

TIETOKANTAPOHJAISEN PALVELUN TOTEUTTAMINEN INTERNETIIN

LAHDEN AMMATTIKORKEAKOULU
Mediatekniikan koulutusohjelma
Teknisen visualisoinnin suuntautumisvaihtoehto
Opinnäytetyö
8.5.2006
Jenny Koivula

**Lahden ammattikorkeakoulu
Mediatekniikan koulutusohjelma**

**KOIVULA, JENNY: Tietokantapohjaisen palvelun toteuttaminen
Internetiin**

Teknisen visualisoinnin opinnäytetyö
37 sivua, 10 liitesivua

Kevät 2006

TIIVISTELMÄ

Tämä opinnäytetyö käsittelee tietokantapohjaisten palveluiden toteuttamista Internetiin. Työssä on mietitty tietokantojen käyttötapoja WWW-sivustolla sekä ohjelmoinnin ja tietokannan yhdistämistä toimivan, helposti ylläpidettävän WWW-palvelun toteuttamiseksi.

Työssä käydään läpi yleisimmät tekniikat dynaamisen sivuston toteuttamiseen ilmaisilla menetelmillä. Tietokantaohjelmistona on käytetty MySQL-tietokannan hallintajärjestelmää ja ohjelmointikielenä PHP:tä. Työssä esitellään MySQL:n ja PHP:n tarpeellisimmat ominaisuudet WWW-palvelun toteuttamiseen ja ylläpitoon. Työssä käsitellään myös tietoturva mm. lomakkeiden lähetyksessä ja tiedostojen salaamisessa.

Casena tehtiin WWW-sivusto, joka hyödyntää MySQL-tietokantaa ja PHP-ohjelmointia. Osuudessa kerrotaan työn etenemisestä sekä ongelmista ja ylimääräisestä työstä, jotka dynaamisuudella olisi voitu välttää. Kehitysympäristönä toimi Apache-palvelinohjelmistolla varustettu kotitietokone. Osuudessa käsitellään myös sivuston julkaisuun ja ylläpitoon liittyviä asioita.

Avainsanat: WWW, WWW-ohjelmointi, tietokannat, relaatiotietokanta, MySQL, PHP

**Lahti University of Applied Sciences
Faculty of Technology**

**KOIVULA, JENNY: Implementing a service based on a database for
the Internet**

Bachelor's thesis in visualization engineering
37 pages, 10 appendices

Spring 2006

ABSTRACT

This thesis deals with implementing services for the Internet which are based on a database. The work presents different ways of using of a database in WWW sites and how to combine programming and a database to implement a functional Internet service which is easy to maintain.

The work provides an overview of the most usual techniques of implementing a dynamic WWW site free of charge. The database software that was used was the MySQL database management system and the programming language was PHP. The work introduces the most necessary attributes of MySQL and PHP to implement and maintain a WWW service. It also deals with data security, for example in sending forms and hiding files.

The empirical part of the work is a WWW site which utilizes MySQL database and PHP programming. This part describes the work process, and problems and extra work that could have been avoided with more dynamic solutions. Development environment that was used was a home computer with an Apache server. This part also deals with the issues of publishing and maintaining the site.

Keywords: WWW, WWW programming, databases, relational database, MySQL, PHP

SISÄLLYS

1	JOHDANTO	1
2	TIETOKANTOJEN KÄYTTÖ INTERNETISSÄ	2
2.1	Tietokantojen tarkoitus ja käyttö	2
2.2	Relaatiotietokanta	2
2.3	MySQL-tietokantaohjelmisto	3
2.3.1	Lyhyesti MySQL:stä	3
2.3.2	MySQL:n ominaisuudet	4
2.3.3	MySQL-tietokannan peruselementit	5
3	YLEISESTI PHP-KIELEN KÄYTÖSTÄ INTERNETISSÄ	6
3.1	Lyhyesti PHP-kielestä	6
3.2	PHP-kielen ominaisuudet	7
3.3	PHP:n ja MySQL:n käyttö	9
3.3.1	Tietokantapohjaisen dynaamisen sivuston suunnittelu	9
3.3.2	PHP-koodi WWW-dokumentissa	10
3.3.3	Yhteyden ottaminen tietokantaan	11
3.3.4	MySQL-kyselyt	12
4	PALVELIMET JA OHJELMISTOT	13
4.1	WWW-palvelimet yleisesti	13
4.2	Palvelimen, tietokannan ja WWW-dokumentin suhteet	14
4.3	Tietokannan ja ohjelmoinnin asettamat vaatimukset Internet-palveluntarjoajalle	15
5	TIETOKANTOJEN SUUNNITTELU JA TOTEUTUS	16
5.1	Suunnittelun lähtökohdat	16
5.2	Taulujen väliset suhteet	16
5.3	Tietokannan luominen ja taulujen teko	18
5.4	Tietokannan ylläpidon vaihtoehdot	19
6	CASE: VAPAA LAHTI	22
6.1	Esittely	22
6.2	Palvelun suunnittelu ja toteutus	23
6.2.1	Suunnittelu ja kehitys	23
6.2.2	Sivuston rakenne	25
6.2.3	Rekisteröityminen, kirjautuminen ja istunnot	26
6.2.4	Hakutoiminto	28
6.3	Julkaisu	29

6.4	Ylläpito.....	30
6.5	Palvelun käyttö	32
6.6	Laajennusmahdollisuudet ja kehitysvaihtoehdot	33
7	YHTEENVETO	34
	LÄHTEET	35
	LIITTEET	37

TERMIT JA LYHENTEET

Apache:

Internetin yleisimmin käytetty avoimeen lähdekoodiin perustuva WWW-palvelinohjelmisto.

Avoin lähdekoodi, engl. Open Source:

Open Source -tuotteen lähdekoodia voi kuka tahansa käyttää, levittää, tutkia, muuttaa ja levittää muokattuja versioita eteenpäin.

C, Java, Perl, C++:

Ohjelmointikieliä.

CSS, Cascading Style Sheets:

Tyylikieli, joka kuvaa HTML-, XHTML- tai XML-dokumenttien ulkoasua.

Htaccess:

Apache-palvelimen asetustiedosto, jolla voidaan muokata palvelimen asetuksia ja käyttöoikeuksia. Käytetään usein WWW-hakemistojen suojaukseen.

HTML, Hypertext Markup Language:

WWW-dokumenttien laadintaan käytettävä merkintäkieli.

HTTP, Hypertext Transfer Protocol:

Selaimen ja palvelimen väliseen tiedonsiirtoon käytettävä yhteyskäytäntö eli protokolla.

MD5:

Merkkijonosta muodostettava 128-bittinen numerosarja, joka voidaan laskea vain yhteen suuntaan. Esitetään usein 32-merkkisenä heksakoodatussa muodossa.

MySQL:

Avoimeen lähdekoodiin perustuva tietokannan hallintajärjestelmä.

ODBC, Open Database Connectivity:

Microsoftin määrittelemä rajapinta, jonka avulla sovellukset voivat kommunikoida tietokantapalvelimen kanssa.

PHP, PHP: Hypertext Preprocessor:

Eryityisesti WWW-sovelluksiin suunniteltu WWW-palvelimessa tulkittava ohjelmointikieli. Perustuu avoimeen lähdekoodiin.

PHP-tulkki:

WWW-palvelimeen asennettu tulkki, joka käsittelee WWW-dokumentissa olevan PHP-koodin.

RDBMS, Relational Database Management System:

Relaatiotietokannan hallintajärjestelmä.

SQL, Structured Query Language:

Tietokantojen käsittelyyn käytettävä standardoitu kieli.

SSH, Secure Shell:

Turvalliseen tiedonsiirtoon tarkoitettu järjestelmä.

WWW, World Wide Web:

Internetissä toimiva maailmanlaajuinen hypermediadokumenttien verkosto.

WWW-dokumentti:

HTML-merkkäuskielellä kirjoitettu tekstidokumentti, jonka selain tulkkaa esitettävään muotoon.

WWW-hotelli:

Palvelintietokone, jolta voidaan vuokrata levytilaa ja palveluita omille WWW-sivuille.

WWW-palvelin:

Tietokone tai ohjelmisto, joka jakaa dokumentteja Internetissä HTTP-yhteyskäytäntöä käyttäen.

XML, Extensible Markup Language:

Tiedon rakennetta kuvaava kieli eli metakieli.

1 JOHDANTO

Yrityksen WWW-sivusto on usein vain pieni osa koko toiminnasta. Sen ei haluta aiheuttavan huolta eikä lisätyötä, sen ylläpito ei saisi maksaa liikaa, sivuston pitäisi olla ajantasainen sekä toimivuudeltaan varma ja luotettava. Siitä täytyisi olla hyötyä henkilöstölle ja asiakkaille.

Helposti ylläpidettävän WWW-sivuston toteuttaminen on yksinkertaistunut ja tarve sellaisille yleistynyt. Yritykset haluavat sivustostaan sellaisen, että sisällön päivittäminen on mahdollista jokaiselle tietokoneen käytön perustaidot hallitsevalle henkilölle. Tällainen sivusto vaatii rakenteen, jossa WWW-dokumentteja ei tarvitse luomisen jälkeen muokata. Silloin dokumentit eivät välttämättä sisällä mitään muuta kuin yksittäisen sivun rakenteen. Sisältö tulee jostain muualta, kuten erillisistä teksti-, tai kuvatiedostoista tai tietokannasta. Tietokannan etuna on suurten tietomäärien tallentaminen ja säilyttäminen sekä tiedon hakemisen ja ylläpidon helppous.

Sivuston sisällön muodostumisesta vastaa dokumentin teossa käytetty ohjelmointikieli. Ohjelmoinnin avulla sivu ottaa itse yhteyden valittuun tietolähteeseen ja sisällyttää halutut elementit sivulle. Tietolähteen muokkaaminen tai uusien tietolähteiden lisääminen voi myös tapahtua ohjelmoinnillisesti. Käytettäessä tietokantaa tietolähteenä, ohjelmointikoodiin lisätään tietokantakyselykieltä. Tietokantakyselykielellä annetaan ehdot, joiden perusteella tietokannasta selaimen tulostettava tai tietokantaan tallennettava tieto määritellään. Kun sivuston ja dokumenttien rakenne suunnitellaan hyvin, ei ylläpitäjän tarvitse huolehtia ohjelmoinnista tai tietokantakyselykielestä. Hyvin suunnitellulla ja yksinkertaisella WWW-pohjaisella ylläpitokäyttöliittymällä onnistuvat helposti sekä sivuston sisällön että tietokannan ylläpito.

Tietokantaohjelmistojen hinnoissa on suuria eroja. Eroja löytyy tietysti myös ohjelmistojen ominaisuuksista. Kuitenkin ilmaisiakin vaihtoehtoja on, eikä toiminnallisuudesta tai luotettavuudesta ole pakko tinkiä. Yksi esimerkki ilmaisesta ja toimivasta tietokantaohjelmistosta on MySQL, jonka ominaisuuksista ja käyttötavoista tässä työssä kerrotaan. MySQL:n kyselykieli on SQL, jota myös käsitellään työssä. Ohjelmointikielistä työ perehtyy PHP-kieleen, joka on suunniteltu erityisesti WWW-sovelluksiin, ja jonka monipuoliset ominaisuudet soveltuvat hyvin juuri MySQL:n kanssa käytävään yhteistyöhön.

2 TIETOKANTOJEN KÄYTTÖ INTERNETISSÄ

2.1 Tietokantojen tarkoitus ja käyttö

Tietokannat ovat nykyään hyvin yleisiä Internetissä ja yleistyvät koko ajan Internetin laajentuessa. WWW-sivuilla olevaa tietoa on helppo järjestää ja muokata tietokannan avulla. Tietoa voidaan jakaa maailmanlaajuisesti riippumatta käyttäjän tietokoneen ominaisuuksista. Tietokannan käyttäjät ja käyttöoikeudet voidaan määritellä. Tietokantoihin voidaan tallentaa myös esimerkiksi kuvatiedostoja.

Tietokantoja käyttävät yksityishenkilöt esimerkiksi ylläpitäessään rekisteriä henkilökohtaisesta omaisuudestaan, jolloin WWW-sivun kautta tiedon muokkaus, haku ja ylläpito ovat yksinkertaisia toimintoja. Tällöin WWW-sivut voivat olla yksityiset eivätkä muut käyttäjät pääse tarkastelemaan niitä, mutta tietokannan haltija pääsee tietoihin käsiksi miltä Internet-yhteydellä varustetulla koneelta tahansa. Hyvä puoli tietokannasta WWW-sivulle tulevassa tiedossa on myös se, ettei päivitettyä WWW-dokumenttia tarvitse siirtää uudelleen palvelimelle.

Tärkeämpää tietokantojen yhdistäminen dynaamisiin WWW-sivuihin on kuitenkin yrityksille, jotka voivat käyttää sitä sisällön hallintaan tai esimerkiksi sähköiseen kauppaan. Tekniikalla voidaan helposti ylläpitää esimerkiksi tuotetietokantoja ja asiakasrekistereitä. Tuoteluettelo voi näkyä sivustolla, jossa asiakkaat näkevät sen, ja henkilökunta voi päivittää tietoja esimerkiksi salasanasuojatun WWW-sivun kautta. Erilaisia oikeuksia eri käyttäjille voidaan määritellä tarpeen mukaan. Koska myös tiedostojen tallentaminen tietokantaan on helppoa, saadaan tuotteiden kuvatkin helposti sivustolle ilman WWW-dokumentin muuttamista. Asiakasrekisteriin asiakas itse voi lisätä tietonsa sivuston kautta saadakseen esimerkiksi tiedotteita ja tarjouksia, ja henkilökunnalla on mahdollisuus nähdä tiedot asiakkaistaan. Yrityksillä on paljon erilaisia käyttötapoja tietokannan ja WWW-sivuston yhdistämiselle liiketoiminnan palvelemiseksi, ja koko ajan tapoja keksitään lisää.

2.2 Relaatiotietokanta

Varhaisia tietokantamalleja ovat hierarkkinen tietokantamalli ja verkkotietokantamalli. Nämä mallit eivät toimineet hyvin suurien tietomäärien käsittelyssä ja säilytyksessä. Nykyään tietokantamalleista käytetyin on relaatiotie-

tokantamalli, jonka kehitti IBM:n tutkija E. F. Codd. Coddin ideana oli käyttää matematiikan oppeja ja rakenteita tietojen käsittelyssä. Vuonna 1970 hän esitteli relaatiotietokantamallin teoksessaan "A Relational Model of Data for Large Shared Databanks". Malli perustui matematiikan joukkoteoriaan ja ensimmäisen kertaluvun predikaattilogiikkaan. (Hernandez 2000, 4-12.)

Relaatiotietokannassa tiedot tallennetaan relaatioina, jotka käyttäjä näkee tauluina. Relaatio muodostuu tietueista ja kentistä, joiden fyysisellä järjestyksellä ei ole merkitystä, vaan jokainen tietue tunnustetaan muista poikkeavan arvon sisältävän kentän avulla. Näin ollen käyttäjän ei tarvitse tietää tietueen fyysistä sijaintia hakeakseen sen sisältämät tiedot. (Hernandez 2000, 12.)

Relaatiotietokantamallin etuja ovat esimerkiksi sisäänrakennettu monitasoinen eheys, datan looginen ja fyysinen riippumattomuus tietokantasovelluksista, taattu tietojen yhtäpitävyys ja oikeellisuus sekä tietojen haun helppous. Relaatiotietokantamallin edut ovat osoittautuneet hyödyllisiksi kaikille tietojen kerääjille ja käsittelijöille. (Hernandez 2000, 16.)

Relaatiotietokantajärjestelmien kyselykielenä käytetään SQL-kieltä (Structured Query Language). SQL on standardoitu kieli, joka on luotu IBM:n tutkimuskeskuksessa 1970-luvun alussa. Standardoidun kielen käyttäminen edistää sovellusten siirrettävyyttä. Perusasiat ovat samat kaikissa suosituissa SQL-kieltä käyttävissä järjestelmissä, vaikka jokaisella tietokantajärjestelmällä on oma joukkonsa poikkeuksia standardista. (Meloni 2003, 10.)

SQL-kiielellä voidaan muokata tietokannan rakennetta, muuttaa järjestelmän turva-asetuksia sekä lisätä käyttäjän oikeuksia tietokantaan, suorittaa kyselyjä tietokannasta ja päivittää tietokantaa (2kmediat/SQL, 2006). Tietokantaa hallitaan tietokannan hallintajärjestelmällä. Yleisin ilmainen relaatiotietokantaohjelmisto on MySQL. Muita ilmaisia relaatiotietokantaohjelmistoja ovat esimerkiksi PostgreSQL ja Firebird.

2.3 MySQL-tietokantaohjelmisto

2.3.1 Lyhyesti MySQL:stä

MySQL on relaatiotietokantojen hallintajärjestelmä, RDBMS (Relational Database Management System). Se varastoi kaikki tietokannat ja niiden sisäl-

tämän tiedon ja käsittelee niitä yhtenä kokonaisuutena. MySQL:lle lähetettävät komennot kirjoitetaan SQL-muodossa. Kommunikointi hallintajärjestelmän kanssa toteutetaan asiakaskäyttöliittymän tai ohjelmointirajapinnan kautta. (Meloni 2003, 10-15.)

MySQL:n kehittivät suomalainen Michael Widenius ja ruotsalaisen David Axmark, alun perin oman ohjelmointiyrityksensä käyttöön. MySQL:n vuonna 1995 julkaistu ensimmäinen versio sai niin laajan suosion, että sen luojat perustivat sen ympärille MySQL AB -nimisen yrityksen, joka keskittyi MySQL-palveluihin ja -tuotteisiin. MySQL AB on kasvanut ja laajentanut toimintaansa siitä asti. Vaikka MySQL:stä puuttui ominaisuuksia, jotka kuuluvat yritystason tietokantatuotteisiin, saavutti se suosiota nopeutensa ja skaalattavuutensa ansiosta. Ominaisuuksia on tullut lisää uusien versioiden myötä. (Gilmore 2005, 511.) Uusin ja suositeltava MySQL-versio on lokakuussa 2005 julkaistu 5.0. Uusimman version voi ladata MySQL AB:n WWW-sivustolta osoitteesta www.mysql.com.

2.3.2 MySQL:n ominaisuudet

MySQL on nykyään kaikkein suosituin ilmainen avoimeen lähdekoodiin perustuva tietokanta sen joustavuuden, monipuolisuuden ja suorituskyvyn ansiosta. Se sopii kaikenkokoisten WWW-palvelujen tietokannaksi ja selviytyy suurista määristä kyselyitä kerrallaan. MySQL:n asennus on helppoa, eikä käyttö vaadi erityistä tietokantaosaamista. MySQL AB sanoo tuotettaan erittäin nopeaksi, luotettavaksi ja helposti muunneltavaksi. MySQL on yhteensopiva useimpien käyttöjärjestelmien kanssa. (MySQL AB/ Top Ten Reasons to Use MySQL, 2006.)

MySQL:n käyttö on ilmaista, kunhan lisenssin ehtoja noudatetaan (GNU General Public License (GPL)). Maksullisen lisenssin tarvitsee esimerkiksi silloin, jos myy MySQL-pohjaisia ohjelmistoja tai ohjelmistoja, jotka toimiakseen tarvitsevat MySQL:n. Kaupallisenkin WWW-sivuston tietokantana se on ilmainen käyttää. (MySQL AB/Licensing Policies, 2006.)

MySQL noudattaa asiakas-palvelin-arkkitehtuuria, eli tietokannan käsittely tapahtuu palvelinohjelmiston kautta. MySQL-palvelimeen voidaan ottaa yhteys yleisimmillä WWW-ohjelmointikielillä. Näitä sanotaan ohjelmointirajapinnoiksi. (Heinisuo 2003, 36.)

MySQL:n uusin versio 5.0 on yhtiön mukaan merkittävin tuotepäivitys sen historiassa. Siinä on parannettu ominaisuuksia asiakkaiden toivomusten mukaisesti tinkimättä kuitenkaan suorituskyvystä, luotettavuudesta tai helpokäyttöisyydestä. Uusien SQL 2003 -standardin mukaisten ominaisuuksien kanssa MySQL on kustannustehokkain ratkaisu laajoihinkin sovelluksiin. 5.0-version uusia ominaisuuksia ovat esimerkiksi tallennetut proseduurit, herättimet, näkymät, cursorit ja hajautetut transaktiot. (MySQL AB/ MySQL 5.0 Now Available for Production Use, 2006.) Näillä ominaisuuksilla MySQL on noussut ammattilaistietokantojen tasolle.

2.3.3 MySQL-tietokannan peruselementit

MySQL-tietokannan viisi peruselementtiä muiden tietokantaohjelmistojen lailla ovat taulu, kenttä, tietue, avain ja indeksi. Tietokantaan luodaan tauluja, jotka sisältävät tietoa. Taulu on tietokannan suurin elementti, ja sen sisällä ovat kentät ja tietueet. Taulu voi sisältää enintään 3398 kenttää. Kentät määritellään taulun luontivaiheessa. Kentälle pitää määritellä tyyppi ja pituus sen perusteella, millaista tietoa siihen tallennetaan. Tietueet ovat kenttiin tallennettavia arvojoukkoja. (Meloni 2003, 20-23.)

Avaimia käytetään taulujen suhteiden määrittelyyn sekä estämään saman tiedon tallentamista. Avaimia on kahdenlaisia: ensisijainen avain eli perusavain ja ainutlaatuinen avain. Perusavain toimii linkkinä taulujen välillä. Kentän määrittely ainutlaatuiseksi estää saman arvojoukon tallentamisen useaan kertaan samaan kenttään. Taulussa voi olla vain yksi perusavain, mutta ainutlaatuisia avaimia voi olla enemmän. (Meloni 2003, 23-24.)

Perusavaimiksi määritellyt kentät lisätään MySQL:ssä automaattisesti indeksiin. Indeksi on tietokannan hakemisto, joka nopeuttaa hakua. Jos taulusta haetaan tietoa jonkun muun kuin indeksoidun kentän perusteella, täytyy koko taulun tiedot käydä läpi. Tämä hidastaa kyselyä ja kuormittaa palvelinta. Indeksit voi myös määrittää itse tarpeen mukaan. (Meloni 2003, 24-25.)

	Perus- avain		KENTTÄ	
Kenttien nimet	esim ID	esimerkki	kommentti	aika
	1	Ykkönen	Kommentointia...	2006-03-08 04:10:13
	2	Toinen	Tekstiä..	2006-03-08 04:10:29
TIETUE	3	Lisää	Lisää rivejä tässä	2006-03-08 04:11:05

Kuva 1: Taulun elementit

3 YLEISESTI PHP-KIELEN KÄYTÖSTÄ INTERNETISSÄ

3.1 Lyhyesti PHP-kielestä

PHP (PHP: Hypertext Preprocessor) on laajalle levinnyt ja yleiskäyttöinen ohjelmointikieli, joka sopii erityisesti WWW-sovelluksiin. Se voidaan upottaa suoraan HTML-dokumentin sisään. PHP on ilmainen ja avoimeen lähdekoodiin perustuva. Yksinkertaisena kielenä se sopii aloittelijalle, mutta tarjoaa monia kehittyneitä ominaisuuksia myös ammattilaisohjelmoijalle.

PHP on WWW-palvelimessa tulkittava ohjelmointikieli, mikä tarkoittaa sitä, että palvelin tulkaa WWW-dokumentissa olevan PHP-koodin ja lähettää vain tulokset selaimen. (PHP/Introduction, 2006.) Näin ollen PHP-koodi ei näy mitenkään käyttäjälle eikä WWW-dokumentin toiminta riipu käytettävästä selaimesta.

PHP-kielen historia

PHP:n kehitystyön aloitti Rasmus Lerdorf, alun perin seuratakseen omien kotisivujensa kävijämäärää. Ensimmäinen tuote oli kokoelma Perl-kielisiä komentosarjoja ja oli nimeltään Personal Home Page Tools. PHP/FI (Personal Home Page / Forms Interpreter) julkaistiin vuonna 1995 open source -tuotteena. Se sisälsi C-kielillä kirjoitetun laajennuksen, ja sillä pystyi kommunikoimaan tietokantojen kanssa. (PHP/ History of PHP and related projects, 2006.)

Ensimmäinen läheisesti nykyistä PHP:tä muistuttava versio, PHP 3, julkaistiin virallisesti vuonna 1998 yhdeksän kuukauden julkisen koekäytön jälkeen. Sen kehittivät Andi Gutmans and Zeev Suraski, ja se saavuttamaan valtavaan suosioon vaikutti monien suunnittelijoiden kiinnostus sen laajennettavuutta kohtaan. Koska kyseessä oli kokonaan uusi kieli, se nimettiin uudelleen PHP:ksi (PHP: Hypertext Preprocessor). (PHP/ History of PHP and related projects, 2006.)

Vuonna 2000 julkaistu PHP 4 sisälsi uuden ytimen, Zend Enginen, joka tuki kolmansien osapuolten ohjelmointirajapintoja ja tietokantoja. PHP 4:n sisältämiä muita ominaisuuksia olivat esimerkiksi tuki monille uusille WWW-palvelimille, HTTP-istunnot sekä turvallisemmat tavat käsitellä käyttäjiltä tulevia syötteitä. (PHP/ History of PHP and related projects, 2006.)

Uusin versio on heinäkuussa 2004 julkaistu PHP 5. Sen ytimenä on Zend Engine II, ja se sisältää monia uusia ominaisuuksia ja parannuksia, kuten täydellisen tuen olio-ohjelmoinnille sekä paljon uusia funktioita ja laajennuksia. (Zend - The php Company/What's New in PHP 5, 2006.)

3.2 PHP-kielen ominaisuudet

PHP-kielen syntaksi eli lauseoppi on lainattu suurimmaksi osaksi C-kielestä, ja siinä on samanlaisia piirteitä kuin Java-, Perl-, ja C++-kielissä. Koska PHP on suunniteltu erityisesti WWW-sovellusten rakentamiseen, sillä on monia etuja WWW-suunnittelijalle verrattuna muihin tekniikoihin, esimerkiksi sen rakenne on helposti opittavissa, ja selaimelta tulevat tiedot ovat suoraan käytettävissä. (Rantala 2005, 9-10.) PHP-kieli on käytännöllinen ja yksinkertainen ohjelmointikieli. Kommentosarja voi koostua yhdestä rivistä, eikä ohjelmakirjastoja välttämättä tarvita toisin kuin C-kielessä (Gilmore 2005, 5).

PHP:n perusrakenteeseen kuuluvat muuttujat ja vakiot, ohjausrakenteet, funktiot sekä oliot ja poikkeukset. Muuttujilla tarkoitetaan tietokoneen muistiin tallennettaviin tietoihin liitettyjä tunnuksia. Muuttujien arvo saattaa vaihdella ohjelman suorituksen aikana. PHP:ssä muuttujan tyyppiä ei tarvitse erikseen määrittää, vaan se määräytyy siihen tallennettavan tiedon perusteella. Muuttujiin viitataan tunnuksella *\$muuttuja*. (Rantala 2005, 25-26.)

Muuttujien yksinkertaisia tietotyyppejä ovat *boolean* (totuusarvo: TRUE tai FALSE), *integer* (kokonaisluku), *float* (liukuluku) ja *string* (merkkijono). Rakenteisia tietotyyppejä ovat *array* (taulukko) ja *object* (objekti, olio). Yksinkertaisiin tietotyyppeihin voi tallentaa yhden arvon, mutta rakenteisissa muuttujissa arvoja voi olla useita. Lisäksi on olemassa tietotyyppi *NULL*, joka tarkoittaa, ettei muuttujalla ole mitään arvoa. (Rantala 2005, 25-36.)

PHP ja WWW-palvelin luovat myös omia ennalta määriteltyjä muuttujiaan riippuen käytettävästä ympäristöstä. Apache-palvelinohjelmiston luoma muuttuja on esimerkiksi `$_SERVER['REMOTE_ADDR']`, joka sisältää pyynnön suorittaneen asiakkaan IP-osoitteen. PHP:n luomia muuttujia ovat esimerkiksi `$_GET`, joka sisältää assosiativisen taulukon muodossa kaikki *GET*-metodilla välitetyt muuttujat sekä `$_POST`, jossa välitetyt muuttujat on välitetty *POST*-metodilla. Käytettävän ympäristön muuttujista saa listan funktiolla `phpinfo()`. (Rantala 2005, 45-46.)

Vakio toimii samoin kuin muuttuja, mutta sen arvoa ei voi muuttaa kesken ohjelman suorituksen. Vakion tietotyyppi voi olla vain yksinkertainen eikä sen edessä tarvita dollarimerkkiä. Ennalta määrättyjä vakiota ovat esimerkiksi `_FILE_`, joka sisältää suoritettavan komentosarjan nimen absoluuttisena polkuna ja `_LINE_`, joka sisältää sen hetkisen tulkittavan rivin numeron. (Rantala 2005, 46-47.)

Ohjausrakenteilla ohjataan ohjelman kulkua. PHP-kielissä ne ovat lähes identtisiä C-kielen ohjausrakenteiden kanssa. Keskeisimmät ohjausrakenteet ovat *if* (valinta), *switch* (valinta), *while* (toisto), *do ... while* (toisto), *for* (toisto) ja *foreach* (toisto). Valintarakenteisissa tilanteissa ohjelma toteuttaa käskyjä, jos annettu ehto täyttyy. Toistorakenteessa käsky toteutetaan niin monta kertaa kuin asetettu toistoehto on voimassa tai ennalta asetetun lukumäärän verran. *Foreach*-toistorakenteella käydään läpi taulukon alkioita. (Rantala 2005, 48-55.)

Funktiot ovat pieniä ohjelmia, jotka suorittavat tietyn tehtävän. Funktiot voidaan jakaa valmiisiin kirjastofunktioihin ja käyttäjän määrittelemiin funktioihin. Funktio palauttaa aina jonkin arvon. PHP:ssä arvo palautetaan vaihtoehtoisella *return*-lauseella, ja jos *return*-lauseetta ei käytetä, palautetaan arvo *NULL*. (Rantala 2005, 56.)

Olio esittää jotain ohjelmaan liittyvää kokonaisuutta. PHP:n perusominaisuuksia olioissa ovat esimerkiksi luokan määrittely, luokan metodit ja omi-

naisuudet sekä luokan ilmentymät eli oliot. PHP:ssä olevat olio-ohjelmointiominaisuudet ovat parantuneet uusimman version myötä. (Rantala 2005, 64-65.)

Vuonna 2004 julkaistussa PHP 5 -versiossa on paranneltu toiminnallisuutta sekä lisätty yleensä kypsiin ohjelmointikieliarkkitehtuureihin yhdistettyjä piirteitä. Näkyvin parannus ovat valtavasti kehittyneet oliopohjaiset ominaisuudet. Tällaisia ovat esimerkiksi eksplisiittiset muodostin- ja tuhoajafunktiot, olion kloonaus, luokan abstraktio, muuttujat käyttöalueen määrittäminen, rajapinnat ja parannus olioiden hallintaan. Muita uusia ominaisuuksia ovat esimerkiksi poikkeusten hallinta, kehittynyt merkkijonojen hallinta sekä parantunut XML (*Extensible Markup Language*) -tuki. (Gilmore 2005, 4.)

3.3 PHP:n ja MySQL:n käyttö

3.3.1 Tietokantapohjaisen dynaamisen sivuston suunnittelu

Ennen kuin sivuston tekemisen aloittaa, kannattaa suunnitella tarkoin, miten palvelun haluaa toimivan. PHP:llä tehtyjen WWW-tiedostojen määrää voi huomattavasti rajoittaa, koska sen avulla WWW-sivu voidaan palvelimella luoda lopulliseen, käyttäjän haluamaan muotoon. Sivun eri elementit, kuten esimerkiksi otsikko, navigointipalkki, reunukset ja pääsisältö, voidaan tehdä omiksi PHP-tiedostoikseen ja koota niistä haluttu sivu. Näin muutoksia tehtäessä tarvitsee muuttaa vain yhtä tiedostoa, mutta vaikutus näkyy kaikissa sivuissa, joissa elementtiä on käytetty.

PHP:n ja tietokannan yhdistämisessä on merkittävänä hyvänä puolena juuri päivitettävyyden helppous. PHP-tiedostossa voidaan määritellä muuttujat, joilla haetaan tietokannasta käyttäjän haluama tieto PHP-pohjaiseen dokumenttiin. Näin voidaan samaa PHP-pohjaa käyttää useassa paikassa, koska sisältö muodostuu dynaamisesti käyttäjän haluamalla tavalla. Kun muutoksia halutaan esimerkiksi sivuston ulkonäköön, tarvitsee muuttaa vain yhtä tiedostoa. PHP:n ominaisuuksiin tutustuminen ennen sivuston tekemistä on hyvin tärkeää, jos haluaa sivustostaan mahdollisimman helposti päivitettävän. PHP:tä käytettäessä myös käyttäjälle selaimessa näkyvä lopputulos on helpompi varmistaa, koska PHP:n suoritus ei riipu selaimesta, kuten HTML:n kohdalla.

Sivustolla olevan linkin perään voidaan koodissa antaa muuttuja, jonka perusteella linkitetty sivu suorittaa halutut tapahtumat. Linkkejä voi olla suuri määrä, joista kaikista siirrytään samalle sivulle, mutta perässä oleva muuttuja määrittää sivuston muodostumisen selaimen. Muuttuja annetaan muodossa *?muuttuja*, ja muuttujan tarkistus suoritetaan *\$_REQUEST* -muuttujalla. Esimerkiksi uloskirjautumisessa voitaisiin siirtyä etusivulle, jolloin linkkinä olisi *index.php?uloskirjautuminen*, ja uloskirjautuminen tarkistettaisiin *index.php*-sivulla *if*-lauseella. Jos sivulle ollaan tultu uloskirjautumislinkin kautta, *if*-lause olisi voimassa, ja istunto suljettaisiin *session_destroy()* -funktiolla.

```
<?php
session_start();

if (isset ($_REQUEST ['logout'] ) )

{
session_destroy();
}
?>
```

Muuttujat voivat tulla myös HTML-lomakkeelta. Näin jokaiselle linkin ei tarvitse päätyä omalle sivulleen, ja dokumenttien määrää saadaan vähennettyä dynaamisuuden avulla.

3.3.2 PHP-koodi WWW-dokumentissa

PHP-koodi upotetaan WWW-dokumenttiin *<?php* ja *?>* -merkkien väliin. Tiedosto tallennetaan *.php-muodossa. Koodin sijainnilla tiedostossa ei ole väliä. PHP-lause päätetään puolipisteellä ja PHP-lausekokoelma merkitään aaltosulkeiden sisään. Kommentit, joiden ei haluta vaikuttavan koodiin, merkitään // ... (yksirivinen kommentti) tai /* ... */ (monirivinen kommentti). Koodia selventävien kommenttien merkitseminen on hyödyllistä koodin jatkokäyttöä ajatellen.

Toinen tapa liittää PHP-koodia WWW-dokumenttiin on ilmoittaa käytettävä kieli HTML:n muotoilukoodissa *script*-elementissä. Seuraavassa esimerkissä esitellään molemmat tavat.

```
<html>
<body>

<?php
// kommentti
print "PHP-koodi1";
?>

<script language="php">
print "PHP-koodi2";
</script>

</body>
</html>
```

HTML-koodia voi myös upottaa PHP-koodin sisälle. PHP:n *print*- tai *echo*-komentojen jälkeen lainausmerkkien sisään kirjoitetut HTML-muotoilukoodit toimivat selaimessa kuten normaalistikin. Silloin täytyy huomioida, että HTML-koodissa olevat lainausmerkit vaikuttavat PHP-koodiin. *Print*- tai *echo*-lauseen sisällä olevat lainausmerkit voi ohittaa laittamalla kenoviivan niiden eteen. Lainausmerkit voi myös muuttaa heittomerkeiksi.

3.3.3 Yhteyden ottaminen tietokantaan

WWW-dokumentti ja tietokanta kommunikoivat PHP-koodin avulla ohjelmointirajapinnan kautta. MySQL-kyselyt sijoitetaan WWW-dokumenttiin PHP-koodin sisälle. Ensimmäisenä täytyy muodostaa yhteys MySQL-tietokantaohjelmistoon. Tämä tapahtuu funktiolla *mysql_connect* (*\$dbhost*, *\$dbuser*, *\$dbpass*), jossa *dbhost* on palvelimen nimi, *dbuser* käyttäjä ja *dbpass* salasana. Funktion arvolle määritellään muuttuja, esimerkiksi *\$yhteys*, joka säilyttää sisäisen viittauksen tietokantayhteyden tietoihin. WWW-palvelimen ja MySQL:n sijaitessa samalla palvelimella voidaan palvelimen nimenä käyttää *localhost*-nimeä. Käyttäjätunnus ja salasana ovat tietokannan tunnukset. *\$yhteys*-muuttujaan lisätään mahdollisen virhetilanteen sattuessa ilmoitus yhdistämisen epäonnistumisesta. Seuraavaksi valitaan haluttu tietokanta funktiolla *mysql_select_db()*, jossa sulkujen sisään kirjoitetaan tietokannan nimi lainausmerkeissä. Lopuksi yhteys suljetaan funktiolla *mysql_close()*. Tietokantaan otetaan edellä mainitulla tavalla yhteys aina, kun tietokannan tietoja muokataan tai haetaan.

```

<?php

$yhteys = mysql_connect ("localhost", "tunnus", "salasana") or die ('I
cannot connect to the database because: ' . mysql_error());

mysql_select_db("tietokannan_nimi");

// tähän kohtaan lisätään varsinaiset tietokantakyselyt

mysql_close($yhteys);
?>

```

Tietokantaan yhdistämisen voi suorittaa myös tekemällä yhdistämisfunktioista PHP-tiedostot, jotka sisällytetään tarvittaessa WWW-dokumenttiin. Tämä on tietoturvaan ajatellen parempi tapa, koska silloin tietokannan salasanat eivät näy dokumentissa. Vaikka käyttäjälle ei PHP-koodi näykään, saattaa palvelimeen tai sen PHP-tulkkiin tulla vikoja tai toimintahäiriöitä, jotka saattavat vaikuttaa PHP:n käsittelyyn ja suoritukseen. Salasanat ja käyttäjätunnukset sisältävän tiedoston voi vielä laittaa muualle kuin WWW-hakemistoon tai suojata sen Htaccess-tiedostolla, jolloin siihen ei Internetin kautta pääse koskemaan. Lisäksi erillisissä PHP-tiedostoissa on hyvänä puolena muutosten tekemisen helppous, jos esimerkiksi palvelin vaihtuu.

3.3.4 MySQL-kyselyt

Kun valittuun tietokantaan ollaan yhteydessä, voidaan suorittaa kyselyitä. Kyselyillä tietokantaa pystyy muokkaamaan ja hakemaan siitä tietoja. PHP välittää SQL-komennon tietokannalle käytettäessä *mysql_query*-komentoa.

INSERT-komennolla tietokannan haluttuun tauluun lisätään tietoja. Lisättävät tiedot eli muuttujien arvot tulevat usein HTML-lomakkeelta käyttäjän antamina syötteinä, jolloin *INSERT*-lauseessa määritellään vain tiedoille tarkoitetut kentät sekä muuttujien nimet. *SELECT*-komennolla haetaan tietoja. Siihen voidaan lisätä paljon määreitä, joilla hauista saa monimutkaisiakin, ja joilla tulostettavien tietueiden määrään ja järjestykseen voi vaikuttaa. Tietokannan tietoja muokataan *UPDATE*- tai *REPLACE*-komennolla, joissa määritellään tietokantaan uudet arvot haluttuihin kohtiin. *DELETE*-komennolla voi tyhjentää kaikki taulun tiedot tai halutut tietueet.

PHP-tiedostoon kirjoitetut kyselyt voivat saada ehtonsa käyttäjän antamien tietojen perusteella. Esimerkiksi HTML-lomakkeelta tulevat tiedot voidaan PHP:llä asettaa muuttujiksi, joiden sisältämä arvo siirretään MySQL-kyse-

lyyn. Ongelmana voi olla erikoismerkkien käyttö käyttäjän antamissa tiedoissa. Esimerkiksi heittomerkin käyttö vaikuttaa SQL-lauseen suoritukseen. PHP-kielessä on funktio *magic_quotes_gpc*, joka lisää kenoviivan HTTP-pyyntöön mukana saapuneessa Get- ja Post-tiedoissa olevien heittomerkkien, lainausmerkkien, kenoviivan ja NULL-merkkien eteen, jotta niiden erityismerkitys poistuisi (Rantala 2005, 196). PHP:n konfiguraatioasetuksissa *Magic Quotes Gpc* -asetus on PHP:n vanhemmissa versioissa oletusarvoisesti päällä, jolloin funktioita ei tarvitse erikseen käyttää tietojen käsittelyssä. Siitä voi kuitenkin olla monissa sovelluksissa haittaa, ja koska sovelluksen tulisi toimia samalla tavoin eri tavoin konfiguroiduissa ympäristöissä, on uusimmissa PHP-ympäristöissä *Magic Quotes Gpc* -asetus oletusarvoisesti pois päältä (Rantala 2005, 196).

HTML-lomakkeelta SQL-kyselyiden sisällyttäminen annettuun tietoon voidaan estää myös käyttämällä *mysql_real_escape_string()* -funktioita, joka muokkaa tiettyjä SQL-kyselyissä erikoismerkityksellisiä merkkejä niiden erikoismerkityksen poistamiseksi. SQL-kielessä erikoismerkityksellisiä merkkejä ovat esimerkiksi heittomerkki, lainausmerkki ja kenoviiva. Erikoismerkitykset poistamalla tieto voidaan turvallisesti liittää MySQL-kyselyyn. (PHP/PHP-käsikirja 2006.)

4 PALVELIMET JA OHJELMISTOT

4.1 WWW-palvelimet yleisesti

WWW-sivusto sijaitsee WWW-palvelimella, joka tarkoittaa tietokonetta, jolle on asennettu palvelinohjelmisto. Palvelin lähettää haetun dokumentin selaimen. Palvelin keskustelee selaimen kanssa käyttäen HTTP-yhteyshäntäntä. Palvelin voi myös kerätä tietoa esimerkiksi dokumenttia hakeneista tietokoneista. Dynaamiset WWW-dokumentit, esimerkiksi PHP-kieltä sisältävät, luodaan valmiiksi vasta palvelimella ohjelmointikielen käskyjen mukaisesti ja lähetetään vasta sitten asiakaskoneelle.

WWW-palvelin voi sijaita myös kotikoneella. PHP-komentojen toimiminen selaimessa tarvitsee palvelimen, joka tulkkaa PHP-kieliset komennot ja lähettää tulokset sen jälkeen selaimen. Kun asennetaan esimerkiksi Apache-palvelinohjelmisto sekä PHP-tulkki omalle koneelle, voi PHP-koodien toimivuutta testata ilman WWW-dokumentin varsinaista julkaisua. Tämä helpottaa työskentelyä ja koodien muokkaamista huomattavasti.

Apache on monipuolinen WWW-palvelinohjelmisto. Se on Internetin yleisimmin käytetty palvelinohjelmisto nopeutensa, varmuutensa ja pitkän historiansa ansiosta. Se on ilmainen ja avoimeen lähdekoodiin perustuva. Apache on yhteensopiva useimpien käyttöjärjestelmien kanssa. (2kmediat/Apache, 2006) Apachen asentaminen kotikoneelle oman tietokantapohjaisen sivuston toimivuuden testaamista varten on hyvä vaihtoehto. Palvelimen voi asentaa siten, ettei sille ole pääsyä muilta koneilta. Apachen asentamista Windows-ympäristöön suositellaan yleisesti vain testikäyttöön, koska tässä yhdistelmässä tietoturvassa on puutteita. Linux on luotettavampi alusta julkisella palvelimella.

4.2 Palvelimen, tietokannan ja WWW-dokumentin suhteet

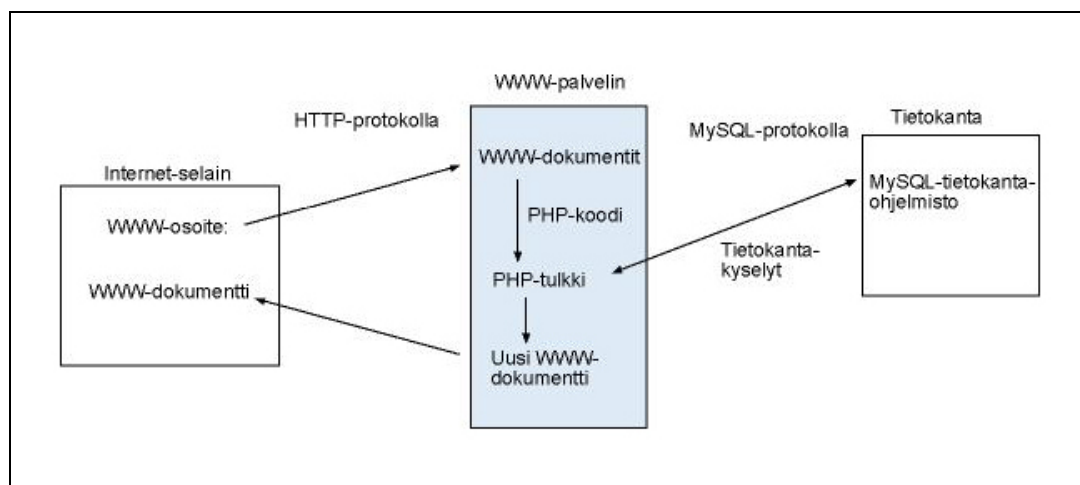
Tietokantapohjaisen, dynaamisen WWW-sivuston toimivuus vaatii tietokantaohjelmiston, palvelimen, joka tukee PHP:tä sekä selaimen, jonka kautta sivustoa katsotaan. Jos kotikoneelle on asennettu esimerkiksi Apache, PHP-tulkki sekä MySQL, voi tietokantapohjaisten PHP-dokumenttien toimivuutta testata ilman Internetiä.

Tietokantapohjaisen WWW-palvelun arkkitehtuuri noudattaa yleisesti niin sanottua kolmikerrosmallia, mikä tarkoittaa sitä, että selaimen ja tietokantapalvelimen väliin toteutetaan kerros, joka toimii asiakkaana tietokantapalvelimelle ja palvelimena selaimelle (Rantala 2005, 254). Yksi esimerkki kolmikerrosmallista on sovellus, jossa asiakaskerroksena toimii WWW-selain, välikerroksena on PHP-tuella laajennettu Apache-palvelin ja pohjakerroksena MySQL-tietokantapalvelin. Kolmikerrosmalliin perustuvan dynaamisen WWW-sivun muodostumisesta on kuvattu kuvassa 2.

Palvelimen ja ohjelmistojen suhteet ovat samanlaiset sivuston sijainnista riippumatta. Käyttäjä hakee selaimella tietystä osoitteesta WWW-dokumentin, joka sijaitsee palvelimella. Palvelimen ja selaimen vuorovaikutukseen käytetään HTTP-yhteyskäytäntöä. Palvelimella suoritetaan PHP-koodit ja PHP:n avulla otetaan yhteys tietokantaan sekä suoritetaan PHP-koodissa määritellyt SQL-kyselyt. PHP:n suoritettua tarvittavat toiminnot, muodostuu dynaaminen WWW-sivu, joka lähetetään selaimen käyttäjän nähtäväksi.

WWW-palvelin ottaa tietokantaan yhteyden käyttämällä joko tietokannan omaa protokollaa tai yleistä ODBC (Open Database Connectivity) -protokollaa. Esimerkiksi MySQL käyttää MySQL-protokollaa. ODBC:n etuna on pa-

rempi siirrettävyys sovellusten välillä. PHP tukee yleisimpien tietokantapalvelimien omia protokollia.



Kuva 2: MySQL/PHP-pohjaisen WWW-sivun muodostuminen

4.3 Tietokannan ja ohjelmoinnin asettamat vaatimukset Internet-palveluntarjoajalle

Julkaistaessa tietokantapohjainen WWW-sivusto on Internet-palveluntarjoajan ominaisuuksissa otettava huomioon tietokannan ja ohjelmoinnin asettamat vaatimukset. Ilmaisissa ja halvimmissa WWW-hotelleissa ei välttämättä ole ohjelmointiominaisuuksia eikä tietokantoja. Jos palveluntarjoajalla on PHP-tuki, voi olla mahdollista, että PHP:n vanhempi versio päivitetään uuteen asiakkaan pyynnöstä. Lisäksi täytyy ottaa huomioon mahdolliselle MySQL:lle annettu levytila ja tietokantojen määrä sekä se, onko tietokantoja ostettavissa lisäpalveluna. Useimmissa MySQL-tietokantoja sisältävissä palveluissa mukaan kuuluu myös tietokannan ylläpitoa ja hallintaa helpottava selainpohjainen hallintatyökalu.

Palveluntarjoajalta kannattaa selvittää, suoritetaanko tiedostojen ja tietokannan varmuuskopiointia ja jos suoritetaan, kuinka usein sekä annetaanko kopiointista minkäänlaista takuuta. PHP:n virheiden automaattinen tilastointi lokitiedostoon on tärkeä ominaisuus. Myös sivuston käytön tilastointi on hyödyllinen WWW-hotellipalvelu. Tilastoinnissa voi selvittää esimerkiksi vierailijoiden määrä, liikennöintimäärä, eniten ladatut sivut, vierailijoiden käyttämät käyttöjärjestelmät ja selaimet sekä niiden versiot.

5 TIETOKANTOJEN SUUNNITTELU JA TOTEUTUS

5.1 Suunnittelun lähtökohdat

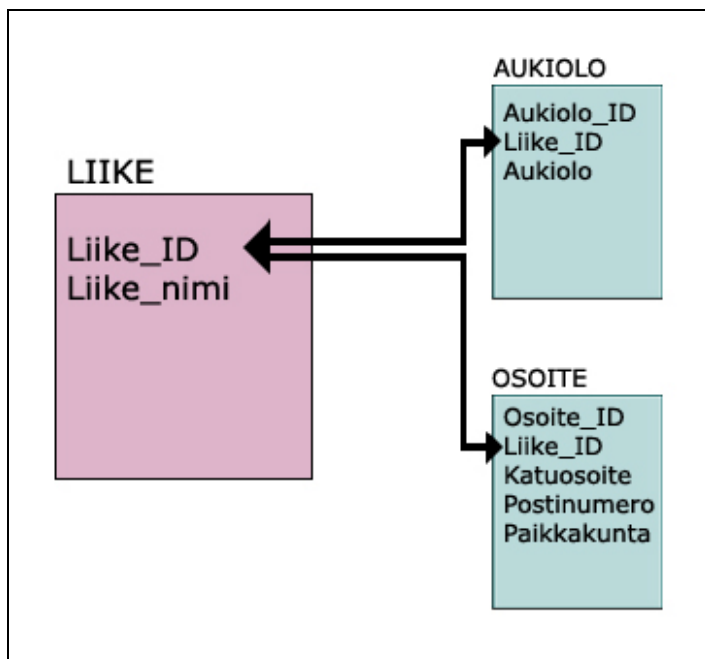
Tietokannan rakennetta suunniteltaessa on mietittävä, mitkä asiat halutaan tuoda sivustolle tietokannasta. Usein muuttuvien tietojen ylläpito WWW-dokumentissa on helppoa tietokannan avulla. Myös tieto, joka esiintyy usealla sivulla samanlaisena ja jota päivitetään useasti, on hyvä hakea sivustolle tietokannasta, jolloin muutoksia tehtäessä tarvitsee muokata vain tietolähdettä eli tietokantaa. Ylläpidettävyyttä voi helpottaa myös PHP-kielillä, mutta silloin tiedostot täytyy kuitenkin siirtää uudelleen palvelimelle, jota tietokannassa oleville tiedoille ei tarvitse tehdä. Lisäksi on yksinkertaisempaa, kun kaikki tieto on samassa paikassa. Interaktiivisessa palvelussa käyttäjän jättämät, sivustolle tarkoitetut tekstit voidaan tallentaa tietokantaan, jolloin esimerkiksi niiden esitysjärjestys sivustolla voidaan määrittellä monella tapaa. Jos tietokanta on käytössä, kannattaa sitä hyödyntää.

Jos sivustolla on mahdollisuus hakuun tietokannasta, täytyy suunnitteluvaiheessa ottaa huomioon miten tiedot kannattaa jäsentää, jotta haku onnistuu ja on nopeaa ja tehokasta. Relaatiotietokannasta tietoja voi hakea monista tauluista samanaikaisesti yhdessä haussa, jos taulujen suhteet ovat oikein määriteltä. Näin kaikkien tietojen ei tarvitse olla yhdessä suuressa, monimutkaisessa taulussa eikä tietoja tarvitse tallentaa moniin eri paikkoihin. Tietokannan tietojen ylläpito vaikeutuu, jos sama tieto esiintyy useassa paikassa. Tietokantaan tehdyt muutokset eivät saisi aiheuttaa ristiriitatilanteita eivätkä tuhota muita tietoja.

5.2 Taulujen väliset suhteet

Tieto kannattaa tallentaa järkevästi vähän tietoa sisältäviin tauluihin ja yhdistellä taulujen tiedot toisiinsa avaimien avulla. Taulut kannattaa suunnitella siten, ettei päällekkäisyyksiä esiinny. Se helpottaa tietokannan ylläpitoa. SQL-kyselyissä tietoja voidaan hakea monista tauluista kerrallaan. Tietokannan tietojen tulisi olla ristiriidattomia. Tietokannan pitäisi toimia niin, että tallennettava tieto on vain yhdessä paikassa. Tauluja ja sarakkeita pitäisi voida lisätä ilman, että ne vaikuttavat aikaisempaan rakenteeseen. Tietokannasta ei pitäisi voida poistaa tietoja, joista toiset tiedot ovat riippuvaisia.

Tauluilla voi olla yksi yhteen -suhteita, yksi moneen -suhteita tai moni moneen -suhteita. Yksi yhteen -suhteet voidaan yleensä poistaa sijoittamalla taulujen tiedot yhteen tauluun, ja tietojen jakaminen eri tauluihin vain vaikeuttaa ja hidastaa kyselyiden suorittamista. Moni moneen -suhteissa taas tiedot kannattaisi usein jakaa useampiin tauluihin, joissa olisi yksi moneen -suhteita. Yksi moneen -suhde on yleisin. Tätä suhdetta kutsutaan myös isä-lapsi-suhteeksi, eli isä-tilulla voi olla monta lapsi-tilua, mutta lapsi-tilulla vain yksi isä-tilu. (Hovi, Huotari, Lahdenmäki 2005, 37-39, 67-68.) Kuvassa 3 on esimerkki tällaisesta suhteesta, jossa päätaulussa määriteltyä kenttää käytetään viiteavaimena kahdessa muussa taulussa. Tässä esimerkissä olevissa lapsi-tiluissa ei tarvitse uudelleen määrittellä liikkeen nimeä, vaan se määrittyy Liike-tilun yksilöllisen Liike_ID-numeron perusteella. Jos päätaulussa olevan liikkeen nimi muuttuu, sitä ei tarvitse muuttaa jokaiseen tauluun, jossa tietoa tarvitaan. Näistä suhteesta olevista tauluista voidaan hakea tietoa samassa haussa, ja eri tauluille voidaan määrittellä erilaiset käyttöoikeudet.



Kuva 3: Esimerkki yksi moneen -suhteesta

Normalisointi on menetelmä, jolla tietokannassa olevan tiedon toistaminen minimoidaan. Normalisoinnilla pyritään pääsemään tietokannassa sellaiseen rakenteeseen, joka toistetun tiedon minimoinnin lisäksi on tehokas päivitysten kannalta sekä on yhdenmukainen ja muutosjoustava. Normalisointiteoriassa tauluille määritellään eri normaalimuotoja, joista yleisimmät ovat en-

simmäinen, toinen ja kolmas normaalimuoto. (Hovi ym. 2005, 85-86.) Normalisoidut tietokannat ovat levytilankäyttönsä puolesta tehokkaampia kuin normalisoimattomat, koska tilaa ei käytetä turhaan, eikä ylimääräistä tilaa jää tyhjäksi.

Ensimmäiseen normaalimuotoon päästään, kun poistetaan tietojen toisteisuus ja luodaan erilliset taulut muille kuin samaan asiaan välittömästi liittyville tiedoille. Toisen normaalimuodon sääntönä on, että mitkään avaimettomat kentät eivät saa olla riippuvaisia ensisijaisen avaimen osasta, vaan kenttien pitää kokonaisuudessaan olla suhteessa ensisijaiseen eli perusavaimeen. Lisäksi tietokannan pitää olla ensimmäisessä normaalimuodossa. Kolmas normaalimuoto edellyttää ensimmäisen ja toisen normaalimuodon toteutumista, ja lisäksi mitkään kentät eivät saa olla riippuvaisia toisista avaimettomista kentistä. (Meloni 2003, 34-35.)

5.3 Tietokannan luominen ja taulujen teko

Ensimmäisenä tietokannan käyttöönotossa täytyy luoda uusi tietokanta ja antaa sille nimi. Tämä tehdään *CREATE*-komennolla. Kun MySQL:ään ollaan yhteydessä, annetaan MySQL:lle komento *CREATE tietokannan_nimi*. Kun tietokantaa halutaan käyttää, kerrotaan se MySQL:lle komennolla *USE tietokannan_nimi*. Kokonainen tietokanta tai yksittäinen taulu poistetaan *DROP*-komennolla.

Ennen kuin tietokantaan voi tallentaa mitään, täytyy tehdä taulu. Uusi taulu tehdään komennolla *CREATE TABLE taulun_nimi*, jonka jälkeen määritellään taulussa olevat kentät sekä niiden tyypit ja pituus. Esimerkiksi *asiakkaat*-niminen taulu, jossa olisi kentät etu- ja sukunimelle, käyttäjätunnukselle sekä sanasanalle luotaisiin näin:

```
CREATE TABLE asiakkaat (etunimi VARCHAR (30), sukunimi  
VARCHAR (30), tunnus VARCHAR (10), salasana VARCHAR (10));
```

jossa kenttien nimet ovat "etunimi", "sukunimi", "tunnus" ja "salasana". Kaikkien kenttien tyyppi on annettu *varchar* eli muuttuvamittainen merkkijono, ja kenttien maksimipituudet on annettu suluissa tyyppimäärittelyn jälkeen.

Kenttätyytit

Taulun luontivaiheessa kentälle annetaan tyyppi, joka kertoo, millaista tietoa kenttään voidaan tallentaa. Kenttätyyppiä on kolmenlaisia: numeerisia, päiväys/kellonaika-tyyppisiä ja merkkijonokenttiä. Kenttien tyyppimäärittely on tietokannan tehokkuuden kannalta olennaista. Kentälle ei kannata turhaan varata tilaa enempää kuin se tarvitsee.

Numeerisia tyyppiä ovat esimerkiksi *INT*-tyypit, pienimmästä suurimpaan *TINYINT*, *SMALLINT*, *MEDIUMINT*, *INT* ja *BIGINT*. Näihin kenttiin voidaan tallentaa tietyinkokoisia kokonaislukuja. Luku voidaan määrittellä etumerkilliseksi *SIGNED*-määreellä tai etumerkittömäksi *UNSIGNED*-määreellä. Esimerkiksi kenttään, jonka tyyppi on *TINYINT UNSIGNED*, voidaan tallentaa lukuja nolasta 255:een, ja *BIGINT SIGNED* -kenttään voidaan tallentaa luvut -2147483648 – 2147483647. Muita numeerisia kenttätyyppiä ovat *FLOAT*, *DOUBLE* ja *DECIMAL*, jotka ovat liukulukuja, joissa täytyy olla etumerkki. Näihin kenttiin voi tallentaa desimaalilukuja.

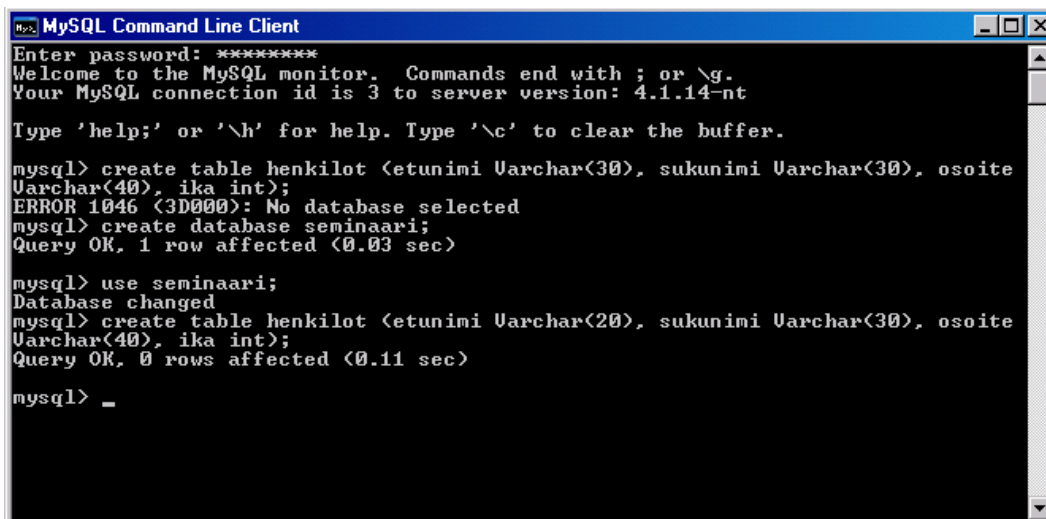
Päivämäärä- ja kellonaikatyyppiä on MySQL:ssä useita. *DATE* tallentaa päivämäärän muodossa VVVV-KK-PP. *DATETIME* tallentaa päivämäärän ja kellonajan muodossa VVVV-KK-PP TT:MM:SS. *TIMESTAMP* eli aikaleima tallentaa tietyn ajankohdan määrittelyssä muodossa, joko 14 merkillä kuten *DATETIME* tai vaihtoehtoisesti 12:lla, 8:lla tai 6 merkillä. *TIME* tallentaa kellonajan muodossa TT:MM:SS, ja *YEAR* tallentaa vuoden joko kahdella tai neljällä merkillä.

Merkkijonotyytit ovat yleisimpiä kenttätyyppiä. Niihin voidaan tallentaa tietynmittaisia tekstejä ja muita merkkijonoja. *CHAR*-kentän pituus on 255 merkkiä, ja siihen lisätään tarvittaessa välilyöntejä pituuden täyttämiseksi. *VARCHAR* on muuttuvamittainen merkkijono, jolle täytyy itse antaa pituus 1-255 merkin väliltä. *BLOB*- ja *TEXT*-kenttiin mahtuu 65535 merkkiä, ja näitä kenttiä on myös *TINY*- *MEDIUM*- ja *LARGE*-muodoissa. *BLOB*-kenttään voidaan tallentaa esimerkiksi kuvia binäärimuodossa. *ENUM*-tyyppiseen kenttään voidaan määrittellä tietyt valinnaiset arvot, jotka siihen voidaan tallentaa.

5.4 Tietokannan ylläpidon vaihtoehdot

MySQL-tietokantaa käytetään jonkin sovelluksen kautta. Ladattaessa MySQL tulee sen mukana komentorivipohjainen sovellus, jota sanotaan

MySQL-monitoriksi. Komentorivisovelluksissa MySQL-komennot kirjoitetaan *mysql/* -kehotteen jälkeen ja komennot loppuvat puolipisteeseen. Tällaisten ohjelmien käyttö vaatii tarkkaa tietämystä käytetystä SQL-kielestä, koska jokainen merkki ja merkin poisjääminen vaikuttaa.



```
MySQL Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3 to server version: 4.1.14-nt

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create table henkilot (etunimi Varchar(30), sukunimi Varchar(30), osoite
Varchar(40), ika int);
ERROR 1046 (3D0000): No database selected
mysql> create database seminaari;
Query OK, 1 row affected (0.03 sec)

mysql> use seminaari;
Database changed
mysql> create table henkilot (etunimi Varchar(20), sukunimi Varchar(30), osoite
Varchar(40), ika int);
Query OK, 0 rows affected (0.11 sec)

mysql> _
```

Kuva 4: MySQL:n tekstipohjainen asiakasohjelma

Tietokantojen varmuuskopiointi on tärkeä osa ylläpitoa. Koska kyseessä on elävä tietokantajärjestelmä, ei varmuuskopiointi välttämättä onnistu täydellisesti vain kopioimalla tiedot sisältävät hakemistot. Jos tietokantaan ollaan juuri tallentamassa tietoa, saattaa kopiosta tulla epätäydellinen. MySQL tarjoaa apuohjelmia varmuuskopioiden tekoon, joita kannattaa hyödyntää. (Meloni 2003, 256.)

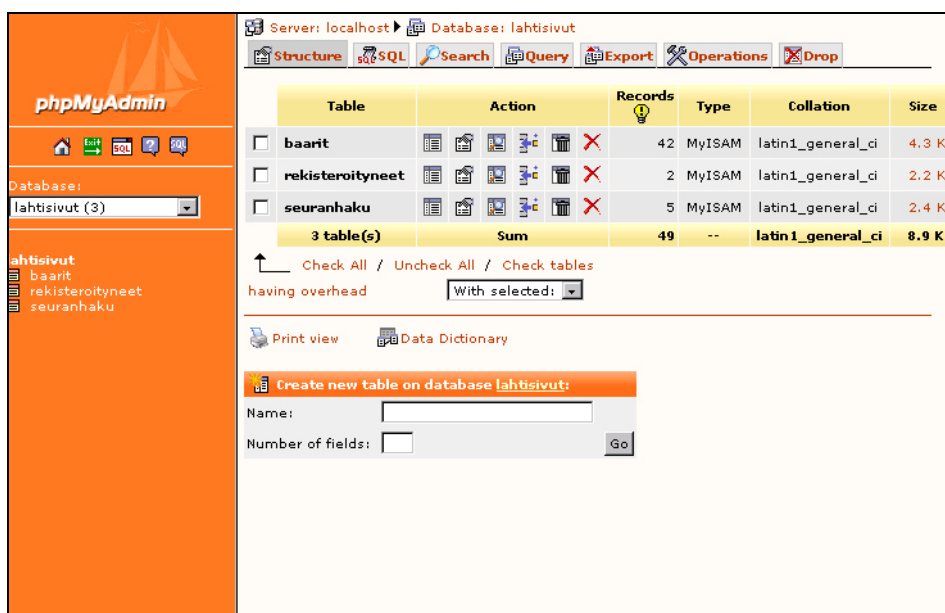
Taulurakenteista saa kopion *mysqldump*-apuohjelmalla, joka sisällyttää kopioon myös komennot, joilla myös taulun sisältö luodaan. Apuohjelmaa käytetään kirjoittamalla komentorivisovellukseen *mysqldump tietokannan_nimi > /polku_tiedostoon/kopiotiedoston_nimi.sql*. Tämä tulostaa tietokannan sisältämien taulujen rakenteet sekä niiden sisällöt SQL-lauseina. Tulostus tallentuu tiedostoon *kopiotiedoston_nimi.sql*. (Meloni 2003, 47-48.)

Toinen kopiointiin käytettävä väline on *mysqlhotcopy*-niminen apuohjelma. Se suorittaa joukon toimintoja, jotka tuottavat vakaan joukon varmuuskopiotiedostoja. *mysqlhotcopy*-apuohjelmaa käytettäessä täytyy olla sopivat oikeudet tietokantajärjestelmään sekä tietokantaan ja sen sisältämiin tauluihin. Apuohjelma ajetaan kirjoittamalla komentorivisovellukseen *mysqlhotcopy [vaihtoehdot] tietokannan_nimi /tallennuspaikka/*, ja tällä käynnistyneessä

prosessissa taulut lukitaan, välimuisti puhdistetaan, tiedostot kopioidaan ja lukot avataan. (Meloni 2003, 256-262.)

Jos sivusto sijaitsee julkisella WWW-palvelimella, on hyvä käyttää SSH (Secure Shell) -suojattua etäyhteyttä tietokannan ylläpitoon. Kun yhteys on muodostettu palvelimeen, valitaan tietokanta, jonka jälkeen päästään muokkaamaan tietoja kyselyjen avulla. Esimerkki ilmaisesta SSH-etäyhteysohjelmasta on PuTTY, joka on tekstipohjainen komentorivisovellus.

MySQL-tietokantaa voidaan ylläpitää helposti erilaisten graafisten, WWW-pohjaisten käyttöliittymien avulla. MySQL-tietokannan tietojen käsittelyyn ja muokkaukseen on monia eri ohjelmia, jolloin MySQL-komentojen osaaminen ei ole välttämätöntä. Esimerkki ilmaisesta ja toimivasta MySQL-tietokannan hallintaohjelmasta on phpMyAdmin. Sen avulla tietokantojen struktuurin, taulujen ja tietueiden hallinta on helppoa. Ohjelmalla voi luoda tietokantoja ja tauluja sekä lisätä, poistaa ja muuttaa niissä olevaa tietoa. Taulujen välisten suhteiden määrittely onnistuu ohjelmalla helposti. Taulujen ja kenttien rakennetta voi muuttaa tarvittaessa jälkikäteenkin. Graafisella sovelluksella tietokannan hahmottaminen on helpompaa, eikä virheitä, kuten väärin tietojen poistamista, tapahdu herkästi. Myös tietokantojen varmuuskopiointi ja siirtäminen ovat yksinkertaisia toimintoja. Muita graafisia MySQL-käyttöliittymiä ovat esimerkiksi MySQL-Front ja WinSQL. Tietokannan sisältävissä WWW-hotelleissa pakettiin kuuluu usein jokin graafinen tietokantasovellus.



Kuva 5: Esimerkki hallintaohjelmasta, phpMyAdmin-ohjelma

PHP-pohjaisen WWW-sivun kautta tietokantojen muokkaus on helppoa, kunhan sivun rakenne on ensin tehty järkeväksi ja toimivaksi. PHP-kielen avulla yhteys tietokantaan muodostuu automaattisesti ja kyselyt suoritetaan sivun PHP-koodissa. Eräs mahdollisuus on esimerkiksi tietokannassa olevan taulukon tulostaminen WWW-sivulla olevalle lomakkeelle, jossa voidaan suorittaa tarvittavat muutokset ja tallentaa ne yhtä nappia painamalla.

WWW-sivun kautta tapahtuva tietokannan päivitys on käyttökelpoinen ratkaisu henkilöille, joiden ei tarvitse luoda uusia tietokantoja tai tauluja tai poistaa niitä. Sivun tekijä voi tehdä ylläpitosivusta paljon yksinkertaisemman ja selkeämmän kuin tietokannan hallintaohjelmat ovat, jolloin kokematonkin käyttäjä voi hoitaa päivitykset helposti.

6 CASE: VAPAA LAHTI

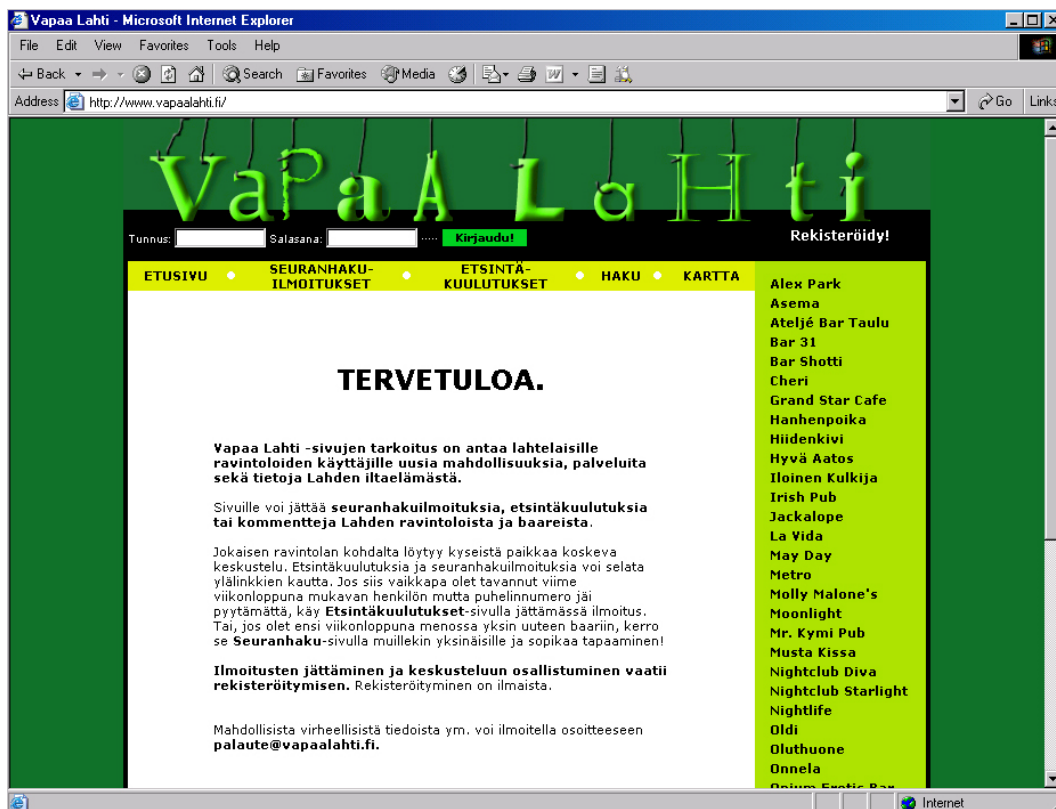
6.1 Esittely

Vapaa Lahti -nimisen WWW-palvelun tarkoitus on antaa lahtelaisille ravintoloita käyttäville seuraihmisille uusi kanava yhteydenpitoon ja yhteydenottoihin sekä keskustelulle. Sivusto on viihteellinen, mutta myös informatiivinen, sillä se sisältää ajankohtaista tietoa, kuten ravintoloiden hintatietoja ja aukioloaikoja.

Palvelussa on Lahden baarien ja ravintoloiden listaus sekä niiden kuvaukset ja sijainnit. Palvelun käyttäjä voi jättää ilmoituksia sekä osallistua keskusteluun. Keskustelut on järjestetty ravintolakohtaisesti, kuten myös seuranhakuilmoitukset ja etsintäkuulutusilmoitukset, mutta myös yleiselle keskustelulle ja tarkentamattomille ilmoituksille on paikkansa. Palvelussa on haku, jonka avulla ilmoituksia voi hakea tietokannasta halutuilla ehdoilla. Palvelun käyttäjäksi voi rekisteröityä, jos haluaa jättää ilmoituksia tai osallistua keskusteluun. Sivusto sijaitsee julkisella WWW-palvelimella, ja sen osoite on www.vapaalahti.fi.

WWW-sivuston rakenne on toteutettu HTML-, PHP- ja CSS-tekniikoilla. Palvelun toteuttamisessa on käytetty MySQL-tietokantaa, johon keskustelut ja ilmoitukset tallennetaan. Palveluun kuuluu myös Flash-ohjelmalla toteutettu kartta, josta selviävät ravintoloiden sijainnit.

Palvelun yhtenä tarkoituksena on kiristää hintakilpailua Lahden olutravintoloissa, joissa vaatimattomimmissakin paikoissa on korkeat hinnat. Uuden tupakkalain astuessa voimaan saattavat ihmiset jäädä viikonloppuisin kotiin nauttimaan halpoja kauppojen tuotteita, jos hinnat eivät laske. Lisäksi tarkoituksena on jatkaa keskustelua Lahteen keskittyneen järjestäytyneen rikollisuuden aiheuttamista ongelmista tavallisille ravintoloiden käyttäjille, ja opastaa ihmisiä niin halutessaan käymään turvallisissa paikoissa.



Kuva 6: Vapaa Lahti -sivusto

6.2 Palvelun suunnittelu ja toteutus

6.2.1 Suunnittelu ja kehitys

Palvelun suunnittelu aloitettiin miettimällä sivuston ominaisuuksia. Seuranhakumahdollisuus sekä etsintäkuulutusermoitukset olivat oleellisimpia, joille on tietenkin ollut kanavia pitkään monissa paikoissa. Palvelu kuitenkin rajoittuu vahvasti Lahden ja vielä etenkin sen keskustan ympäristöön, joten ihmisten etsiminen ja tapaaminen on helpompaa. Toiseksi pääasiaksi pal-

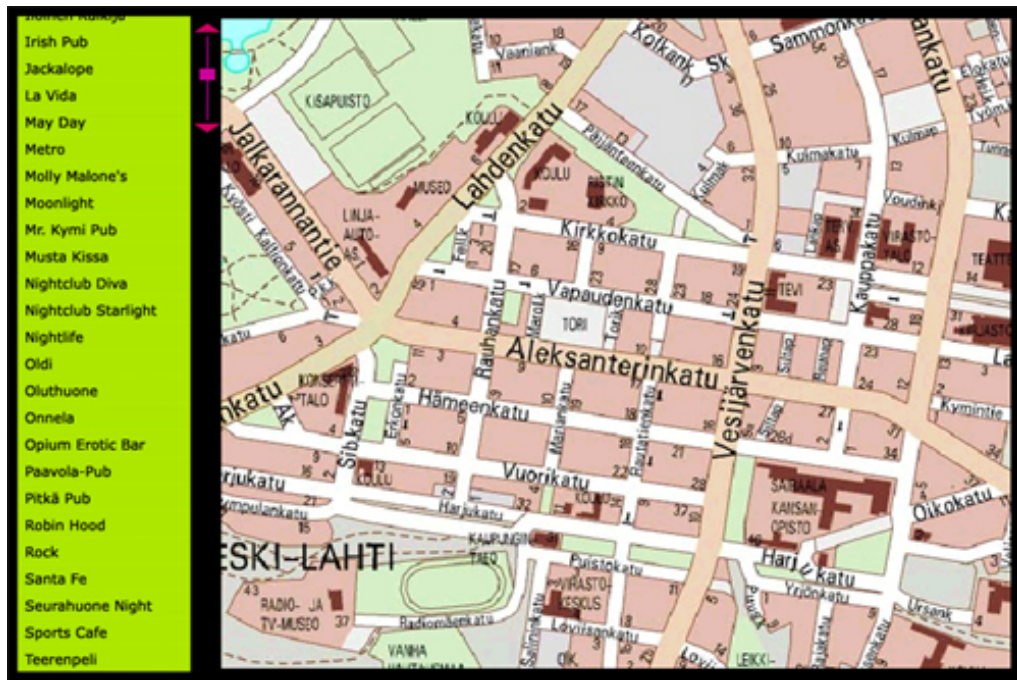
velussa nousivat baarien ajankohtaiset tiedot, kuten sijainti ja aukioloajat. Lisäksi yleinen keskustelu ravintoloista haluttiin mahdollistaa.

WWW-dokumenttien sisältö ja suhteet mietittiin ennen työn aloittamista. PHP:n avulla dokumenttien määrää saatiin pienennettyä. Tietokantaan tallennettava tieto mietittiin, minkä perusteella suunniteltiin tietokannan rakenne. Sivuston hakupalvelun takia esimerkiksi ravintoloiden osoitetiedot sijoitettiin erillisiin kenttiin, jotta haun voi suorittaa vaikka katuosoitteen perusteella. Tietokanta on kuitenkin niin pieni ja yksinkertainen, että taulujen määräkin jäi vähäiseksi.

Sivuston käyttöliittymän rakenne haluttiin yksinkertaiseksi ja helppokäyttöiseksi kaikenlaisille Internetin käyttäjille. Linkkejä on vähän, ja ulkoasusta on yritetty tehdä selkeä. Kirkkaat värit helpottavat hahmottamista. Heti ensimmäisellä sivulla, joka käyttäjälle aukeaa, on kerrottu palvelun tarkoitus ja perustoiminnot. Rakenne on perinteinen: otsikko on ylhäällä, linkit otsikon alapuolella ja keskellä sisältö. Oikeaan reunaan on lisätty linkkilista kaikista palveluun kuuluvista ravintoloista, jolloin niitä pääsee selaamaan aina halutessaan.

Palveluun päätettiin tehdä rekisteröitymismahdollisuus. Palvelua voi selata kuka tahansa, mutta keskusteluun osallistuminen ja muiden viestien jättäminen vaatii rekisteröitymisen. Tämän tarkoituksena on lähinnä keskusteluun osallistuneiden asiattomien kirjoitusten helpompi poisto ja jatkossa mahdollinen käyttäjäprofiilin määritysmahdollisuus. Tällä hetkellä palvelun käyttäjän nimimerkkiä eikä muita tietoja ilmene viesteissä, mutta kiinnostuksen lisääntyessä voisi olla hyvä, jos esimerkiksi ilmoitusten jättäjälle voisi vastata yksityisesti sivuston kautta ilman, että ilmoituksen jättäjä joutuu antamaan yhteystietojaan kaikkien nähtäväksi. Toisaalta palvelun tarkoituksena onkin sopia tapaamisia anonyymisti tiettyihin paikkoihin ”sokkotreffityyppisesti”, jolloin ilmoituksen jättäjän henkilöllisyyden paljastaminen jää kyseisen henkilön päätettäväksi.

Ravintoloiden sijainnit haluttiin esittää myös kartalla. Tähän tarkoitukseen tehtiin Flash-ohjelmalla Lahden keskustan alueen kartta, jossa näkyy valitun ravintolan sijainti. Karttapalvelulla on paljon jatkokehitysmahdollisuuksia, kuten erityylisten ravintoloiden erilaiset merkinnät ja valintamahdollisuuksien monipuolistaminen.



Kuva 7: Karttapalvelu Vapaa Lahti -sivustolla

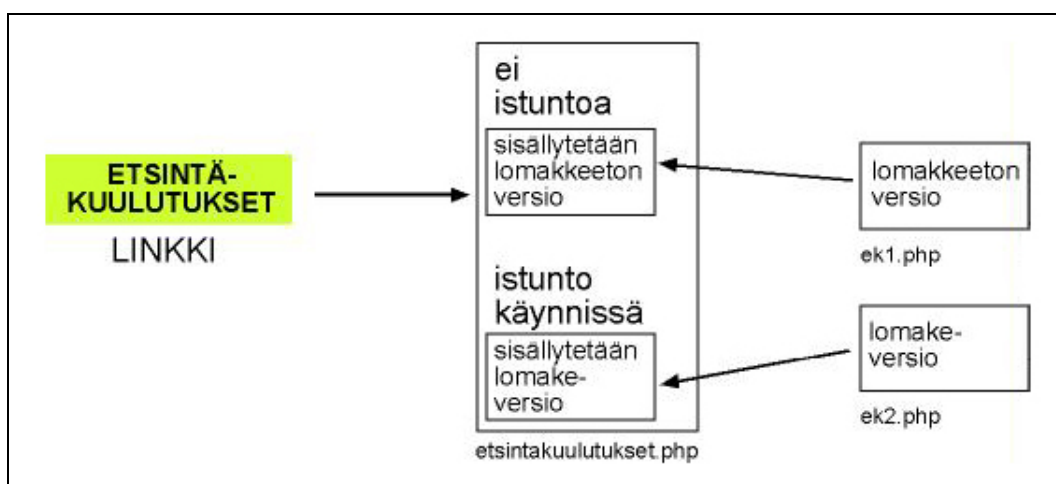
Sivusto on tehty HTML-merkintäkielellä, jonka sisään on upotettu PHP-kieltä. Yksittäiset sivut ovat lähes kaikki *.php-muotoisia, koska tietokannasta tulevaa tietoa tai muuta PHP-koodia on lähes joka sivulla. Tietokannaksi valittiin MySQL sen ilmaisuuden ja ominaisuuksien takia. Tiedostot on kirjoitettu Crimson Editor -nimisellä tekstieditorilla.

Palvelun kehitys- ja toteutusvaiheessa sivusto sijaitsi kotikoneelle asennetulla Apache-palvelimella. Palvelimelle asennettiin myös PHP. Näin PHP-tiedostot saatiin näkymään oikein. Tietokantaohjelmisto MySQL asennettiin myös samalle palvelinkoneelle. Tietokantaa hallittiin phpMyAdmin-ohjelman avulla.

6.2.2 Sivuston rakenne

Koska aiheen opiskelu jatkui vielä sivustoa suunniteltaessa ja tehdessä, ei rakenteesta tullut paras mahdollinen. Sivuston dokumenttimäärän kasvessa huomattiin, että paremmilla PHP-tiedoilla ja -taidoilla rakenteesta olisi saanut paljon järkevämmän, ja työmäärä olisi vähentynyt huomattavasti. Myös virheiden korjaaminen tai muutosten tekeminen olisi paljon helpompaa ja nopeampaa, jos PHP:tä olisi hyödynnetty enemmän ja sivustosta olisi tehty dynaamisempi.

Sivuston rakenne jakautuu kahteen osaan: kirjautuneiden käyttäjien dokumentit ja yleiset dokumentit. Kun käyttäjä kirjautuu palveluun, PHP sisällyttää dokumentteihin versiot, joissa on lomakkeet viestien lähetystä varten. Muuten sisällytetään lomakkeettomat versiot. Sisällytettävät lomakkeelliset WWW-dokumentit laitettiin erilliseen hakemistoon, joka suojattiin Htaccess-tiedostolla. Kuvassa 8 selvennetään tätä sivunmuodostusmenetelmää. Kun jokaisella ravintolalla on oma sivunsa, ja jokaisesta niistä kaksi versiota, kasvaa sivumäärä melkoiseksi verrattuna siihen, että saman voisi toteuttaa yhdellä dokumentilla, jossa linkin mukana lähetetyn `$_REQUEST`-muuttujan perusteella sisällytettäisiin dokumenttiin oikeat tiedot tietokannasta. Sivusto kuitenkin toimii halutulla ja tarkoituksenmukaisella tavalla, joten rakennetta muutetaan myöhemmin, jos tarve ilmenee.



Kuva 8: Esimerkki sivuston rakenteesta

6.2.3 Rekisteröityminen, kirjautuminen ja istunnot

Palvelun käyttäjäksi rekisteröidyttyään käyttäjä voi osallistua viestien kirjoittamiseen. Rekisteröitymiseen tehtiin HTML-lomake, jossa pyydetään käyttäjän etunimi, sukunimi, syntymävuosi, kotikunta ja valinnainen käyttäjätunnus ja salasana. Tietojen asianmukaisuus tarkistetaan ennen tietojen lähetystä PHP:n avulla.

Tyhjiä kenttiä ei saa olla, ja tämä tarkistetaan *if*-lauseen ja *empty*-funktion avulla, jossa kerrotaan HTML-lomakkeen kentän nimi. Jos kenttä on tyhjä, pyydetään käyttäjää palaamaan takaisin lomakkeen täyttöön.

```

if (empty ($_POST['etunimi']) )
{
die ("Jätit tyhjiä kenttiä. Anna etunimesi rekisteröityäksesi.");
}

```

Käyttäjätunnuksen täytyy olla yksilöllinen, joten MySQL-tietokannassa tunnukselle varattu kenttä on määritelty ainutlaatuiseksi avaimeksi. Käyttäjätunnuksen pituus on myös määritelty korkeintaan 20 merkkiä sisältäväksi. Tunnuksen pituus lasketaan PHP:n *strlen()*-funktiolla. Jos käyttäjän valitsema tunnus on jo käytössä, ei *INSERT*-lauseen suoritus onnistu, ja käyttäjää pyydetään palaamaan takaisin ja valitsemaan uusi tunnus. Tämä suoritetaan lisäämällä MySQL-kyselyn suorituskomennon perään käyttäjälle näytettävä teksti, jos tietokantaan tallennus ei onnistu.

```

elseif (strlen ($_POST['tunnus']) > 20)
{
echo ("Tunnus max. 20 merkkiä!");
exit;
}

```

```

$stallennus = "INSERT INTO rekisteroityneet VALUES ('$etunimi',
'$sukunimi', '$syntymavuosi', '$paikkakunta', '$tunnus',
md5('$salasana'))";

```

```

mysql_query ($stallennus) or die ("Käyttäjätunnus on jo käytössä.
Valitse uusi.");

```

Salasanan pituuden tarkistus suoritetaan samalla tavalla kuin käyttäjätunnuksen kohdalla, mutta siihen lisätään myös minimipituus. Salasanan tallennuksessa käytetään *md5()*-funktiota, jonka avulla salasana tallentuu tietokantaan 32-merkkisessä kryptatussa eli salatussa muodossa. Tämä estää väärinkäytön, jos tietokantaan murtauduttaisiin. Myöskään tietokannan haltija ei näe käyttäjän määrittelemää salasanaa.

Rekisteröityminen on voimassa heti. Rekisteröitymisen jälkeen palveluun täytyy kirjautua. Kirjautumismahdollisuus löytyy etusivun lisäksi muiltakin pääsivuilta. Käyttäjä antaa määrittelemänsä käyttäjätunnuksen ja salasanan HTML:llä tehtyyn lomakkeeseen. Istunnon käynnistäminen, tarkistus ja lopettaminen suoritetaan sivustolla siten, että ensimmäiseksi tarkistetaan, onko kirjautumislomakkeelta tullut tietoa. Jos on, otetaan käyttäjän antama tunnus sekä salasana talteen PHP:n muuttujiin. Salasana kryptataan MD5-muotoon. Seuraavaksi luodaan istuntokohtainen ainutlaatuinen tunnus, myös MD5-muodossa. Tämä lisätään käyttäjän tietoihin tietokantaan. Jos

tietokantaan tallentaminen onnistuu, eli käyttäjän antamat tunnukset löytyvät tietokannasta ja istuntotunnus tallentuu, annetaan istuntotunnus istunnolle. Jos tunnuksia ei löydy tietokannasta, ohjataan käyttäjä takaisin kirjautumislomakkeelle.

Istuntotunnuksella tarkistetaan jokaisella sivulla, onko käyttäjä kirjautunut. Käyttäjän rekisteröitymisessä määrittelemä käyttäjätunnus näkyy sivustolla istunnon ajan. Se saadaan näkyviin hakemalla istuntotunnuksen avulla käyttäjän tiedot tietokannasta ja tulostamalla käyttäjätunnus sivulle. Kirjaututtaessa ulos istuntotunnus nollataan ja istunto tuhoetaan. Istunto loppuu myös selaimen sulkeuduttua.

6.2.4 Hakutoiminto

Palveluun tehtiin hakumahdollisuus tietokannan tiedoista. Haussa voi hakea ravintoloiden tietoja nimen, katuosoitteen tai postinumeron perusteella. Haku onnistuu myös nimen tai katuosoitteen osalla. Esimerkiksi, kirjoittamalla katuosoitekenttään "alek", hakee palvelu kaikki ravintolat joiden katuosoitteessa on kyseinen hakuehto, eli ainakin kaikki ravintolat, jotka sijaitsevat Aleksanterinkadulla.

Hakutoiminnon toteutuksessa tehtiin ensin HTML-lomake, jossa käyttäjä antaa hakuehdot. Lomakkeiden kenttien nimet on koodissa nimetty "nimi", "katuosoite" ja "postinumero". Kun käyttäjä lähettää lomakkeen, siirrytään PHP-sivulle, jossa tarkistetaan *if*-lauseella mistä kentästä hakusana tulee. Jos hakusana tulee nimi-kentästä, tehdään kysely ravintoloiden tiedot sisältävän taulun nimikentästä käyttäjän antamalla hakusanalla. Sivun tulostaa kaikkien hakuehdon täyttävien ravintoloiden tiedot WWW-dokumenttiin HTML-*taulukko*n. Samalla tavalla *if*-lause toimii, jos hakulomakkeella lähetetty tieto on tullut HTML-lomakkeen katuosoite- tai postinumerokentästä, ja hakee tiedot joko tietokannan taulun katuosoite- tai postinumerokentästä.



Kuva 9: Sivuston hakutoiminto

6.3 Julkaisu

Ennen julkaisua palvelun toimivuutta testattiin eri selaimilla: Internet Explorerilla, Mozilla Firefoxilla, Operalla ja Netscapella. HTML- ja CSS-kielet luetaan selaimessa, ja selaimet puuttuvat mahdollisiin virheisiin eri tavoin. Jos HTML-koodia käytetään dokumentissa paljon eikä koodi ole täysin validia eli käytettävän HTML-version mukaisesti oikeellista, tulokset saattavat olla todella erinäköisiä esimerkiksi Internet Explorerilla ja Netscapella katsottuina. Validiin HTML-koodiin on tarkoituksenmukaista pyrkiä, ja tarkistukseen voi käyttää validaattoria eli dokumentin oikeellisuuden tarkastavaa erillistä ohjelmaa.

Kun sivusto oli saatu toiminnallisesti ja ulkonäöllisesti julkaisukelpoiseen kuntoon, se siirrettiin julkiselle palvelimelle. Tätä varten oli tilattu WWW-hotellipalvelu, johon kuului tarpeeksi levytilaa, MySQL-tietokantoja, WWW-pohjainen tietokantojen hallintasovellus sekä PHP-tuki. WWW-hotellin tilauksen yhteydessä rekisteröitiin verkkotunnus. Tunnuksen päätteeksi valittiin yksityishenkilöille vapautunut fi-verkkotunnus. Sivuston osoitteeksi tuli siten ”www.vapaa-lahti.fi”.

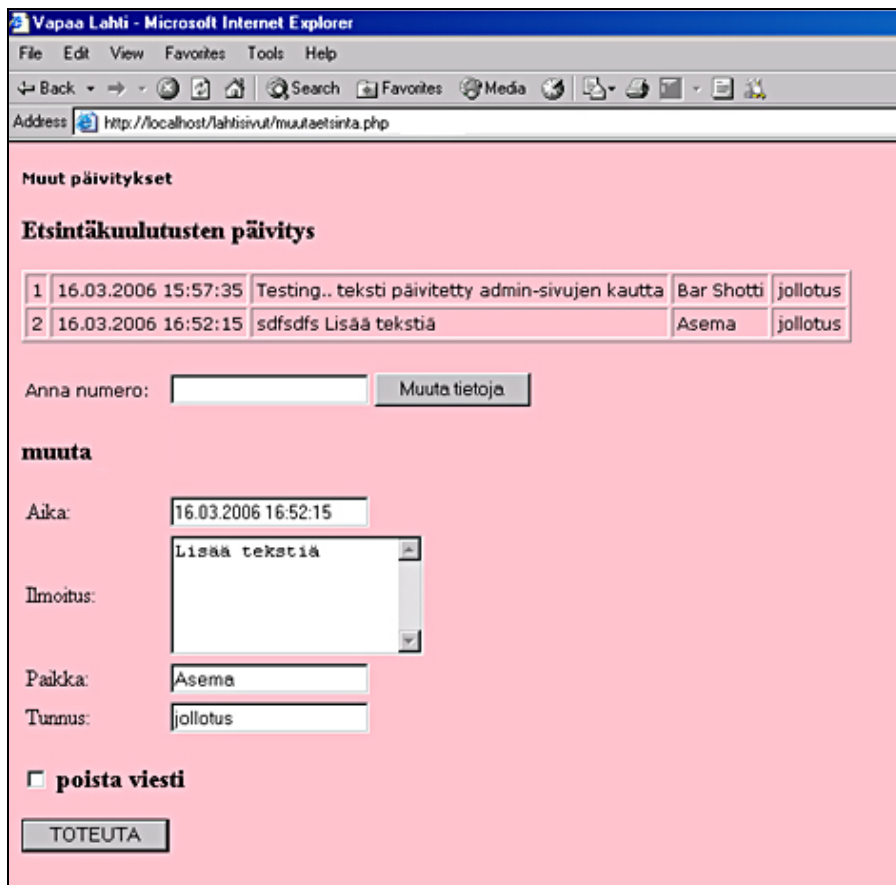
Tietokannan sisältö siirrettiin uudelle palvelimelle. Kopiointi onnistui helposti phpMyAdmin-ohjelmalla, jolla suoritettiin ensin kotikoneen palvelimella sijainneen tietokannan rakenteen ja sisällön kopiointi. Ohjelma teki näistä tekstimuotoisen SQL-kyselyn, joka annettiin uuteen tietokantaan phpMyAdmin-ohjelmalla lisäämällä teksti SQL-kyselyille tarkoitettuun kohtaan. Tietokannan nimen eteen tuli uudella palvelimella verkkotunnuksen kahdeksan ensimmäistä kirjainta, joten muutoksia täytyi tehdä PHP-dokumenteissa funktioon, joka ottaa yhteyden MySQL:ään ja valitsee tietokannan. Koska yhdistäminen tietokantaan oli erillisessä PHP-dokumentissa, joka sisällytetään tarvittaessa varsinaiseen dokumenttiin, muutos piti tehdä vain yhteen kohtaan.

Julkaisun yhteydessä otettiin käyttöön uusi sähköpostiosoite. WWW-hotellipalveluun kuului rajattomasti sähköpostitilejä. Palvelun yhteyteen tarvittiin kuitenkin tässä vaiheessa vain yksi osoite, johon voi lähettää tietoja virheellisistä tiedoista sivustolla sekä yleistä palautetta. Jatkossa sähköpostiosoitteita voi luoda lisää tarpeen mukaan. Sähköpostitilien luominen ja hallinta onnistuu WWW-hotellipalvelun mukana tulleen cPanel-ohjelman avulla. Sähköpostiosoitteelle määritellään tunnus, loppuosana on verkkotunnus. Palautepostin osoitteeksi tuli palaute@vapaalahti.fi.

6.4 Ylläpito

Sivuston tietokannan ylläpitoa helpottamaan tehtiin erillinen WWW-pohjainen ylläpito-osuus. Ylläpitosivuilla voi nopeasti ja miltä tahansa Internetiin yhteydessä olevalta tietokoneelta poistaa tai muuttaa asiattomia viestejä keskustelu-, seurailmoitus- ja etsintäkuulutustauluista.

Sivujen toiminta on toteutettu siten, että ensin haetaan valitun taulun tiedot tietokannasta ja tulostetaan ne WWW-sivulle HTML-taulukkoon. Silloin nähdään muutosten ja poistojen tarve, ja tällaisia sisältävien tietueiden järjestysnumero. Numero annetaan sivulla olevaan HTML-lomakkeeseen. Lomakkeen lähetyksen jälkeen WWW-dokumentti hakee annetun numeron perusteella oikean tietueen HTML-lomakkeeseen, jossa tietoja voi muuttaa. Lomakkeessa on myös valintaruutu, joka rastitettaessa antaa lähetyksen jälkeen käskyn poistaa valittu tietue tietokannasta kokonaan. Muutoslomakkeen lähetyksen jälkeen siirrytään sivulle, jossa varsinaiset muutokset suoritetaan ottamalla yhteys tietokantaan, liittämällä muutoslomakkeelta tulleet tiedot SQL-kyselyihin ja lähettämällä kyselyt MySQL-tietokantaan.



Kuva 10: Tietokannan WWW-pohjainen ylläpitolomake

Ylläpitosivut on tarkoitettu vain henkilölle, jolle on annettu oikeudet tietokannan tietojen muokkaamiseen. Muilta henkilöiltä pääsy ylläpitosivuille pitää estää. Tämä toteutettiin suojaamalla salattavat WWW-dokumentit sisältävä kansio Htaccess-tiedostolla. Htaccess on Apache-palvelimen asetustiedosto, jolla voidaan estää WWW-hakemistoon pääsy ilman käyttäjätunnusta ja salasanaa. Htaccess-tiedostossa määritellään käyttäjät, käyttäjälle näkyvässä kirjautumislomakkeessa näkyvä teksti sekä tiedostopolku salasanatiedostoon. Htaccess-tiedosto lisätään suojattavaan hakemistoon muodossa .htaccess, eli tiedostolle ei anneta nimeä. Salasanatiedosto eli .htpasswd-tiedosto sisältää käyttäjätunnuksen ja salasanan. Salasana on tiedostossa kryptatussa muodossa.

Palvelun ylläpitoon kuuluu lähinnä tietokannan ylläpito. Jos palveluun sisällytetään uusia ravintoloita, täytyy HTML-koodiin puuttua. Koska palvelun sisältö tulee suurimmaksi osaksi tietokannasta, muuta ei tarvitse perusylläpidossa muokata. Tietokantaa ylläpidetään phpMyAdmin-ohjelmalla, joka tuli WWW-hotellin tilauksen yhteydessä. Ulkonäölliset ja toiminnalliset ominaisuudet on yritetty ennen julkaisua saada sellaisiksi, ettei niitä tarvitsisi enää

muuttaa. Jos toiminnallisia ominaisuuksia ilmenee käytön yhteydessä, täytyy sivuston rakennetta harkita uudelleen.

Tietokannan sisältämät tiedot sijaitsevat ainoastaan WWW-palvelimella, joten niistä kannattaa ottaa varmuuskopioita. Valittu Internet-palveluntarjoaja tekee sivuston tiedoista varmuuskopiot kerran viikossa, mutta ei takaa niiden virheetöntä palauttamista. Kannattaa siis säilyttää tärkeät tiedot ja dokumentit itselläänkin. WWW-hotellipalvelun mukana tulleen cPanel-nimisen sivustonhallintaohjelman kautta tietokannan varmuuskopiointi sekä varmuuskopiointin palauttaminen palvelimelle onnistuu helposti.

6.5 Palvelun käyttö

Palvelun käyttäjäksi on rekisteröidyttävä, jos tahtoo osallistua viestien kirjoittamiseen. Rekisteröitymisen yhteydessä käyttäjän on annettava oma etu- ja sukunimensä, syntymävuotensa ja asumiskuntansa. Annettuja tietoja ei kuitenkaan voida mitenkään todentaa oikeiksi, mutta se ei ole olennaista, koska varsinkaan nimitiedoilla ei ole mitään jatkokäyttöä. Mitkään käyttäjän antamat tiedot eivät tule kenenkään asiaankuulumattoman henkilön tietoon. Käyttäjä määrittelee itse korkeintaan 20-merkkisen käyttäjätunnuksensa sekä 5-10 -merkkisen salasanan. Käyttäjätunnus on uniikki. Salasana tallennetaan tietokantaan kryptatussa muodossa, joten tietokannan haltijakaan ei tiedä sitä.

Käyttäjän kirjautuessa sivustolle tallentuu tietokantaan käyttäjän tiedot sisältävään tauluun istuntokohtainen tunnus sekä käyttäjän IP-osoite. Jättäessään viestejä sivustolle, tallennetaan valittuun tauluun myös käyttäjän käyttäjätunnus, vaikkei sitä sivustolla näkyvässä viestissä näytetäkään. Näin ollen tietokannan haltija näkee, jos joku tietty käyttäjätunnus aiheuttaa keskusteluissa tai viestien kirjoituksessa häiriötä. Jos tietystä käyttäjästä aiheutuu useasti häiriötä, voidaan hänen IP-osoitteestaan estää pääsy palveluun. Tämä on toteutettu Htaccess-tiedostolla. Htaccess-tiedostossa voidaan määrittellä tietyt IP-osoitteet, joilta pääsy sivustolle estetään, ja sallia pääsy kaikilta muilta.

Jätettäessä viestiä keskusteluihin, etsintäkuulutuksiin tai seuranhakuilmoituksiin, täytyy käyttäjän täyttää kyseisen HTML-lomakkeet kentät asianmukaisesti. Tyhjiä viestejä ei lähetetä. Lomakkeen täyttö tarkistetaan PHP:n avulla.

6.6 Laajennusmahdollisuudet ja kehitysvaihtoehdot

Palvelusta muodostui suurimmaksi osaksi tarkoituksellisesti mutta osittain myös puutteellisten ohjelmointitaitojen takia hyvin yksinkertainen. Jatkokehitykselle on kuitenkin olemassa hyvät lähtökohdat.

Mikäli kiinnostusta ilmenee, ravintoloita voidaan lisätä palveluun myös keskustan ulkopuolelta. Jos palvelua aletaan mainostaa ja sivustolla alkaa käydä vierailijoita, täytyy tiedot joka tapauksessa pitää ajantasaisina. Keskustan alueella tapahtuu ravintoloiden kohdalla muutoksia kuitenkin, joten sivujen lisääminen palveluun ei ole erityisen vaivalloista.

Palveluun voisi lisätä valokuvia joko erillisen kuvagallerian muodossa tai ravintolakohtaisesti jokaisen ravintolan omalle sivulle. Tässä on kuitenkin omat vaikeutensa ihmisten yksityisyyden suojan säilyttämisessä. Valokuvissa ei saisi ihmisiä tunnistettavasti näkyä, joten kuvaaminen olisi hankalaa. Yksi vaihtoehto olisi sisällyttää käyttäjien lähettämiä valokuvia palveluun, mutta myös siitä voisi koitua yksityisyyden suojaan liittyviä ongelmia.

Tietokantaan voisi tallentaa enemmän tietoja rekisteröityneen käyttäjän palvelunkäytöstä. Hyödyllistä olisi, jos käyttäjä voisi itse poistaa omat viestinsä ilmoituspalstoilta niiden vanhennuttua. Tämänhetkisellä ratkaisulla voi ilmentyä ongelmia viestien säilyttämisen keston suhteen.

Ongelmana saattaa myös olla jätettyihin ilmoituksiin vastaaminen. Jos viestin kirjoittajalle ei voi vastata suoraan, täytyisi mahdollisen vastaajan jotenkin pystyä yksinkertaisesti osoittamaan, kelle tai mihin viestiin hän vastaa. Tämä asia täytyisi korjata. Yhtenä lisäpalveluna sivustolle voisi lisätä PHP-pohjaisen keskustelualueen, jossa voisi reaaliaikaisesti vaihtaa ajatuksia ja hakea seuraa, ja jossa viestejä ei tallennettaisi tietokantaan.

Karttapalvelussa on paljon kehittämisen ja laajentamisen varaa. Kartassa voisi olla erilaiset symbolit erityyppisille ravintoloille, esimerkiksi yökerhoille omansa ja baareille omansa. Kartalle voisi myös hakea merkintöjä ravintolan tyyppin mukaan tai näyttää kaikki ravintolat. Karttaan voisi lisätä suurennusmahdollisuuden.

Yksi kehitysvaihtoehto on muodostaa rekisteröityneille käyttäjille itse muokattavat profiilit, jotka valinnaisesti voidaan asettaa muiden käyttäjien nähtäviksi. Profiilissa voisi olla esimerkiksi käyttäjän ikä, kuva ja tietoja harrastuk-

sista ja mielenkiinnon kohteista. Profiilin kehitys kuitenkin riippuu, niin kuin moni muukin kehityksellinen seikka, ainakin mahdollisten käyttäjien ikäkaumasta sekä rekisteröityneiden henkilöiden määrästä.

7 YHTEENVETO

Dynaamiset WWW-sivut ja tietokanta ovat toimiva yhdistelmä. Käyttötarkoituksia on paljon, ja niitä keksitään koko ajan lisää. Ilmaiset, avoimeen lähdekoodiin perustuvat tuotteet ovat toimivia ja tehokkaita. PHP on ohjelmointikielenä helppokäyttöinen ja monipuolinen. MySQL-tietokantaohjelmisto on suositeltava vaihtoehto yksityishenkilöiden ja yritysten WWW-sivuston tietokantaratkaisuksi, koska se on suorituskykyinen ja luotettava.

Kotikoneella tehtävä tietokantapohjaisen WWW-palvelun kehitystyö onnistuu helposti muuttamalla kotikone palvelimeksi Apache-palvelinohjelmistolla sekä asentamalla palvelimelle MySQL-tietokantaohjelmisto sekä PHP.

PHP-kielen ominaisuuksiin kannattaa tutustua tarkasti ennen palvelun toteuttamisen aloittamista. Työmäärää voi huomattavasti vähentää hyödyntämällä dynaamisuutta sekä toteutuksessa että ylläpidossa. Sivuston rakenteen toteuttamiseen on yleensä monta vaihtoehtoa, ja hyvällä suunnittelulla ratkaisusta saadaan toimiva ja helppokäyttöinen.

Suurempia tietomääriä käsiteltäessä tietokantojen suunnittelu on tärkeää. Tietokannan ylläpito on vaivattomampaa ja hausta saadaan nopeampi ja varmempi, jos tieto on järjestetty järkevästi ja tieto on ristiriidatonta ja eheää.

Tietokantojen hallintaan kannattaa käyttää graafista hallintasovellusta. Tietokannat ja niissä olevat taulut on helpompi hahmottaa, ja tietojen hallinta on helpompaa ja nopeampaa. Myös varmuuskopiointi ja varmuuskopioiden palauttaminen sekä oikeuksien hallinta ovat yksinkertaisia toimenpiteitä. SQL-kielen taitoa vaaditaan kuitenkin WWW-dokumentteihin kirjoitettavissa SQL-kyselyissä.

LÄHTEET

Kirjalliset lähteet:

Gilmore, W. Jason. 2005. PHP & MySQL : Tehokas hallinta. Readme.fi, Helsinki.

Heinisuo, Rami. 2003. PHP ja MySQL: Tietokantapohjaiset verkkopalvelut. 2. painos. Korkeakoulu-sarja. Talentum Media Oy, Helsinki.

Hernandez Michael J. 2000. Tietokannat : suunnittelu ja toteutus. Edita, IT Press, Helsinki

Hovi, A., Huotari, J., Lahdenmäki, T. 2005. Tietokantojen suunnittelu & indeksointi. 2. laitos, 1. painos. Peruskirjat. Docendo Finland Oy, Jyväskylä.

Meloni, Julie. 2003. MySQL Trainer Kit. Trainer Kit. IT Press, Helsinki.

Rantala, Ari. 2005. Web-ohjelmointi. Peruskirjat. Docendo Finland Oy, Jyväskylä.

Sähköiset lähteet:

2kmediat. SQL [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. <http://www.2kmediat.com/sql/alkeet.asp>.

2kmediat. Apache [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. <http://www.2kmediat.com/apache/apachehistoria.asp>.

MySQL AB. Top Ten Reasons to Use MySQL [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. <http://www.mysql.com/why-mysql/toptenreasons.html>.

MySQL AB. Licensing Policies [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. <http://www.mysql.com/company/legal/licensing/>.

MySQL AB. MySQL 5.0 Now Available for Production Use [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. http://www.mysql.com/news-and-events/news/article_976.html.

PHP. Introduction [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. <http://fi2.php.net/manual/en/introduction.php>.

PHP. History of PHP and related projects [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. <http://fi2.php.net/manual/fi/history.php>.

PHP. PHP-käsikirja [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. <http://fi2.php.net/manual/fi/function.mysql-real-escape-string.php>.

Zend - The php Company. What's New in PHP 5 [verkkodokumentti]. 2006 [viitattu 27.3.2006]. Saatavissa: liite-CD. <http://www.zend.com/php5/and-book-excerpt.php>.

LIITTEET

LIITE 1: Kuva Vapaa Lahti -sivuston etusivusta

LIITE 2: Kuva Vapaa Lahti -sivuston rekisteröitymissivusta

LIITE 3: Kuva Vapaa Lahti -sivuston seuranhakuilmoitusivusta

LIITE 4: Kuva Vapaa Lahti -sivuston seuranhakuilmoitusivusta (kirjaututtu)

LIITE 5: Kuva Vapaa Lahti -sivuston etsintäkuulutussivusta

LIITE 6: Kuva Vapaa Lahti -sivuston etsintäkuulutussivusta (kirjaututtu)

LIITE 7: Kuva Vapaa Lahti -sivuston hakusivusta

LIITE 8: Kuva Vapaa Lahti -sivuston hakutulossivusta

LIITE 9: Kuva Vapaa Lahti -sivuston ravintolasivusta

LIITE 10: Kuva Vapaa Lahti -sivuston ravintolasivusta

LIITE 11: Liite-CD, sisältö:

Opinnäytetyö (PDF-tiedosto)

Tiivistelmä (Word-asiakirja)

Englanninkielinen abstrakti (Word-asiakirja)

Sähköiset lähteet

Vapaa Lahti - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print Mail News RSS

Address http://www.vapaahti.fi/index.php

Vapaa Lahti

Tunnus: Salasana: [Kirjaudu!](#) [Rekisteröidy!](#)

[ETUSIVU](#) • [SEURANHAKU-ILMOITUKSET](#) • [ETSINTÄ-KUULUTUKSET](#) • [HAKU](#) • [KARTTA](#)

TERVETULOA.

Vapaa Lahti -sivujen tarkoitus on antaa lahtelaisille ravintoloiden käyttäjille uusia mahdollisuuksia, palveluita sekä tietoja Lahden iltaelämästä.

Sivuille voi jättää **seuranhakuilmoituksia, etsintäkuulutuksia tai kommentteja Lahden ravintoloista ja baareista.**

Jokaisen ravintolan kohdalta löytyy kyseistä paikkaa koskeva keskustelu. Etsintäkuulutuksia ja seuranhakuilmoituksia voi selata ylälinkkien kautta. Jos siis vaikkapa olet tavannut viime viikonloppuna mukavan henkilön mutta puhelinnumero jäi pyytämättä, käy **Etsintäkuulutukset**-sivulla jättämässä ilmoitus. Tai, jos olet ensi viikonloppuna menossa yksin uuteen baariin, kerro se **Seuranhaku**-sivulla muillekin yksinäisille ja sopikaa tapaaminen!

Ilmoitusten jättäminen ja keskusteluun osallistuminen vaatii rekisteröitymisen. Rekisteröityminen on ilmaista.

Mahdollisista virheellisistä tiedoista ym. voi ilmoitella osoitteeseen palaute@vapaahti.fi.

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhenoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluhuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

LIITE 2

LAHTISIVUT - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print Mail

Address http://www.vapalahti.fi/rekisteroidy.php

VaPaA LoHti

Tunnus: Salasana:

ETUSIVU • **SEURANHAKU-ILMOITUKSET** • **ETSINTÄ-KUULUTUKSET** • **HAKU** • **KARTTA**

Osallistuaksesi keskusteluun ja jättääksesi ilmoituksia sinun täytyy rekisteröityä palvelun käyttäjäksi. Täytä kaikki kentät ja keksi itsellesi käyttäjätunnus (max 20 merkkiä) sekä salasana (5-10 merkkiä). Tietosi eivät näy ilmoituksia/kommentteja jättäessäsi eikä niitä anneta eteenpäin. Ikä- ja paikkakuntatietoja käytetään palvelun kehittämiseen.

Etunimi

Sukunimi

Syntymävuosi

Paikkakunta

Tunnus

Salasana

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhenpoika
- HiiDENkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

Vapaa Lahti - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print

Address http://www.vapaalahti.fi/seuraa.php

Vapaa Lahti

Tunnus: Salasana:

ETUSIVU • SEURANHAKU-ILMOITUKSET • ETSINTÄ-KUULUTUKSET • HAKU • KARTTA

SEURANHAKUILMOITUKSET

Valitse ravintola:

Aika	Viesti	Paikka
01.05.2006 18:07:29	Seuranhakuilmoitukset, esimerkki 1	Asema

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhenoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

Vapaa Lahti

Olet kirjautunut, XXXXX, Kirjaudu ulos

ETUSIVU • SEURANHAKU-ILMOITUKSET • ETSINTÄ-KUULUTUKSET • HAKU • KARTTA

SEURANHAKUILMOITUKSET

Valitse ravintola:

Ilmoituksesi:

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhempoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

Vapaa Lahti - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print

Address http://www.vapaalahti.fi/ek3.php

Vapaa Lahti

Tunnus: Salasana:

[ETUSIVU](#) •
 [SEURANHAKU-ILMOITUKSET](#) •
 [ETSINTÄ-KUULUTUKSET](#) •
 [HAKU](#) •
 [KARTTA](#)

ETSINTÄKUULUTUKSET

Valitse ravintola:

Aika	Viesti	Paikka
01.05.2006 18:08:18	Etsintäkuulutus, esimerkki 1	Asema

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhenoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

Vapaa Lahti

Olet kirjautunut, XXXXX [Kirjaudu ulos](#)

ETUSIVU • SEURANHAKU-ILMOITUKSET • **ETSINTÄ-KUULUTUKSET** • HAKU • KARTTA

ETSINTÄKUULUTUKSET

Valitse ravintola:

Ilmoituksesi:

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhenpoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

Vapaa Lahti - Haku - Microsoft Internet Explorer

Address: http://www.vapaa-lahti.fi/haku.php

Vapaa Lahti

Tunnus: Salasana: [Kirjaudu!](#)

[ETUSIVU](#) •
 [SEURANHAKU-ILMOITUKSET](#) •
 [ETSINTÄ-KUULUTUKSET](#) •
 [HAKU](#) •
 [KARTTA](#)

HAKU

Voit hakea tietoja Lahden keskustan ravintoloista nimen, katuosoitteen tai postinumeron avulla. Hakusanaksi riittää myös osa ravintolan tai kadun nimeä.

Nimihaku:

Osoitehaku:

Postinumerohaku:

[Hae!](#)

- Alex Park
- Asema
- Ateljé Bar Tauli
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhenoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

Vapaa Lahti - Haku - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print

Address http://www.vapaahti.fi/hae.php

Vapaa Lahti

Tunnus: Salasana: Kirjaudu!

ETUSIVU • SEURANHAKU-ILMOITUKSET • ETSINTÄ-KUULUTUKSET • HAKU • KARTTA

HAKU

Haku katuosoitteen perusteella. Hakusana 'alek'.

Bar 31	Aleksanterinkatu 31, 15140, Lahti (03) 782 176
Bar Shotti	Aleksanterinkatu 20, 15140, Lahti (03) 781 107
Cheri	Aleksanterinkatu 25, 15140, Lahti su-ma 09-01, ti, to (03) 544 655 09-02, ke, pe-la 09-03
Grand Star Cafe	Aleksanterinkatu 23, 15110, Lahti
Hyvä Aatos	Aleksanterinkatu 30, 15140, Lahti
Irish Pub	Aleksanterinkatu 25, 15140, Lahti (03) 781 940
May Day	Aleksanterinkatu 3, 15110, Lahti (03) 734 851 ke, pe-la 20-03, ti, to, su 20-02
Oldi	Aleksanterinkatu 9, 15110, Lahti (03) 752 006
Onnela	Aleksanterinkatu 24, 15140, Lahti (03) 781 250 ke-la 22-04, su 23-04
Rock	Aleksanterinkatu 1, 15110, Lahti pe-la 22-04
Santa Fe	Aleksanterinkatu 10, 15110, Lahti
Seurahuone Night	Aleksanterinkatu 14, 15110, Lahti (03) 85 111 pe-la 22-04
Sports Cafe	Aleksanterinkatu 26, 15140, Lahti
Wanha Mestari	Aleksanterinkatu 23, 15110, Lahti (03) 783 916

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhempoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

Vapaa Lahti - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://www.vapaalahti.fi/asema.php

Vapaa Lahti

Tunnus: Salasana:

ETUSIVU • SEURANHAKU-ILMOITUKSET • ETSINTÄ-KUULUTUKSET • HAKU • KARTTA

ASEMA



Osoite: Jalkarannantie 1
15110, Lahti

Puhelinnumero: (03) 875 540

Aukioloajat: ma-to 06-22, pe 06-02, la 09-02, su 09-24

Kuvaus: Linja-autoaseman yhteydessä oleva kahvila/baari.

Hintaesimerkkejä: Olut (0,5 l) 3,80 €
Kahvi 1,50 €

KESKUSTELU:

Aika	Viesti
01.05.2006 18:10:28	Keskustelu, esimerkki 1

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhempoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)

Vapaa Lahti - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://www.vapaalahti.fi/asema.php

Vapaa Lahti

Olet kirjautunut, XXXXX [Kirjaudu ulos](#)

ETUSIVU • SEURANHAKU-ILMOITUKSET • ETSINTÄ-KUULUTUKSET • HAKU • KARTTA

ASEMA



Osoite: Jalkarannantie 1
15110, Lahti

Puhelinnumero: (03) 875 540

Aukioloajat: ma-to 06-22, pe 06-02, la 09-02, su 09-24

Kuvaus: Linja-autoaseman yhteydessä oleva kahvila/baari.

Hintaesimerkkejä: Olut (0,5 l) 3,80 €
Kahvi 1,50 €

KESKUSTELU:

Aika	Viesti
01.05.2006 18:10:28	Keskustelu, esimerkki 1

Viestisi:

- Alex Park
- Asema
- Ateljé Bar Taulu
- Bar 31
- Bar Shotti
- Cheri
- Grand Star Cafe
- Hanhenpoika
- Hiidenkivi
- Hyvä Aatos
- Iloinen Kulkija
- Irish Pub
- Jackalope
- La Vida
- May Day
- Metro
- Molly Malone's
- Moonlight
- Mr. Kymi Pub
- Musta Kissa
- Nightclub Diva
- Nightclub Starlight
- Nightlife
- Oldi
- Oluthuone
- Onnela
- Opium Erotic Bar
- Paavola-Pub
- Pitkä Pub
- Robin Hood
- Rock
- Santa Fé
- Seurahuone Night
- Sports Cafe
- Teerenpeli
- Tehobaari
- Ten Dollars Saloon
- Tikanpesä
- Tirra
- Torvi
- Valentina
- Vesku 36
- Virtanen
- Wanha Mestari (kesk.)