after which all the found bugs were fixed.

The site was designed to be easily extended, as it was known that the web store would be enhanced with further features in the future. The developed site was deemed to comply with the requirement specification, and was handed over to the client of the thesis work on the fall of 2010.

SISÄLTÖ

TIIVISTELMÄ.....	3
ABSTRACT.....	4
SISÄLTÖ.....	5
1 JOHDANTO	6
2 WWW-KEHITYSTYÖKALUT	7
2.1 Palvelinkone	7
2.2 WWW-palvelinohjelmisto	8
2.3 HTML-kieli	8
2.4 Dynaamisten WWW-sivujen luominen	11
2.4.1 ASP.NET	12
2.4.2 PHP	13
2.5 Tietokanta	14
2.5.1 Tietokannan hallintajärjestelmät	16
2.5.2 SQL-kyselykieli	17
3 TYÖSSÄ KÄYTETYT MENETELMÄT	20
3.1 Linux-serverikone	20
3.2 Apache-WWW-palvelin	20
3.3 WWW-ympäristön ohjelmointikielät	21
3.3.1 PHP	21
3.3.2 JavaScript	21
3.4 PostgreSQL-tietokantapalvelin	22
4 HITCASTER-NETTIMUSIIKKIKAUPPA.....	23
5 HITCASTER-NETTIKAUPAN HALLINNOINTISIVUSTO.....	25
5.1 Hallinnointisivuston suunnittelu	25
5.1.1 Vaatimusmäärittely	25
5.1.2 Valikkorakenne	26
5.2 Hallinnointisivuston toteutus.....	27
5.3 Hallinnointisivuston testaus	35
6 YHTEENVETO	36
LÄHTEET	38

1 JOHDANTO

Musiikin ja muun viihteen hankkimisesta internetin kautta on 2000-luvulla tullut suosittu ilmiö. Opinnäytetyön tilaajana toimiva Hitmedia Oy on vuonna 2002 perustettu mobiilialan ohjelmistoyritys, jonka päätuotteina ovat musiikin ja muun materiaalin toistoon kehitetyt mobiilisovellukset.

Hitmedian soittimet käyttävät omia, yrityksen sisällä kehitettyjä audio- ja videokoodikkeja, ja soittimet toistavat vain näillä koodikkeilla pakattua sisältöä. Siksi riittävä sisällön saatavuus on ratkaiseva tekijä soitinten suosiossa.

Yksi käytetyistä kanavista on Hitmedian oma nettikauppa HitCaster, joka tällä hetkellä tarjoaa sisältöä musiikkisoittimelle albumien muodossa. HitCasterin toimintaperiaatteena on, että artisti itse, tai esimerkiksi levy-yhtiön edustaja voi rekisteröityä sivustolle, ja rekisteröitymisen jälkeen voi käyttää sivuston työkaluja ladatakseen musiikkiaan myyntiin.

Monimutkainen nettikauppa tarvitsee myös laajan hallintatyökalun, jolla kyetään tarkistamaan ja hyväksymään uudet rekisteröitymiset ja myyntiin ladatut albumit sekä suorittamaan muut tarpeelliset ylläpitotehtävät vaivattomasti ja nopeasti. HitCasteria luotaessa ja kehitettäessä on kasattu alkeellinen ylläpitotyökalu akuuteimpia tehtäviä varten. Sen rajallisuudet tulevat kuitenkin hyvin ilmeisiksi hallittavan sisällön määrän kasvaessa.

Opinnäytetyön aiheena onkin suunnitella ja toteuttaa HitCasteriin riittävän laaja ja tehokas hallintatyökalu. Tavoitteena on helpottaa nettikaupan ylläpitoa sekä olennaisesti vähentää sen viemää aikaa ja työtaakkaa.

2 WWW-KEHITYSTYÖKALUT

Yksinkertaisten WWW-sivujen, jotka eivät sisällä dynaamisesti muuttuvaa tietoa, ylläpitoon tarvitaan yksinkertaisimmillaan vain palvelinkone, jossa ajetaan WWW-palvelinohjelmistoa. Käytännössä nykyisin kuitenkin halvimmatkin palveluntarjoajat tarjoavat tukea vähintään jollekin dynaamisten sivujen luomiseen käytetylle kielelle. Usein myös tietokannan saa käyttöönsä ainakin lisämaksua vastaan.

2.1 Palvelinkone

WWW-palvelimen tukijalka on fyysinen PC-palvelinkone, jolla ajetaan halutun käyttöjärjestelmän alla valittua WWW-palvelinta sekä mahdollisesti tietokantapalvelinta. Nykyisin erityisesti palvelinkoneita vuokraavien yritysten keskuudessa on suosittua käyttää myös virtualisointia, jolloin yksi fyysinen PC-palvelin sisältää useita virtuaalisia palvelinkoneita, jotka kaikki ajavat omaa käyttöjärjestelmäänsä ja ovat täysin riippumattomia toisistaan.

Palvelinkoneen voi vuokrata, jolloin kone sijoitetaan fyysisesti palveluntarjoajan tiloihin ja konetta hallitaan etähallintatyökaluilla. Nettiyhteys jaetaan samoissa tiloissa olevien muiden palvelinten kanssa, ja liikenteen maksimimäärä voi olla kuukausittain rajoitettu. Etuna vuokraamisesta on, että vuokraaja voi antaa palveluntarjoajan huolehtia osasta palvelimen ylläpitoa, kuten tietojen varmuuskopioinnista sekä varavirran tarjoamisesta sähkökatkojen aikana UPS-laitteiden avulla.

Palvelinkoneen voi myös ostaa itse ja sijoittaa omiin tiloihin. Tällöin kaikki ylläpito ja vikatilanteisiin varautuminen jää luonnollisesti palvelimen hankkijan omaksi tehtäväksi. Etuina ovat muun muassa että palvelin saadaan varmasti sijoitettua turvallisiin tiloihin, jolloin fyysiset murtautumisyrietykset käyvät hankaliksi, sekä koneen hallittavuus myös tilanteissa, joissa etähallintatyökalut eivät jostain syystä ole käytettävissä.

2.2 WWW-palvelinohjelmisto

WWW-palvelinohjelmisto on palvelinsovellus, joka jakaa dokumentteja HTTP-protokollan avulla asiakasohjelmille ja koneille. HTML-kielellä luodut dokumentit muodostavat WWW-sivuja, jotka voidaan näyttää asiakaskoneella WWW-selaimen avulla. (1.)

Erilaisia palvelinohjelmistoja on tuhansia erilaisia, mutta ylivoimaisesti suosituimmat ovat Apache Foundationin Apache HTTP Server, sekä Microsoftin Internet Information Services, jotka yhdessä vastaavat yli kolmesta neljäsosasta maailmanlaajuisesti käytetyistä palvelinohjelmistoista. (2.)

Apache HTTP Server -palvelinohjelmisto on saatavissa useille käyttöjärjestelmille, myös Unixeille (BSD:t, Linuxit) sekä Windowseille. Pelkkä Apache tarjoaa tuen ainoastaan staattisten tiedostojen jakamiselle HTTP-protokollan ylitse. Sitä voidaan kuitenkin täydentää useilla moduuleilla, joiden avulla Apacheen voidaan lisätä tuki muun muassa dynaamisten sivujen luontiin tarvittaville kielille, kuten Perl, Python, Tcl ja PHP. (3.)

Internet Information Services on saatavissa vain Microsoftin Windows-käyttöjärjestelmille. IIS tukee dynaamisten WWW-sivujen luomiseen oletuksena Microsoftin omaa ASP.NET-ohjelmointiympäristöä. Siihen kuitenkin on asennettavissa tuki useimmille Apachellekin saatavilla oleville kielille. (4.)

2.3 HTML-kieli

HTML-kuvauskieli on WWW-sivujen tukijalka. Sekä dynaamiset että staattiset WWW-sivut koostuvat käyttäjän selaimen saapuessaan miltei kokonaan HTML-kielestä, jonka selain tulkkaa käyttäjälle näkyväksi

nettisivuksi. HTML-standardia ylläpitää vuonna 1994 perustettu kansainvälinen yritysten ja yhteisöjen yhteenliittymä World Wide Web Consortium, lyhyesti W3C.

Uusimman 4.01-version rinnalle on muokattu myös XML:n muotovaatimukset täyttävä XHTML 1.0. Nämä eroavat toisistaan lähinnä XHTML:n tiukemmilla muotosäännöillä, kuten pakollisella pienten kirjainten käytöllä kaikissa tunnisteissa sekä attribuuttien arvojen pakollisella merkitsemisellä lainausmerkein.

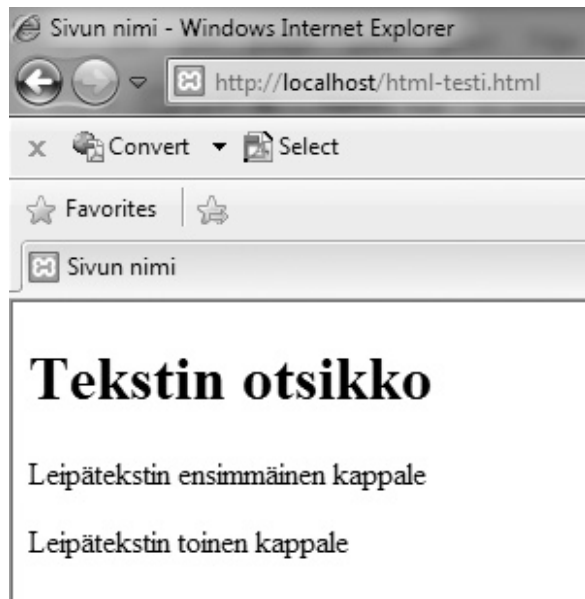
HTML-kieli rakentuu pääasiallisesti elementeistä ja näiden attribuuteista sekä sisällöstä. Esimerkiksi elementti

```
<a href="http://www.oamk.fi">OAMK Etusivu</a>
```

koostuu aloitustunnisteesta <a>, sen sisältämästä attribuutista href, attribuutin arvosta "http://www.oamk.fi", elementin sisällöstä "OAMK Etusivu" sekä lopetustunnisteesta . Selaimen tulkkamana kyseinen rivi näyttää sivulla tekstin "OAMK Etusivu" hyperlinkkinä, jota klikkaamalla selain siirtyy href-attribuutin osoittamalle sivulle.

ESIMERKKI 1. Yksinkertainen XHTML-sivu

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
<link rel="stylesheet" type="text/css" href="style.css" />
<title>Sivun nimi</title>
</head>
<body>
<h1>Tekstin otsikko</h1>
<p>Leipätekstin ensimmäinen kappale</p>
<p class="kursiivi">Leipätekstin toinen kappale</p>
</body>
</html>
```

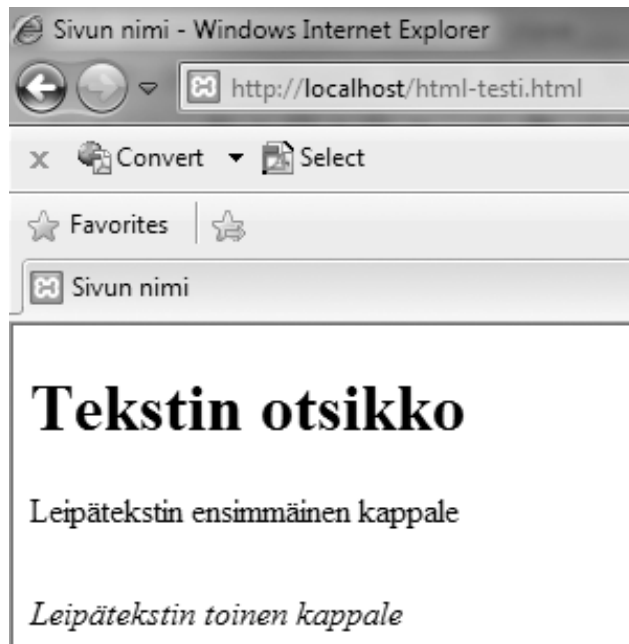


KUVA 1. Esimerkin 1 mukainen HTML-sivu selaimessa

W3C suosittelee erottamaan sivun ulkoasu- ja asettelumäärittelyt erilliseen CSS-tyyliohjetiedostoon. Voidaan sanoa, että CSS-tiedosto kertoo, miltä sivun tulisi näyttää käyttäjälle, ja HTML-tiedosto kertoo, mitä sisältöä näytetään. Ulkoasun ja sisällön erottaminen yksinkertaistaa HTML:ää sekä helpottaa sivujen luomista ja päivittämistä. Esimerkin 1 sivu lukeekin style.css-nimisen CSS-tiedoston. Mikäli esimerkiksi leipätekstin kappalevälin korkeutta haluaisi säätää ja toisen kappaleen tekstin muuttaa kurssiiviksi, onnistuisi se lisäämällä tiedostoon esimerkin 2 mukainen määrittely, joka asettaa p-elementtien alapuolisen marginaalin 30 pikseliin ja kurssiiviluokkaisen p-elementin tekstisisällön kurssiiviksi. Kuvat 1 ja 2 havainnollistavat, miltä esimerkin 1 sivu näyttää ennen ja jälkeen CSS-määrittelyjen käyttöönoton.

ESIMERKKI 2. Esimerkki CSS-määrittelystä

```
p {  
    margin-bottom: 30px;  
}  
  
p.kursiivi {  
    font-style:italic;  
}
```



KUVA 2. Esimerkin 1 HTML-sivu esimerkin 2 CSS-määrittelyillä muotoiltuna selaimessa

2.4 Dynaamisten WWW-sivujen luominen

Dynaamisilla WWW-sivuilla tarkoitetaan sivuja, jotka muodostetaan muuttuvista tiedoista. Staattisia sivuja voidaan säilyttää palvelimella sellaisinaan HTML-kielisinä, jolloin WWW-palvelinohjelmisto yksinkertaisesti lähettää tiedoston HTTP-protokollan kautta käyttäjälle. Staattiset sivut muuttuvat vain, mikäli tiedostoa muokataan palvelimella.

Dynaaminen sivu taas luodaan vasta, kun selain pyytää sitä palvelimelta. Pyyntö käynnistää WWW-palvelinohjelmistossa toimintoja, jotka johtavat uuden sivun luomiseen ja lähettämiseen selaimelle. Tyypillisesti tämä merkitsee pyydetyn tiedoston ajamista valitun ohjelmointikielitulkin läpi, joka tulostaa HTML-muotoisen tuloksen lähetettäväksi selaimelle.

Dynaamisesti luotujen sivujen etuna on, että sivujen sisältö voi riippua vaikkapa kellonajasta tai käyttäjän antamista parametreista. Selaimelle ei ole merkitystä onko sivu dynaaminen vai staattinen, sillä saapuessaan selaimen molemmat rakentuvat HTML-kielestä. Käytännössä nykyisin kaikki suositut ja monimutkaiset sivustot ovat dynaamisia.

Dynaamisten sivujen luomiseksi täytyy valita ohjelmointikieli, jota valittu WWW-palvelinohjelmisto tukee joko suoraan tai asennettavilla lisämoduuleilla.

Ehdottomasti suosituimmat ohjelmointiympäristöt dynaamisille WWW-sivuille ovat PHP Groupin PHP, jota käytetään arviolta kolmasosassa WWW-palvelimista, sekä Microsoftin ASP.NET, jonka käyttäjiä on arviolta 23 %. (5.)

Valinta ympäristöjen välillä riippuu pitkälti yrityksen sisällä aiemmin käytetyistä tekniikoista, budjetista sekä sivuston arvioidusta monimutkaisuudesta sekä käyttötasosta. ASP.NETillä voi saavuttaa paremman suorituskyvyn suurilla ja monimutkaisilla sivustoilla, mutta se miltei varmasti sitoo yrityksen käyttämään Microsoftin IIS-WWW-palvelinta, sillä huolimatta ASP.NETin muutamista käännöksistä eri alustoille, sitä suositellaan tuotantoympäristössä käytettävän vain IIS:n yhteydessä. (6.) Tämä pakottaa hankkimaan palvelinkoneeseen Windows-pohjaisen käyttöjärjestelmän, joka maksullisena nostaa kuluja.

PHP on ilmainen ratkaisu, joka pyörii vakaasti myös avoimen lähdekoodin Linux-käyttöjärjestelmiin asennettujen Apache-WWW-palvelinten yhteydessä. Useiden muiden WWW-ympäristössä käytettyjen ohjelmointikielten tapaan PHP häviää ASP.NETille jonkin verran nopeudessa, sillä PHP-koodi tulkitaan joka kerta uudelleen selaimen pyytäessä PHP-sivua. ASP.NET taas tulkaa koodin vain kerran, ja säilyttää tulkatun koodin DLL-tiedostossa palvelimella, kunnes havaitsee lähdekoodin muuttuneen ja tulkaa koodin uudelleen. (4.) Käyttäjälle tämä ero näkyy viiveenä vasta todella vaativissa tai kuormittuneissa sivustoissa.

2.4.1 ASP.NET

ASP.NET ei itsessään ole ohjelmointikieli, vaan ohjelmointiympäristö, joka sallii Microsoftin .NET Frameworkin tukemien kielten käyttämisen WWW-sivujen luomisessa. Tavallisin WWW-ympäristössä käytetty kieli on Visual

Basicin skriptimuunnos VBScript. VBScriptiä on mahdollista ajaa myös käyttäjän selaimessa JavaScriptin tapaan, mutta vain Microsoftin oma Internet Explorer tarjoaa sisäänrakennetun tuen tälle.

ASP.NETin ensimmäinen versio julkaistiin vuonna 2002 samaan aikaan Microsoftin Visual Studio .NET -ohjelmankehitysympäristön kanssa. ASP.NETin edeltäjän, ASP Classiciksi nimetyn ympäristön ensimmäinen versio julkaistiin jo vuonna 1996.

ASP.NETin etuja ovat valmiiksi käännettyjen sivujen tuoma nopeus sekä mahdollisuus käyttää ennestään tuttuja .NET-kieliä WWW-ympäristössä. Kun selain pyytää palvelimelta uuden tai muuttuneen ASP.NET-sivun, palvelin kääntää sen DLL-tiedostoksi, jota säilytetään ja ja ajetaan jatkossa, kunnes sivun lähdekoodin havaitaan muuttuneen. Koodia ei siis turhaan tulkita joka kerta, kun käyttäjä pyytää sivua, kuten esimerkiksi PHP:ssa tai ASP Classicissa. (4.)

Tästä voi olla haittana, että ensimmäinen sivun lataaja saattaa havaita viivettä sivun latautumisessa. Tämä voidaan kuitenkin kiertää kääntämällä koodi ennen sivujen asettamista julkisiksi. Tästä on etuna myös se, ettei sivujen lähdekoodia tarvitse säilyttää internetiin näkyvällä palvelimella.

2.4.2 PHP

PHP on vuonna 1995 julkaistu tulkittava ohjelmointikieli, joka on suunnattu ja jota käytetään erityisesti dynaamisten WWW-sivujen luomiseen. Sen uusin viitosversio on kohonnut johtavaksi WWW-ympäristön ohjelmointikieleksi; muun muassa Facebook ja Wikipedia on toteutettu PHP:lla. (7.)

PHP muistuttaa syntaksiltaan C:tä. Selkeimpänä erona on, että PHP:n muuttujat ovat heikosti tyyhitettyjä, toisin sanoen muuttujien tyyppiä ei erikseen määritellä, vaan ne voivat saada minkä tahansa arvon riippuen kontekstista. PHP:lla on myös laaja luokkakirjasto, jonka avulla sillä on

yksinkertaista käsitellä muun muassa tietokantoja, sähköpostia, kuva-, tai vaikkapa PDF-tiedostoja.

PHP on alustariippumaton kieli, joten se toimii sellaisenaan missä tahansa ympäristössä, jossa sille on saatavilla tulkki. Se on myös avoimen lähdekoodin tuote, joten periaatteessa kuka tahansa voi osallistua sen kehitykseen.

Koska PHP on tulkattava kieli, lähdekoodin sisältäviä tiedostoja säilytetään palvelimella, ja kun käyttäjän selain pyytää niitä, PHP-tulkki tulkkaa koodin ja lähettää tulkatun sivun käyttäjälle. Koodia voidaan kirjoittaa suoraan HTML:n sekaan. PHP:n aloitus- ja lopetustagit ovat `<?php` ja `?>`. Mikäli selain pyytää esimerkin 3 mukaista sivua palvelimelta, se saapuu selaimen esimerkin 4 näköisenä.

ESIMERKKI 3. PHP-tiedosto palvelimella

```
<html>
<head>
<title>PHP-sivu</title>
</head>
<body>
<?php
    echo "<p>Tämä on PHP-sivu!</p>";
?>
</body>
</html>
```

ESIMERKKI 4. PHP-tiedosto sen saapuessa selaimen

```
<html>
<head>
<title>PHP-sivu</title>
</head>
<body>
<p>Tämä on PHP-sivu!</p>
</body>
</html>
```

2.5 Tietokanta

Monimutkaiset järjestelmät, kuten laajat, dynaamiset WWW-sivut, tarvitsevat usein paikan tallentaa tietojaan. Yksinkertaisimmillaan tietoa voi

säilöä palvelimella tekstitiedostoissa. Mikäli tietoa on paljon, tai se sisältää riippuvuuksia toisiin tietoihin, relaatiotietokanta tarjoaa tehokkaan tavan säilöä, hakea ja järjestellä tietoa.

Matemaattisen pohjan relaatiotietokannoille, relaatiomallin, kehitti 1960-luvun lopulla E.F. Codd. Ensimmäiset kaupalliset toteutukset tulivat markkinoille 1970-luvun lopulla. (8.)

Tietokanta on kokoelma tietoja, joilla on yhteys toisiinsa. Relaatiotietokannassa nämä tiedot ovat jaettuina taulukoihin, joita kutsutaan tauluiksi. Taululla on tietty rakenne, ja jokainen taulussa oleva rivi sisältää rakenteen määrittelemät sarakkeet.

Column	Type	Not Null	Default
id	integer	NOT NULL	nextval('kayttajat_id_seq'::regclass)
kayttajatunnus	character varying(32)		
salasana	character varying(32)		
email	character varying(100)		
viimeisin_ip	cidr		
vieraili_viimeksi	timestamp without time zone		
kieli	character varying(4)		'fi'::character varying

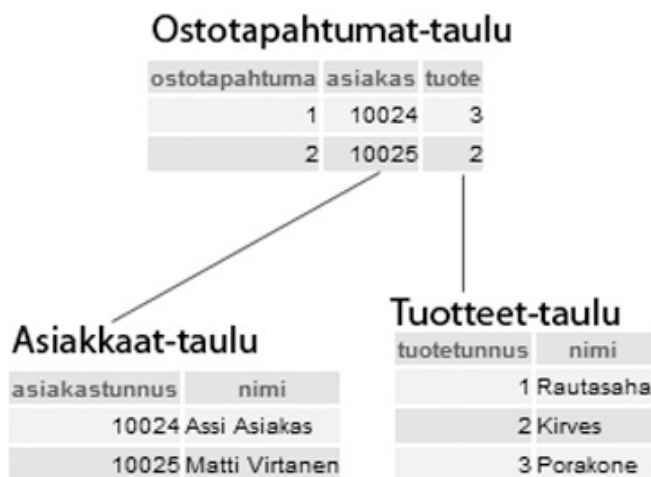
KUVA 3. Esimerkki relaatiotietokannan taulun rakenteesta

Kuvassa 3 esitellään erään esimerkkitaulun rakenne. Tauluun on mahdollista tallentaa vaikkapa kirjautumista vaativan sivuston käyttäjätunnukset. Jokaisella taululla tulee olla perusavaimeksi kutsuttu sarake, jonka avulla rivit yksilöidään. (9.) Yhdenkään rivin perusavain ei siis saa saada samaa arvoa jonkin toisen samassa taulussa olevan rivin perusavaimen kanssa. Esimerkiksi henkilötunnus on yksilöllinen, joten se kävisi perusavaimesta, olettaen että tauluun luodaan vain yksi rivi henkilöä kohden. Kuvan tapauksessa perusavaimena käytetään id-sarakkeeseen tallennettavaa juoksevaa numerointia.

Jokaisella sarakkeella on myös type-attribuutti, joka kertoo minkälaista tietoa sarakkeeseen voidaan tallentaa. Tyyppi voi olla kokonaisluku, määrätynpituisen merkkisarja tai vaikkapa päivämäärän ja kellonajan

sisältävä aikaleima. Not Null -attribuutti kertoo, saako sarake jäädä riviä luodessa tyhjäksi. Default-attribuutti taas kertoo, minkä arvon sarake saa, mikäli käyttäjä ei erikseen syötä sarakkeeseen arvoa riviä luodessa.

Toinen relaatiotietokantojen käyttämä avaintyyppi on viiteavain. Viiteavaimen avulla voidaan luoda kahden taulun välille yhteys. (9.) Kuvan 4 esimerkissä ostotapahtumat-taulu sisältää viiteavaimen sekä asiakkaat- että tuotteet-tauluihin. Näin voidaan pitää kirjaa ostetuista tuotteista sekä asiakkaista tarvitsematta kopioida samaa tietoa montaa kertaa useaan eri tauluun. Viiteavaimen ei tarvitse olla yksilöllinen saman taulun sisällä. Esimerkin tilanteessa siis sama asiakas voi tehdä useita ostoksia, tai samaa tuotetta voidaan ostaa useita kertoja.



KUVA 4. Yksinkertainen tietokanta

2.5.1 Tietokannan hallintajärjestelmät

Kaupallisesti tai avoimen lähdekoodin lisenssillä saatavilla olevat tietokantapalvelimet ovat tarkalleen ottaen tietokannan hallintajärjestelmiä (Database Management System, DBMS) eli kokoelma ohjelmistoja, jotka tukevat varsinaista tietokantamootoria sekä helpottavat ylläpitotoimenpiteitä.

Tärkein hallintajärjestelmän tehtävä on valvoa tietokannan eheyssääntöjen toteutumista. Näillä varmistetaan, että tietokannan tiedot pysyvät eheinä kaikissa tilanteissa. Hallintajärjestelmä esimerkiksi estäisi kuvan 4 tapauksessa kumman tahansa asiakkaan tietojen poistamisen tietokannasta, sillä ostotapahtumat-taulussa on viittauksia molempiin asiakkaisiin. Pystyäkseen poistamaan rivin jostain taulusta, täytyy ensin muista tauluista poistaa kaikki rivit, jotka viittaavat kyseiseen riviin.

Kaupallisista hallintajärjestelmistä suosituimmat ovat Oraclen, Microsoftin sekä IBM:n valmistamia. Avoimen lähdekoodin järjestelmistä suosituimpia ovat jokin aika sitten Oraclen hankkima MySQL, PostgreSQL sekä SQLite. (9.)

2.5.2 SQL-kyselykieli

SQL (Structured Query Language) on IBM:n kehittämä standardoitu kyselykieli, joka mahdollistaa relaatiotietokannan hallinnan suorittamalla erilaisia hakuja, lisäyksiä ja muutoksia. (10.)

Useimmin käytetyt SQL-komennot ovat SELECT, UPDATE, INSERT ja DELETE, jotka mahdollistavat tiedon hakemisen, muokkaamisen, lisäämisen ja poistamisen. SQL mahdollistaa myös uusien taulujen luomisen ja taulujen rakenteen muokkaamisen.

ESIMERKKI 5. SQL-haku SELECT-komennolla

```
SELECT kirjan_nimi
FROM Kirjat
WHERE hinta > 20.00
ORDER BY kirjan_nimi;
```

Esimerkissä 5 suoritetaan haku Kirjat-nimiseen tauluun, joka sisältää ainakin kirjan_nimi- sekä hinta-sarakkeet. Haku palauttaa niiden kirjojen nimet, joiden hinta on suurempi kuin 20,00 euroa järjestettynä kirjan nimen mukaan.

ESIMERKKI 6. Uuden rivin lisääminen tauluun

```
INSERT INTO Kirjat
```

```
(kirjan_nimi, hinta)
VALUES ('Tuntematon Sotilas', '29.90');
```

Esimerkissä 6 lisätään Kirjat-tauluun uusi rivi. Mikäli taulu sisältää kirjan_nimi- ja hinta-sarakkeiden lisäksi muitakin sarakkeita, täytyy niiden olla taulun rakenteen määrittelyssä sallittu jäävän tyhjiksi tai niillä täytyy olla jokin oletusarvo, muutoin lisäys epäonnistuu. Koska kumpikaan lisättävistä sarakkeista tuskin toimii taulun perusavaimena, voidaan olettaa että perusavainten luominen on jätetty hallintajärjestelmän vastuulle juoksevalla numeroinnilla.

Käytännössä kaikki relaatiotietokantojen hallintajärjestelmät tukevat SQL-standardin mukaista kyselykieltä. Osa hallintajärjestelmistä kuitenkin laajentaa SQL-standardia omilla lisäyksillään.

Useimpiin ohjelmointikieliin on saatavilla luokkakirjastoja, jotka mahdollistavat tiedonsiirron tietokantapalvelimen kanssa käyttäen SQL-kieltä. Muun muassa PHP:ssa luokkakirjasto on mukana oletuksena.

ESIMERKKI 7. SQL-haku PHP-sivulla

```
<html>
<head>
<title>SQL-haku PHP-sivulla</title>
</head>
<body>
<?php
pg_connect('host=localhost dbname=tietokanta user=kayttaja
password=salasana');
$haku = 'SELECT kirjan_nimi, hinta FROM Kirjat ORDER BY
kirjan_nimi';
$tulos = pg_query($haku);
if (pg_num_rows($tulos) > 0)
{
    echo '<h1>Tietokannassa olevat kirjat:</h1>';
}
while ($rivi = pg_fetch_array($tulos)
{
    echo 'Kirjan nimi: ' . $rivi['kirjan_nimi'] . ', hinta: ' .
    $rivi['hintaa'] . '<br />';
}
?>
</body>
</html>
```

Esimerkin 7 PHP-koodissa avataan yhteys PostgreSQL-tietokantapalvelimeen `pg_connect`-komennolla ja suoritetaan SQL-kysely `pg_query`-komennolla. `Pg_query` palauttaa haun tuloksen resurssimuuttujana, joka ohjataan `$tulos`-muuttujaan. Mikäli haulla löytyi tietokannasta rivejä, tulostetaan otsikko ja käydään rivit yksi kerrallaan läpi tulostaen ne sivulle. Lopuksi jokaisen rivin loppuun lisätään XHTML-kielen rivinvaihtokäsky `
`. Lopputulos voi näyttää selaimessa vaikkapa kuvan 5 mukaiselta.



KUVA 5. Esimerkin 7 PHP-sivu selaimessa

3 TYÖSSÄ KÄYTETYT MENETELMÄT

3.1 Linux-serverikone

Asiakasyrityksellä oli jo entuudestaan käytössään openSUSE-Linuxeilla varustettuja palvelinkoneita hoitamassa eri tehtäviä. Siispä kun nettikaupan palvelimeksi hankittiin tehokas moniydinprosessorilla varustettu PC-palvelin, oli luontevaa varustaa sekin Linux-käyttöjärjestelmällä. Linux-jakeluksi valikoitui Ubuntun 8.04-versio. Tämä versio valittiin, koska se oli Ubuntun uusin kahden vuoden välein julkaistavista ns. LTS (Long Term Support)-versioista, joita kehittäjä lupaa tukea päivityksin viiden vuoden ajan version julkaisusta.

Koska nettikauppa ja tämän tietokanta sijaitsevat kyseisellä palvelimella, oli järkevin ratkaisu sijoittaa myös hallintasovellus samalle palvelimelle. Näin sovellus pääsee käsiksi kaupan tietokantaan paikallisesti, käyttämättä lähiverkkoa.

3.2 Apache-WWW-palvelin

Apache on suosituin Linux-jakeluissa käytetty WWW-palvelin. Sitä pidetään vakaana, tehokkaana ja turvallisena. Vaikkei se olekaan kevyin ja tehokkain Linuxille saatavilla olevista palvelimista, sen muokattavuus, vakaus ja suuri valikoima laajennusmoduuleja tekevät siitä oivan valinnan WWW-palvelimeksi.

3.3 WWW-ympäristön ohjelmointikielet

3.3.1 PHP

Koska käyttöjärjestelmäksi valittiin Linux, ASP.NETin käyttö ohjelmointikielenä ei ollut mielekäs valinta mahdollisten vakausongelmien vuoksi. Nettikaupan toteutuskieleksi valittiin PHP, sillä se on tehokas oliopainotteinen ohjelmointikieli, joka dynaamisten WWW-sivujen luomiseen suunnattuna integroituu helposti Apache-palvelimeen. PHP:lla on myös erittäin laaja luokkakirjasto sekä netissä saatavilla oleva kattava dokumentointi, jonka avulla olemassa olevien funktioiden etsiminen ja niiden toiminnan selvittäminen on helppoa.

Nettikaupan hallinnointisovelluksen olisi voinut halutessaan toteuttaa myös eri kielellä kuin millä itse kaupan. PHP oli kuitenkin luonteva valinta myös sen luomiseen.

3.3.2 JavaScript

Nimestään huolimatta JavaScriptillä ei ole juuri mitään yhteistä Java-ohjelmointikielen kanssa. JavaScriptiä käytetään useisiin tarkoituksiin, usein luomaan WWW-sivustolle dynaamista toiminnallisuutta. JavaScript poikkeaa PHP:sta siten, että se ajetaan käyttäjän selaimessa. Mikäli käyttäjä on asettanut selaimensa olemaan ajamatta JavaScriptiä tai selain ei tue sitä (nykyselaimista käytännössä ainoastaan tekstipohjaiset selaimet), sitä ei suoriteta lainkaan. Siksi onkin hyvän ohjelmointitavan mukaista, että sivut toimivat ongelmitta myös ilman JavaScriptiä.

Sekä nettikauppa että sen hallintajärjestelmä käyttävät JavaScriptiä vain kevyisiin, sivustoja tukeviin tehtäviin, kuten jonkin linkin avaamiseen pop-up-ikkunassa tai WWW-lomakkeen sisällön oikeellisuuden tarkistukseen ennen sen lähettämistä palvelimelle. Tarkistusta ei kuitenkaan voi jättää pelkästään JavaScriptin nojaan, vaan lomakkeen tiedot tarkistetaan myös palvelimen päässä. Tässä tapauksessa JavaScriptiä käyttämällä helpotetaan lähinnä

sivuston käyttöä, sillä kun JavaScript ei salli lomakkeen lähettämistä virheellisillä tiedoilla, käyttäjä voi vain korjata virheelliset tiedot ja koettaa lähettämistä uudelleen, sen sijaan että palvelin palauttaisi virheilmoitussivun ja käyttäjä joutuisi mahdollisesti täyttämään lomakkeen kokonaan uudelleen.

3.4 PostgreSQL-tietokantapalvelin

Linuxille saatavia SQL-palvelimia on useita, näistä suosituimpina kenties MySQL sekä PostgreSQL. Asiakasyrityksen muut järjestelmät käyttivät entuudestaan PostgreSQL-tietokantapalvelinta, joten se oli looginen valinta myös nettikaupan tietokannaksi.

Koska hallintasivusto käsittelee nettikaupan tietokantaa ja tarvitsee pääsyn siihen, oli yksinkertaisinta luoda nettikaupan tietokannan yhteyteen hallintasivuston omaan käyttöön varattu taulu. Tauluun tallentuu sivuston tarvitsema tieto ylläpitäjien käyttäjätunnuksista ja näiden asetuksista.

4 HITCASTER-NETTIMUSIIKKIKAUPPA

HitCaster on elokuussa 2009 perustettu nettimusiikkikauppa, joka myy musiikkialbumia mobiililaitteille. Albumit tarjotaan Hitmedian kehittämässä tiedostomuodossa, joka suoja ne siten, että albumit voi toistaa vain tilauksen yhteydessä määritellyltä mobiililaitteelta tai muistikortilta. (11.) Syksyn ja talven 2010 aikana myyntivalikoimaan lisättiin myös karaoke sekä äänikirjat. Kuva 6 havainnollistaa, miltä HitCasterin tyypillinen musiikkialbumien etusivu näyttää.

HitCaster

Karaoke Musiikki Äänikirjat

ETUSIVU

LATAA ÄÄNIKIRJIA

LATAA ALBUMI

SOITTIMET

MIKSI KÄNNYKKÄ

MITEN HITCASTER TOIMII

UUTISKIRJE

YHTEYSTIEDOT

Musiikki

HitCaster Kännykkäalbumit on suunniteltu parantamaan musiikkielämystä matkapuhelimissa. Albumit voidaan toistaa Hitmedia Album Playerilla jonka voi ladata ilmaiseksi HitCasterista. Valikoimasta löytyy tällä hetkellä indie-musiikkia.

Lue lisää kännykkäalbumista ja HitCasterista "Miksi kännykkä" ja "Miten HitCaster toimii" -sivuilta.

HitCaster ja oululainen grungefunk-bändi Dian tarjoavat ilmaisen maistiaisena Dianin uusimman promo-albumin "In The Name Of Love".

Dianin musiikissa yhdistyy raskaampi cross-over rock-tuuttaus sekä letkeämmät ja paikoitellen äkkiväärät funkrock -groovet

Uutta HitCasterissa:

13.10.2010
Karaoke HitCasterissa

12.10.2010
HitCaster Helsingin musiikkimessuilla

08.01.2010
Tilaa HitCaster Uutiskirje

14.10.2009
Oulu Rokkaa Radio Megassa

14.10.2009
Koekuuntelumahdollisuus HitCasterissa

Haku

Hae

Artistit

ABCDEF GHIJKLMN OPQRSTUVWXYZ ÄÄ Ö Muu

Uusimmat albumit:

1,99 EUR Osta
Sound Alive
Speechless

2,99 EUR Osta
Haterial
HATERIAL - Demo
2010

1,99 EUR Osta
FALLEN
Is there hope?

1,99 EUR Osta
Korina
Underground

SINUN MUSIIKKIASI HITCASTERIIN

REKISTERÖIDY

KIRJAUDU SISÄÄN

KUVA 6. HitCasterin musiikkialbumien etusivu 24.1.2011

Musiikkiaan HitCasterin kautta myyntiin haluava artisti voi rekisteröityä HitCasteriin, minkä jälkeen hänellä on mahdollisuus ladata albumeja myytäväksi suoraan HitCasterista sekä tarkastella albumien myyntiä. Albumien lataaminen aloitetaan syöttämällä albumille nimi sekä lataamalla palvelimelle kansi- sekä valinnainen tervetulokuva, minkä jälkeen albumin raidat ladataan palvelimelle yksitellen WAV-tiedostomuodossa. Samaan aikaan syötetään tarpeelliset tiedot raidoista. Lopuksi valitaan albumin myyntihinta, hyväksytään sopimusehdot ja merkitään albumi hyväksytyksi myyntiin. Levy ei kuitenkaan vielä tässä vaiheessa ilmesty näkyville nettikauppaan.

Palvelimella automaattinen prosessi tarkistaa, onko uusia raitoja ladattu, ja jos on, käynnistää niiden enkoodauksen Hitmedian omaan tiedostomuotoon. Kun koko albumin enkoodaus on suoritettu, tarvitaan vielä ylläpitäjän tarkistus ja hyväksyntä, minkä jälkeen se lisätään automaattisesti sivustolle myyntiin.

HitCaster mahdollistaa myös tuottaja- ja ylläpitäjätunnukset. Tuottajat voivat ladata tunnuksensa alaisille artisteille levyjä, ja seurata näiden myyntiä. Pääkäyttäjät voivat tehdä tämän kaikille rekisteröityneille artisteille.

5 HITCASTER-NETTIKAUPAN HALLINNOINTISIVUSTO

5.1 Hallinnointisivuston suunnittelu

Suunnittelu käynnistettiin laatimalla vaatimusmäärittely, joka kertoo mitä ominaisuuksia hallinnointisivustolla tulee olla. Myös ulkoasua hahmoteltiin. Ohjenuoraksi valittiin, että hallinnointisivuston tulisi ulkoasultaan pääpiirteittäin muistuttaa itse nettikauppaa. Tästä poikettiin luomalla hallinnointisivusto siten, että se kykenee skaalautumaan sivusuunnassa, sillä näytettävän tiedon määrä on joissakin tapauksissa varsin laaja. Itse nettikauppa on sivusuunnassa aina samanlevyinen. Vaatimusmäärittelyn perusteella luotiin myös alustava valikkorakenne.

5.1.1 Vaatimusmäärittely

Vaatimusmäärittelyyn listattiin ominaisuudet, jotka valmiilla sivustolla tulee olla:

- uusien HitCaster-käyttäjien luominen
- käyttäjätunnuksen asettaminen tuottaja- tai ylläpitäjätunnukseksi
- uusien artistien lisääminen
- artistien liittäminen tuottajatunnusten alaisiksi
- uusien käyttäjärekisteröitymisten tarkistus ja hyväksyminen tai hylkääminen
- artistien ja tuottajien lisäämien albumien tietojen tarkistaminen ja hyväksyminen myyntiin lisättäväksi
- virhelokin tarkastelu, varmuuskopiointi ja nollaus
- postituslistan tilanneiden tarkastelu, tilaajien poistaminen ja lisääminen, sähköpostiosoitteiden helppo siirtäminen sähköpostisovellukseen
- postituslistastatistiikan näyttäminen, tilausten ja peruutusten määrä
- HitCasterissa olevien levyjen tietojen muokkaaminen
- artistien tietojen muokkaaminen

- myynnin seuranta
- myyntiraporttien luominen ja tulostaminen esim. Teostoa ja muita tahoja varten
- uusista rekisteröitymisistä ja tarkistettavista levyistä ilmoittaminen sähköpostilla valittuihin osoitteisiin
- uusien hallinnointisivuston käyttäjätunnusten luominen ja vanhojen muokkaus tai poisto
- sivun käytettävyys englanniksi tai suomeksi
- selkeä ilmoitus kirjautuneelle käyttäjälle tarkistamattomien levyjen tai käyttäjien lukumäärästä

5.1.2 Valikkorakenne

Vaatusmäärittelyn perusteella kyettiin luomaan sivuston valikkorakenne alavalikkoineen:

- Käyttäjät
 - Käyttäjät
 - Lisää käyttäjä
- Artisti
 - Artistit
 - Lisää artisti
- Levyt
- Julkaisijat
- Uutiskirje
 - Tilaaajat
 - Hallitse tilauksia
 - Statistiikka
- Seuranta
 - Myynnin seuranta
 - Seurantaraportit
- Virheloki
 - Virheloki
 - Varmuuskopioi ja tyhjennä

- Hallinta
 - Asetukset
 - Hitadmin-käyttäjät
 - Lisää käyttäjä

5.2 Hallinnointisivuston toteutus

Toteutus käynnistettiin luomalla ulkoasu ja sivuston rakenne PHP:n, HTML:n sekä CSS:n avulla. PHP:n avulla sivuista kyettiin tekemään modulaarisia. Asetukset, tietokanta- ja muut funktiot, tekstit, valikko, sekä kaikille sivuille samat sivun alussa ja lopussa olevat HTML-elementit voitiin tallentaa erillisiin tiedostoihin, jotka PHP:n avulla liitetään kaikkiin sivuston sivuihin. Näin voitiin minimoida saman koodin toistaminen turhaan useissa eri tiedostoissa. Lisäksi järjestely helpottaa muutosten tekemistä. Mikäli valikkoon halutaan tehdä muutos, tarvitsee vain muokata valikon sisältävää PHP-tiedostoa, sen sijaan että täytyisi tehdä sama muokkaus jokaiseen valikon sisältävään sivuun.

Runko luotiin siten, että PHP:n `require_once`-käskyt liittävät sivulle kaikille sivuille yhteiset osuudet, ja `if`-lause tarkistaa, onko käyttäjä kirjautunut vai ei. Kuten kuvasta 7 nähdään, etusivun tapauksessa kirjautunut käyttäjä näkee vain kehotuksen valita valikosta haluamansa toiminto, mutta käytännössä kaikki alisivut voitiin toteuttaa samalla rungolla. Jokaiselle alisivulle omat, uniikit toiminnot sisällytetään `if`-lauseeseen sisään. Tällä menetelmällä ulkoasu saatiin yhtenäiseksi ja jokaista alisivua luotaessa kyettiin keskittymään vain alisivun oman toiminnallisuuden ohjelmointiin.



KUVA 7. Hallintasivuston etusivun näkymä kirjautuneelle käyttäjälle

Kun runko oli valmis, alisivuja kyettiin lisäämään luomalla uusi PHP-tiedosto johon aluksi lisättiin `require_once`-käskyt ja tämän jälkeen lisättiin alisivukohtaiset toiminnot `if`-lauseeseen sisään. Lopuksi lisättiin valikon sisältävään PHP-tiedostoon valinta, joka johtaa kyseiselle sivulle. Sivuja ryhdyttiin kehittämään valikkorakenteen mukaisessa järjestyksessä.

HitAdmin

Käyttäjät (1) Artistit Levyt Julkaisijat Uutiskirje Seuranta Virheloki Hallinta Language / Kieli Kirjaudu ulos

Käyttäjät Lisää käyttäjä

HitCaster-käyttäjät

Tällä välilehdellä voit hallita HitCasterin käyttäjiä.

Käyttäjätunnus	Oikeudet	Artisti	Maksettu	Aktivoitu
b	Normaalikäyttäjä	b	Kyllä	Ei
a	Normaalikäyttäjä	Dian	Kyllä	Kyllä
d	Normaalikäyttäjä	Testiartisti	Kyllä	Kyllä
e	Normaalikäyttäjä	e	Kyllä	Kyllä
c	Tuottaja		Kyllä	Kyllä

25.01.2011 20:32

KUVA 8. Käyttäjät-sivu

Kuva 8 havainnollistaa käyttäjät-alasivua, jolta hallitaan nettikaupan artistitunnuksia. Kuva on otettu testiympäristöstä, jossa on käytetty lyhyitä, yhden merkin pituisia käyttäjätunnuksia nopeuttamaan kirjautumista. Pääsivu näyttää tunnusten tärkeimmät tiedot, kuten onko käyttäjä maksanut rekisteröitymismaksun, ja onko tunnus aktivoitu. Valikossa näkyvä pieni (1) osoittaa hallintasivuston käyttäjälle, että yksi uusi rekisteröityminen on hyväksymättä. Alavalikon Lisää käyttäjä -valinnasta pääsee sivulle, jolla kyetään luomaan uusia artisti-, tuottaja- tai pääkäyttäjätunnuksia nettikauppaan. Käyttäjätunnusta klikkaamalla pääsee kuvan 9 mukaiseen näkymään muokkaamaan olemassa olevia tunnuksia.

HftAdmin

Käyttäjät (1) | Artistit | Levyt | Julkaisijat | Uutiskirje | Seuranta | Virheloki | Hallinta | Language / Kieli | Kirjaudu ulos

Käyttäjät | Lisää käyttäjä

<-- Palaa

b

Käyttäjätunnus: b

Sähköposti: b@b.bb

Rekisteröitymispäivämäärä: 2010-01-08 10:56:25,96

Rekisteröitymis-IP: 127.0.0.1/32

Artistisopimuksen hyväksytty versio: 1

Kirjautui viimeksi: 2010-01-08 10:56:25,96

Viimeisin IP: 127.0.0.1/32

Kieli: Suomi

Maksettu: Kyllä

Pankki:

Tuottaja: Ei

Pääkäyttäjä: Ei

Artisti: b

Aktivoitu: Ei

Uusi salasana:

Uusi salasana uudelleen:

Hyväksy Poista

25.01.2011 20:41

KUVA 9. Olemassa olevan käyttäjätunnuksen tietojen muokkaussivu

Artistien, levyjen ja julkaisijoiden hallinta on toteutettu samalla kaavalla kuin käyttäjienkin. Pääsivu näyttää oleelliset tiedot ja alasivuilla kyetään luomaan uusia ja muokkaamaan vanhoja lisäyksiä. Tiedot haetaan tietokannasta ja muuttuneet tiedot päivitetään sinne.

Lopuilla alasivuista on jonkin verran toisistaan poikkeava rakenne. Uutiskirjeen pääsivu listaa kuvan 10 mukaisesti kaikki tällä hetkellä uutiskirjettä tilaavat sähköpostit eritellen ne suomeksi ja englanniksi tilattuihin. Teknisten syiden takia parhaaksi katsottiin, ettei uutiskirjettä voi kirjoittaa ja lähettää suoraan hallintasivustolta. Siksi sivu vain helpottaa sähköpostiosoitteiden kopioimista sähköpostisovellukseen.

HitAdmin

Käyttäjät (1) Artistit Levyt Julkaisijat **Uutiskirje** Seuranta Virheloki Hallinta Language / Kieli Kirjautu ulos

Tilaajat Hallitse tilauksia Statistiikka

HitCasterin uutiskirje

Tällä välilehdellä voit hallita uutiskirjeiden tilauksia ja peruutuksia sekä tarkastella tilausstatistiikkaa.

Postituslistan osoitteet

Suomeksi tilatut:
a@a.aa

Valitse kaikki

Englanniksi tilatut:
c@c.cc

Valitse kaikki

25.01.2011 21:07

KUVA 10. Uutiskirje-sivun pääsivu

Hallitse tilauksia -alasisivulla käyttäjä voi poistaa sähköposteja tilanneiden joukosta ja lisätä uusia, kuten kuva 11 havainnollistaa. Statistiikka-alasivu taas listaa kaikki tilaukset ja peruutukset sekä näyttää yhteenvedon lukumääristä. Kuvassa 12 statistiikka-sivun näkymä testipalvelimella.



HftAdmin

Käyttäjät (1) Artistit Levyt Julkaisijat **Uutiskirje** Seuranta Virheloki Hallinta Language / Kieli Kirjautu ulos


Tilaaajat **Hallitse tilauksia** Statistiikka

Sähköpostiosoite	Tilattu	Kieli	Poista?
a@.aa	2010-07-29 15:57:16.079	Suomi	<input type="checkbox"/>
c@c.cc	2010-07-29 14:25:10.95	Englanti	<input type="checkbox"/>

Lisää sähköpostiosoite listaan: Suomi

25.01.2011 21:14

KUVA 11. Uutiskirje-sivun tilaustenhallinta-allasivu



HftAdmin

Käyttäjät (1) Artistit Levyt Julkaisijat **Uutiskirje** Seuranta Virheloki Hallinta Language / Kieli Kirjautu ulos

Tilaaajat **Hallitse tilauksia** Statistiikka

Päivämäärä	Sähköpostiosoite	Tapahtuma
2010-01-14 13:03:34.866	a@a.aa	Tilaus
2010-01-14 13:03:55.139	a@a.aaa	Peruutus
2010-07-29 15:31:39.394	a@a.aa	Tilaus
2010-07-29 15:31:48.849	b@bb.bb	Tilaus
2010-07-29 15:44:21.871	a@a.aa	Peruutus
2010-07-29 15:44:54.82	a@a.aaa	Peruutus
2010-07-29 15:45:02.261	a@a.aa	Tilaus
2010-07-29 15:45:06.629	a@a.aa	Peruutus
2010-07-29 15:45:06.629	b@b.bb	Peruutus
2010-07-29 15:55:58.986	b@bb.bb	Peruutus
2010-07-29 15:57:16.079	a@a.aa	Tilaus
Tilauksia yhteensä	5	
Peruutuksia yhteensä	6	

25.01.2011 21:15

KUVA 12. Uutiskirje-sivun statistiikka-allasivu

Seuranta-sivut olivat sivuston työläin yksittäinen osuus toteuttaa. Alasivuja on kaksi, joista toisella tulostetaan seurantaraportteja yrityksen sisäiseen käyttöön, ja toisella raportteja ulkopuolisille tahoille, kuten Teostolle. Molemmat sivut toimivat samalla periaatteella. Ensin valitaan tulostettava raportti, aikaväli jolta tietoja raporttiin haetaan ja halutaanko tulostaa myös albumit, joilla ei ole ollut myyntiä kyseisellä aikavälillä. Raportit voidaan tallentaa CSV-muodossa, jolloin ne on helppo viedä esimerkiksi Microsoft Exceliin. Raportit tulostetaan aina englanninkielisinä.

Seurantaraportit

Tällä välilehdellä voit tulostaa erilaisia raportteja HitCasterin tilastoista.

Valitse seurantaraportti ja aikaväli

Raportti:

Aikaväli (PP.KK.VVVV):
 -
Kuluva Edellinen 3kk 6kk 12kk

Näytä myös albumit, joilla ei myyntiä:

Levykohtainen myynti

Row	Album No	Album	Artist	Album Price	Royalty Payment	Sold Albums	Total Cash Flow	Total Royalty Payment	Hitmedia Total Profit
1	2	Testialbumi	Dian	0,99	0,20	1	0,99	0,20	0,79
2	22	In The Name of Love	Dian	0,00	0,10	1	0,00	0,10	-0,10

Tulosta CSV-tiedostona

25.01.2011 21:47

KUVA 13. Seuranta-sivun yrityksen sisäisten raporttien luontiin tarkoitettu myynnin seuranta -alasivu

Kuvassa 13 on myynnin seuranta -alasivu, joka on tarkoitettu yrityksen sisäisten raporttien luomiseen. Raporteista on valittu levykohtaisen myynnin tarkastelu ja aikaväliksi vuosi 2010. Mahdollisia raportteja ovat myynnin yleinen tarkastelu, ostotapahtumien seuranta päivämäärän mukaan, statistiikka, joka listaa kaikki myyntitapahtumat, kokonaismyynnin sekä erinäisiä muita tilastoja myynnistä, maksutavat, joka listaa eri maksutavoilla

(nettipankki, luottokortti, PayPal) tehtyjen ostosten määrät, sekä levykohtainen myynti, joka tulostaa tarkempia tietoja jokaisen levyn kappalemyynnistä, rojaltimaksuista sekä kokonaistuotosta.

HitAdmin

Käyttäjät (1) Artistit Levyt Julkaisijat Uutiskirje **Seuranta** Virheloki Hallinta Language / Kieli Kirjautu ulos

Myyntin seuranta Seurantaraportit

Valitse seurantaraportti ja aikaväli

Raportti:
Teosto

Aikaväli (PP.KK.VVVV):
01.01.2010 - 31.12.2010
Kuluva Edellinen 3kk 6kk 12kk
Q1 Q2 Q3 Q4 Kuluva Edellinen

Näytä myös albumit, joilla ei myyntiä:
Kyllä

OK

Teosto

Album Catalog	Track Catalog	Artist	Track Name	Composer	Author	ISRC	Length	Publisher	Album Price	Number of sold Tracks
1	1	Dian	Testiraita 1	Testisäveltäjä	Testisanoittaja	123456789012	2:03	Artist	1,99	0
2	2	Dian	Testiraita 1	Säveltäjä			0:00	Artist	0,99	1
22	3	Dian	In The Name of Love	Testisäveltäjä	Testisanoittaja		2:05	Artist	0,00	1

Tulosta CSV-tiedostona

25.01.2011 21:58

KUVA 14. Seuranta-sivun alisivu, jolla luodaan raportit ulkopuolisille tahoille

Toisella alisivulla kyetään tulostamaan raportteja ulkopuolisille tahoille. Ohjelman luonnin aikaan raportteja tarvittiin vain Teostoa varten, mutta sovellukseen on myöhemmin mahdollista luoda uusia raportteja. Kuvassa 14 teostoraportti on tulostettu vuoden 2010 tiedoista. Raportin tiedot koostuvat Teoston vaatimista tiedoista jokaisesta Teoston alaisesta myynnistä. Raporttiin tulostetaan tiedot vain niiden albumien osalta, joihin lataaja on merkinnyt jonkin raidoista olevan Teoston alainen kappale.

Virheloki-sivuilla käyttäjä voi tarkastella nettikaupan virhelokia. Nettikauppa on ohjelmoitu siten, että virhetilanteissa tallentuu tieto päivämäärästä, käyttäjästä, virhetapahtuman sijainnista koodissa sekä muita tietoja tilanteesta. Ylläpitäjä voi näitä tietoja tarkastelemalla tutkia nettikaupan

lähdekoodia ja pyrkiä paikantamaan ongelman synnyn, ja tätä kautta korjaamaan vian. Sivuilta onnistuu myös virhelokin varmuuskopiointi ja nollaus.

Hallinta-sivuilla käyttäjä voi muokata hallintasivuston tunnuksensa asetuksia vaihtamalla sähköpostiosoitettaan tai salasanaansa. Lisäksi käyttäjä voi tilata kerran vuorokaudessa lähetettävän ilmoituksen uusista rekisteröitymisistä tai ladatuista albumeista. Mikäli käyttäjän on sallittu hallita myös muita hallintasivuston tunnuksia, on sivuilla mahdollista myös muokata muiden käyttäjien asetuksia sekä luoda uusia hallintatunnuksia.

5.3 Hallinnointisivuston testaus

Kehitysprosessin valmistuttua jäljellä oli vielä sivuston testaus mahdollisten virheiden ja ongelmien löytämiseksi ja korjaamiseksi. Testaus suoritettiin testipalvelimella, johon oli asennettu myös nettikaupan testausversio. Testaajina toimivat Hitmedian työntekijät, joiden käyttöön sovellus tulisi päätymäänkin.

Testausjakso kesti noin viikon, minkä alussa testaajia ohjeistettiin kokeilemaan sivuston kaikkia ominaisuuksia ja raportoimaan virheellisestä toiminnasta. Jakson aikana suuria ongelmia sivuston toiminnassa ei havaittu. Käyttäjät kuitenkin sekoittivat usein käyttäjät-sivun nettikaupan käyttäjätunnukset ja hallinta-sivun hallintasivuston käyttäjätunnukset keskenään, jolloin tunnuksia luotiin usein väärälle sivustolle. Tämä korjattiin lisäämällä tarkentavia ohjetekstejä sivuille, joissa sen katsottiin olevan aiheellista. Tekstit selventävät sivun toimintojen käyttötarkoitusta.

Testijakson aikana havaitut ongelmat korjattiin ja sivuston todettiin vastaavan vaatimusmäärittelyä. Sovellus luovutettiin ja otettiin käyttöön syksyllä 2010.

6 YHTEENVETO

Insinööriyön tavoitteena oli suunnitella ja toteuttaa hallinnointisivusto Hitmedian HitCaster-nettimusiikkikaupalle. Samanaikaisesti nettikauppaa kehitettäessä oli luotu alkeellinen hallintasivu, jolla onnistui kuitenkin vain kriittisimmät toiminnot, kuten rekisteröitymisten ja levyjen hyväksyntä. Käytön hankaluus ja rajoittuneisuus poiki idean kattavan hallintasivuston kehittämistä insinööriyöksi.

Suunnitteluvaiheessa listattiin tulevan sivuston vaatimukset, minkä jälkeen käynnistettiin kehitys. Kehityksen alussa oli hyödyllistä suunnitella sivustolle rakenne, jonka avulla uusien sivujen lisääminen olisi mahdollisimman helppoa. Tämä toteutettiin siirtämällä kaikille sivuille yhteinen erillisiin tiedostoihin, jotka PHP:n avulla liitettiin jokaiseen sivuun. Tämän avulla sivuja kehitettäessä kyettiin keskittymään vain alisivulle ominaisen toiminnallisuuden tekemiseen. Myös sivuston laajentaminen tulevaisuudessa helpottuu.

Kehitystyökaluna käytettiin PHP Development Tools -lisäosaa avoimen lähdekoodin Eclipse-ohjelmointiympäristön kanssa. Kun sivusto täytti suunnittelun aikana luodun vaatimusmäärittelyn, se otettiin testikäyttöön, jonka aikana havaitut pienet ongelmat korjattiin. Lopulta sovellus luovutettiin ja otettiin asiakasyrityksessä käyttöön syksyllä 2010.

Vaikka PHP ja WWW-sivujen kehitys olikin jo ennestään tuttua, tarjosi insinööriyö mahdollisuuden itsenäisesti suunnitella ja toteuttaa alusta loppuun kokonainen sivustokokonaisuus. Prosessin suunnittelemisesta ja toteuttamisesta saatu kokemus tulee epäilemättä olemaan hyödyksi tulevassa työelämässä.

Valittu kehitysprosessi osoittautui hyvin toimivaksi kyseisenlaisen sivuston toteuttamisessa. Kattavan suunnittelutyön ansiosta kehitysvaiheessa ei jouduttu tekemään suunnitelmiin muutoksia. Tässä varmasti auttoi myös

aiempi kokemus PHP:n kanssa, sillä jossain määrin kyettiin ennakoimaan millaisia ongelmia kehitysvaiheessa voisi ilmetä. Verrattain lyhyehkön testausjakson aikana käytiin usean koehenkilön kanssa läpi kaikki sivuston ominaisuudet, ja voitiin kohtuullisella varmuudella todeta niiden toimivan kuten pitääkin.

Sivuston jatkokehitys luotiin mahdollisimman helpoksi luomalla sivupohja, jota kaikki sivut käyttävät, ja luomalla usein tarvituista toiminnoista funktioita turhan koodin toistamisen välttämiseksi. Työstä saatujen positiivisten kokemusten perusteella olisi hienoa, mikäli asiakasyrityskin pitäisi kehitettyä tuotetta hyödyllisenä.

LÄHTEET

1. WWW-palvelin. 2010. Saatavissa <http://fi.wikipedia.org/wiki/WWW-palvelin>. Hakupäivä 28.9.2010.
2. March 2010 Web Server Survey. 2010. Saatavissa http://news.netcraft.com/archives/2010/03/17/march_2010_web_server_survey.html. Hakupäivä 13.10.2010.
3. FAQ - Httpd Wiki. 2011. Saatavissa <http://wiki.apache.org/httpd/FAQ>. Hakupäivä 31.1.2011.
4. Microsoft Web Platform - Server, IIS, Internet Information Services. 2011. Saatavissa <http://www.microsoft.com/web/platform/server.aspx>. Hakupäivä 31.1.2011.
5. Framework Usage Statistics. 2010. Saatavissa <http://trends.builtwith.com/framework>. Hakupäivä 14.10.2010.
6. PHP and ASP.NET - a feature list. 2010. Saatavissa http://www.codeproject.com/KB/applications/php_asp_net.aspx. Hakupäivä 14.10.2010.
7. PHP Usage Statistics. 2011. Saatavissa <http://trends.builtwith.com/framework/PHP>. Hakupäivä 31.1.2011.
8. Relational model. 2010. Saatavissa http://en.wikipedia.org/wiki/Relational_model. Hakupäivä 26.10.2010.
9. Relational database. 2010. Saatavissa http://en.wikipedia.org/wiki/Relational_database. Hakupäivä 26.10.2010.

10. SQL Introduction. 2011. Saatavissa

http://www.w3schools.com/sql/sql_intro.asp. Hakupäivä 31.1.2011.

11. HitCaster - Musiikkikokoelma mukana ja kotona. Saatavissa

<http://www.hitcaster.net/guide-index.php>. Hakupäivä 24.1.2011.