

TAMPEREEN AMMATTIKORKEAKOULU
Tietotekniikka, Ohjelmistotekniikka
Mika Manninen

Tutkintotyö

Mika Manninen

WEB-PORTAALIN SUUNNITTELU JA TOTEUTUS

Työn valvoja
Työn teettäjä

Ohjelmistotekniikan lehtori, Pekka Pöyry
Amurin Willa Isännöinti Oy, Kari Eerola

Tekijä: Mika Manninen
Työn nimi: Web-portaalin suunnittelu ja toteutus
Sivumäärä: 29 sivua ja 3 liitesivua

Työn ohjaaja: Ohjelmistotekniikan lehtori, Pekka Pöyry
Työn teettäjä: Amurin Willa Isännöinti Oy, isännöitsijä Kari Eerola
Päivämäärä: 13.4.2005
Avainsanat: PHP, MySQL, portaali, helppokäyttöinen

Internetin avulla tapahtuva tiedottaminen on helpottanut yritysten toimintaa monella tapaa. Yritystoiminnan kannalta ajateltuna tietoa voidaan jakaa informatiivisesti tai mainonnallisesti alasta riippumatta. Tässä työssä pyritään käyttämään Internetin tarjoamia kanavia tiedon jakamiseen.

Tutkintotyössä on pyritty toteuttamaan helposti käytettävä web-portaali, jonka avulla isännöinnissä käytettävien dokumenttien käsittely sähköisesti on entistä helpompaa. Portaali on toteutettu PHP- ja HTML-kielillä ja tietokantana on käytetty MySQL-tietokantaa.

Työssä on esitelty web-portaalin toimintaa. Käsiteltyjä järjestelmän kohteita ovat käyttäjän tunnistaminen, tiedostojen syöttäminen tietokantaan ja sinne syötettyjen tiedostojen hallitseminen. Lisäksi on käsitelty palautteen ja ilmoitusten antamista sekä niihin vastaamista.

Author:	Mika Manninen
Thesis name:	Web-portal planning and implementation
Page count:	29 pages + 3 appendix pages

Thesis supervisor:	Lector of Software Engineering, Pekka Pöyry
Commission company:	Amurin Willa Isännöinti Oy, superintendent Kari Eerola
Date:	13.4.2005
Keywords:	PHP, MySQL, portal, user friendly

Internet is able to deliver large amounts of information and it has eased companies in many ways. Information can be delivered informatively and with advertises in spite of business branch. In this study I am going to use Internet's channel's to deliver information.

I have tried to create a user friendly web-portal that helps superintendent to deliver documents electrically. The portal is made with PHP- and HTML-language and the database is MySQL.

In this study web-portal's operation is introduced. Objects of the system that are handled are user identification, sending files to the database and the control of the files in the database. Also sending feedback, giving announcements and answering to them are also handled.

ALKUSANAT

Työ sai alkunsa työharjoittelusta Tampereen Fujitsu Services Oy:ssä kesällä 2004, jolloin työtoverini, ohjelmistosuunnittelija Kimi Niittyniemi kertoi mahdollisesta tutkintotyötarjouksesta isännöintiin liittyvänä. Lyhyehkön palaverin jälkeen työn teettäjän, Kari Eerolan kanssa, että työ aloitetaan samana kesänä.

Tutkintotyön etenemistä helpotti Amurin Willa Isännöinti Oy:öön tilattu palveluntarjoaja Nebulan WebHotel Pro -webhotelli, joka sisälsi PHP-tulkin ja MySQL-tietokannan. Lisätietoa Nebulan palveluista saa osoitteesta <http://www.nebula.fi>.

Tutkintotyö on tarkoitettu Amurin Willa Isännöinti Oy:n tarpeisiin, mutta toteutuksessa on huomioitu myös tehdyn web-portaalin helppo muunneltavuus muiden isännöintitoimistojen käyttöön. Tutkintotyö otetaan käyttöön kesäkuussa 2005 osoitteessa <http://www.awioy.fi>.

Haluan kiittää Kimi Niittyniemeä suunnittelussa avustamisesta, Kari Eerolaa asiakasvaatimuksissa joustamisesta ja työn valvojaa, Pekka Pöyryä.

Tampereella huhtikuussa 2005

Mika Manninen

KÄYTETYT LYHENTEET

PHP	Hypertext preprocessor – ohjelmointikieli.
MySQL	Avoimen lähdekoodin ilmainen relaatiotietokanta.
HTML	Hypertext Markup Language – kuvauskieli.
HTTP-palvelin	Hypertext Transfer Protocol – palvelin, asiakkaalta tulee pyyntö www-sivusta, palvelin palauttaa sen.
GPL	General Public License – lisenssi varmistaa, että ohjelma on vapaa kaikille käyttäjille.
URL	Universal Resource Locator – Internet-sisältöjen paikkatietojärjestelmä.
SSH	Secure Shell – suojattu pääteyhteysohjelma.
Microsoft Word	Tekstinkäsittelyohjelma.
SQL	Structured Query Language – tietokannan kyselykieli.
IE	Internet Explorer – Internet-selain.
OPERA	Internet-selain.
FIREFOX	Mozilla Firefox, Internet-selain.
CSS	Cascading Style Sheets – tiedostoja, jotka sisältävät www-sivun ulkoasuun liittyviä määritteitä.
AES	Advanced Encryption Standard – kehittynyt salaus standardi
BLOB	Binary large objects – suuret binääriset objektit, käytetään tietokannoissa sarakkeen tietotyypinä

SISÄLLYSLUETTELO

TIIVISTELMÄ	I
ABSTRACT	II
ALKUSANAT	III
KÄYTETYT LYHENTEET	IV
1 JOHDANTO	1
2 TUTKINTOTYÖN MÄÄRITTELY	1
3 KÄYTETYT KEHITYSYMPÄRISTÖT	2
3.1 PHP-KIELI	2
3.2 MYSQL	3
4 TUTKINTOTYÖN TAVOITE JA TARKOITUS	4
5 TUTKINTOTYÖN TOTEUTUS	4
5.1 KÄYTETYT OHJELMAT	5
5.2 ASIAKASVAATIMUKSET	6
5.3 TOIMINTA	7
5.4 TESTAUS JA OHJEISTUS	9
6 TUTKINTOTYÖN TULOKSET	10
6.1 SIVUSTON TIETOTURVA	10
6.2 TIETOKANTA	11
6.3 SIVUSTO	15
6.3.1 Vieras	17
6.3.2 Asukas ja Osakas	19
6.3.3 Välitystoimiston käyttäjä	21
6.3.4 Huoltoyhtiön käyttäjä	22
6.3.5 Hallituksen jäsen	23
6.3.6 Pääkäyttäjä	25
6.4 ALKUASETUKSET JA YLLÄPITO	27
7 POHDINTA	27
LÄHTEET	29

1 JOHDANTO

Yritysten tarve sähköistää palveluitaan on lisääntynyt Internetin tarjoamien mahdollisuuksien kautta. Mainonnan ja informaation jakamisen eri keinoilla yritysten on mahdollista saada näkyvyyttä Internetin avulla. Tässä tutkintotyössä Internetin tarjoamia mahdollisuuksia käytetään isännöinti-toimiston eri toimintojen sähköistämiseen ja tarkoituksena on esittää helppokäyttöinen web-sivusto isännöintiin liittyvien toimintojen hallintaan.

Tutkintotyössä on keskitytty toteuttamaan selkeä ja toimiva ulkoasu web-sivustolle. Sivujen muodostamiseen käytetään rakennemäärittelyä sisältäviä CSS-tyylitiedostoja, koska haluttiin mahdollisuus muuttaa sivuston visuaalista ulkoasua tulevaisuudessa. Sivut on toteutettu PHP- ja HTML-kielellä ja taustalla on MySQL-tietokanta. Web-sivuston keskeisenä asiakasvaatimuksena oli toteuttaa sivuston käyttäjille roolit eri kokonaisuuksien rajoittamiseksi.

Ratkaisussa on keskitytty tiedostojen hallintaan ja erityisesti niiden näkyvyyteen sekä taloyhtiöiden tietojen selaamiseen. Työssä on paneuduttu myös asuntojen asukkaiden jättämiin vikailmoituksiin ja niihin vastaamiseen huoltoyhtiön toimesta. Palautteen käsittelyyn ja käyttäjien hallintaan on myös paneuduttu. Työssä on käytetty mahdollisimman paljon ilmaisia ja vapaan lähdekoodin ohjelmia kustannusten rajoittamiseksi.

2 TUTKINTOTYÖN MÄÄRITTELY

Tutkintotyönä toteutetaan web-sivusto isännöintiin liittyvien toimintojen helpottamiseksi ja se pyritään toteuttamaan ohjelmistotuotannossa käytetyn vesiputousmallin mukaisesti. Tutkintotyö etenee esitutkimuksesta määrittelyn kautta suunnitteluun, josta päästää toteutukseen ja testaukseen sekä viimein käyttöönottoon ja ylläpitoon. /2./ Sivusto ohjelmoidaan PHP- ja HTML-kielellä ja tietovarastona toimii MySQL-tietokanta. Sivujen rakenteen muodostamiseen käytetään CSS-tyylitiedostoja.

Sivusto voidaan jakaa julkiseen osaan ja rajoitettuihin osiin. Julkisessa osassa ei vaadita käyttäjän tunnistamista ja sisältö on vähäisempi. Rajoitetut osat on jaettu viiteen kategoriaan. Kategorioihin jakaminen on ratkaistu siten, että jokaisella rekisteröidyllä käyttäjätunnuksella on oma roolinsa, jonka perusteella käyttäjä ohjataan oikealle sivulle. Nämä viisi roolia ovat pääkäyttäjä, asukas ja osakas, välitystoimiston käyttäjä, huoltoyhtiön käyttäjä tai tietyn taloyhtiön hallituksen jäsen. Julkiselle sivulle kirjaututtaessa tai sivustolle rekisteröidytessä käyttäjällä on oletuksena vieraan rooli. Rekisteröityneiden käyttäjien rooli voidaan vaihtaa pääkäyttäjän toimesta. Sivuston keskeisiä toimintoja roolin mukaan ovat taloyhtiöiden esitteiden tilaaminen ja selaaminen, taloyhtiöiden asukkaiden vikailmoituksiin vastaaminen ja taloyhtiöiden hallituksen jäsenien asiakirjojen hallinnointi sekä niiden julkaiseminen. Pääkäyttäjän roolilla voidaan hallinnoida käyttäjiä ja näiden rooleja, lisäksi tiedotteiden julkaiseminen ja niiden näkyvyyden hallinnointi on mahdollista.

3 KÄYTETYT KEHITYSYMPÄRISTÖT

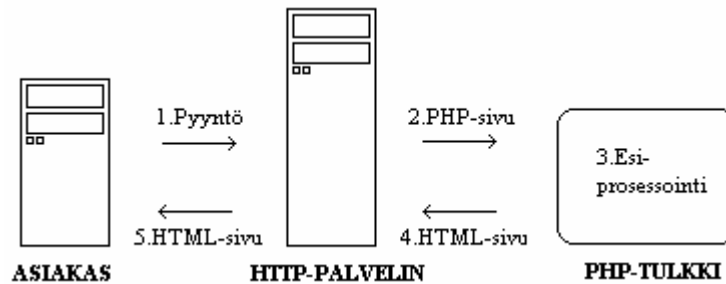
Tässä luvussa luodaan yleissilmäys käytettyihin kehitysyökaluihin. Linkit ympäristöihin löytyvät lähdeluettelosta. /10, 11./

3.1 PHP-kieli

PHP on yleiskäyttöinen skriptikieli, joka soveltuu web-sovellusten kehittämiseen. PHP:n syntaksi muistuttaa C-kieltä, Javaa sekä Perliä ja on oppimiskynnykseltään matala. PHP:n avulla voidaan kehittää jopa graafisia käyttöliittymiä ja komentorivisovelluksia, mutta parhaaseen tulokseensa se pääsee web-sivustoissa. PHP on lisenssivapaa ns. open source -tuote. Sitä siis voidaan käyttää maksutta myös kaupallisiin tarkoituksiin. PHP-ajoympäristö on saatavissa lähes kaikkiin yleisimpiin HTTP-palvelimiin ja usein PHP yhdistetäänkin Apache-palvelinohjelmistoon. /1, 6./

Ohjelmointikielenä PHP on helppo oppia. Se soveltuu hyvin aloittelijoille, mutta on myös edistyksellinen työkalu ammattilaisille. PHP sisältää tuen useimmille tietokannoille ja kielessä alkaa olla myös olio-ominaisuuksia. PHP:n viimeisin versio 5.0.3 sisältää olio-ominaisuudet ja on täten kehittynyt edeltäjiinsä verrattuna. Tässä tutkintotyössä on käytetty PHP:n versiota 4.3.10. /1, 6./

Seuraava kuva selittää PHP:n toiminnan yleisesti.



Kuva 1 PHP:n toiminta /6./

HTTP-palvelin käsittelee asiakkaalta tulevan pyynnön ja tunnistaa tiedoston päätteen perusteella PHP-sivun ja välittää sen PHP-tulkille. Tulkki kääntää sivun ja suorittaa tiedoston PHP-koodin. Lopuksi PHP-tulkki palauttaa HTTP-palvelimelle ainoastaan HTML-koodia, jonka palvelin välittää takaisin asiakkaalle. Asiakas ei näe PHP-koodia sivuston selaamisen yhteydessä missään vaiheessa. /1, 6./

3.2 MySQL

MySQL on monipuolinen, suorituskykyinen ja ilmainen relaatiotietokanta, jota käytetään yleisesti niin suurten kuin pientenkin web-palvelujen taustalla. Se on alun perin ruotsalaisen T.c.X Ab:n tuotantoa ja se oli tarkoitettu kyseisen konsultointiyrityksen sisäiseen käyttöön. Ilmainen tietokanta tarkoittaa tässä tapauksessa GPL-lisenssiä, jonka mukaan sitä saa käyttää vapaasti, mutta sitä ei saa levittää kaupallisesti. MySQL on saatavana useille eri käyttöjärjestelmille ja se on ilmaiseksi ohjelmaksi kohtuullisen hyvin tuotteistettu. /1, 6./

MySQL noudattaa asiakas-palvelin -arkkitehtuuria, jossa sovellukset eivät koskaan käsittele tietokantaa suoraan, vaan käsittelyt tapahtuvat aina palvelinohjelman kautta. MySQL-palvelimeen voidaan ottaa yhteyttä monella eri tapaa, joista yleisemmin käytettyjä ovat PHP-, Perl- ja Java-tekniikat. Tietokanta on helppo asentaa ja sitä on kevyt ylläpitää. Se ei vaadi toimiakseen jokapäiväistä ylläpitoa kalliiden tietokantasovellusten tapaan ja on näin mainio tietokanta pienten web-sovellusten tietovarastoksi. Yleisimmin MySQL:n avulla on toteutettu julkaisujärjestelmiä ja sähköisen kaupan järjestelmiä, kuten suosittu Internet-huutokauppa huuto.net. /1, 6./

4 TUTKINTOTYÖN TAVOITE JA TARKOITUS

Tutkintotyön tavoitteena on toteuttaa ja suunnitella selkeä ja helppokäyttöinen web-sivusto, jonka avulla isännöintiin liittyvät toiminnot helpottuvat.

Tarkoituksena on toteuttaa PHP-kielellä sivusto, jonka taustalla toimii MySQL-tietokanta tietovarastona. Asiakirjojen ja tiedotteiden hallintaan käytetään tätä tietovarastoa, jonne tiedostot tallennetaan binääritietona ja josta sivuston pääkäyttäjä voi julkaista tiedotteita.

Sivuston osia on tarkoitus rajoittaa käyttäjäkohtaisilla rooleilla, joiden perusteella käyttäjä ohjataan roolin oikeuttavalle sivulle. Sivuston toteutuksessa on pyritty huomioimaan mahdollisuus muokata sivustoa visuaalisesti tulevaisuuden jatkokehityksen ja mahdollisen myyntitarkoituksen mielessä.

5 TUTKINTOTYÖN TOTEUTUS

Tässä luvussa käsitellään toteutukseen liittyviä asioita, joilla on ollut vaikutuksia tutkintotyön lopputulokseen ja web-sivuston toimintaan. Tarkoituksena on selittää web-sivuston toimintaa esimerkkien avulla.

5.1 Käytetyt ohjelmat

Tutkintotyön raportoinnissa on käytetty Microsoft Word -tekstinkäsittely-ohjelman versiota 2002. Lähdekoodin kirjoittamisessa on käytetty web-kehitykseen suunnattua Maguma Studio -ohjelman versiota 1.3.3. /7./ Ohjelman voi rekisteröidä ja saada käyttöönsä ilmaisesta versiosta poistetut toiminnot korvausta vastaan, joita en kustannussyistä halunnut ottaa käyttöön.

Maguma Studiossa on hyödyllinen toiminto, jonka avulla tiedostot voidaan siirtää suoraan HTTP-palvelimelle. Toiminnon käyttö jäi vähäiseksi, koska käytin tiedostojen siirtoon SSH Secure Shell -ohjelmaa. /8./ Tässä ohjelmassa on tiedostojen siirron lisäksi myös terminaali-ikkuna, jonka avulla voidaan ottaa yhteys MySQL-tietokantaan ja ylläpitää sekä muokata tietokantaa ja sen tauluja. Tietokantaan syötetty testidata ja taulujenluomiskäskyt kirjoitettiin Microsoft Windows XP:n mukana tulevalla Notepad -muistiolla.

Tietokannan suunnittelussa ja mallintamisessa on käytetty ilmaista DBDesigner 4 -ohjelmiston versiota 4.0.5.6 beta. /9./ Ohjelman beta-versiosta huolimatta ohjelman toiminta oli moitteetonta eikä vikoja ilmaantunut käytön aikana. Ohjelmalla on MySQL:n tapaan GPL-lisenssi, jonka mukaan siis ohjelmaa saa käyttää vapaasti, mutta sitä ei saa levittää kaupallisesti.

Kehitysympäristönä toimii web-hotellipalveluita tarjoavan Nebulan Web-hotel Pro, joka tarjoaa PHP 4.3.10 ja MySQL 3.23.58 -versiot. Sivujen testaus suoritettiin varsinaisen osoitteen alihakemistossa www.awioy.fi/kehitysversio, josta valmiit sivut siirrettiin osoitteeseen www.awioy.fi. Testauksessa ja selaamisessa käytettyjä Internet-selaimia olivat Internet Explorer, Mozilla Firefox /13./ ja Opera /12./, joista kaikista oli käytössä viimeisin versio.

Ohjelmien linkit löytyvät lähdeluettelosta.

5.2 Asiakasvaatimukset

Web-sivuston asiakasvaatimukset pyrittiin kartoittamaan ja rajoittamaan järkeviksi projektin alkuvaiheessa. Pääasiallinen vaatimus oli saada sivuston käyttäjille eri roolit toimintojen rajoittamiseksi. Vaatimuksena oli myös, että taloyhtiöiden tietoja tulisi voida lukea roolista riippumatta. Roolien määräksi päätettiin 6 kappaletta.

1. Vieras
2. Asukas ja osakas
3. Välitystoimiston käyttäjä
4. Huoltoyhtiön käyttäjä
5. Hallituksen jäsen
6. Isännöintitoimiston pääkäyttäjä

Vierailijalla on mahdollisuus selata taloyhtiöiden talojen tietoja, katsoa julkaistuja dokumentteja ja ottaa yhteyttä isännöintitoimistoon. Kaikilla luetelluilla rooleilla on oletuksena samat toiminnot kuin vieraalla.

Asukkaalla tai osakkaalla on mahdollisuus jättää vikailmoituksia, kun taas välitystoimiston käyttäjällä on mahdollisuus tilata esitteitä taloyhtiöistä isännöintitoimistolta. Huoltoyhtiön käyttäjä voi tutkia päivämäärittäin asukkaiden tai osakkaiden antamia vikailmoituksia ja hän voi aloittaa vian korjaamisen sekä lopettaa aloitetun työn. Hallituksen jäsenellä on mahdollisuus julkaista ja selata omaan taloyhtiöön kuuluvia asiakirjoja, jotka näkyvät ainoastaan taloyhtiön hallituksen jäsenille. Isännöintitoimiston pääkäyttäjä voi julkaista tiedotteita tai tilauksia ja kaikkien käyttäjien hallinnointi on myös mahdollista. Kaikilla rooleilla on mahdollisuus muokata omia tietojään paitsi vierailija-roolilla.

5.3 Toiminta

Sivustolle voi rekisteröityä kuka tahansa ulkopuolinen, joka rekisteröityessään saa oletuksena vieraan roolin. Pääkäyttäjä voi muuttaa rekisteröityneen käyttäjän roolia. Kun ollaan kirjautumassa sivulle, syötetty tunnus tarkistetaan, ja jos tunnus on validi, päästetään käyttäjä sivustolle roolin mukaisissa rajoituksissa. Tarkastetun tunnuksen roolia kuljetetaan selaimen osoiterivillä niin kauan kuin käyttäjä on kirjautuneena sivustolle. Seuraava esimerkki selittää navigointilinkkien muodostumista sivustoon, jos käyttäjän rooli on pääkäyttäjä. Muiden roolien osalta linkkien generoituminen on vastaavanlaista.

```
<?php
//Linkkivalikoima generoidaan roolin mukaisesti
require_once("funktiot.php");
Linkit($_GET[rooli]);
?>
```

Kuva 2 Esimerkki *Linkit* -funktion kutsumisesta

Jokaisella web-sivuston sivulla kutsutaan lataamisen alussa ensin kuvassa 3 olevaa *Linkit(\$rooli)*-funktioita, jossa *\$rooli*-muuttuja sisältää kirjautumisen yhteydessä tarkastetun käyttäjän roolin. Kuvaa 2 tarkasteltaessa ensin toiminnolla *require_once("funktiot.php")* otetaan käyttöön *funktiot.php* -tiedosto, josta löytyy *Linkit()*-funktio. Funktioon syötetään parametrina Internet-selaimen osoiteriviltä saatu tieto käyttäjän roolista muuttujalla *\$_GET[rooli]*. Tämä tarkastus tehdään jokaisen sivun alussa. Seuraavassa esitetään *Linkit()*-funktion toiminta.

```
function Linkit($rooli)
{
    $rooli_id = base64_decode($rooli);
    $pvm = date(" j.n.Y");

    //Vierailija, rooli 1
    if($rooli_id == 'vierailija')
    {
        //Vieraalle generoitaisiin tässä omat linkit
    }

    //Roolien mukaisia muita elseif -lohkoja on poistettu esimerkin
    //selkeyttämisen vuoksi.
```

```
//Isännöintitoimiston pääkäyttäjä 6
elseif($rooli_id == 'isannoitsija_paakayttaja')
{
    $rooli_id = base64_encode($rooli_id);
    echo "<span style='padding:5px;font-size:11px;'>";
    echo "<a href = 'etusivu.php?rooli=$rooli_id'>ETUSIVU </a>";
    echo "| ";
    echo "<a href = 'taloyhtiot.php?rooli=$rooli_id'>TALOYHTIÖT </a>";
    echo "| ";
    echo "<a href = 'tiedotteet.php?rooli=$rooli_id'>TIEDOTTEET </a>";
    echo "| ";
    echo "<a href = 'ota_yhteytta.php?rooli=$rooli_id'>OTA YHTEYTTÄ </a>";
    echo "| TÄNÄÄN ON $pvm";
    echo "</span>";
    echo "<br>";
    echo "<span style='padding:5px;font-size:11px;'>";
    echo "<a href = 'tiedostot.php?rooli=$rooli_id'>TIEDOSTOT </a>";
    echo "| ";
    echo "<a href = 'omat_tiedot.php?rooli=$rooli_id'>OMAT TIEDOT </a>";
    echo "| ";
    echo "<a href = 'yllapito.php?rooli=$rooli_id'>YLLÄPITO</a>";
    echo "</span>";
}
}
```

Kuva 3 Linkkivalikoiman muodostuminen, jos rooli on pääkäyttäjä

Roolin ollessa pääkäyttäjä, funktiossa suoritetaan vain viimeinen *elseif*-lohko, josta tulostetaan linkkivalikoima pääkäyttäjälle. Esimerkiksi ylläpitosivun linkki näkyy vain pääkäyttäjälle, koska selaimen osoiteriville on tallennettu muuttujaan *rooli* merkkijono *isannoitsija_paakayttaja*. Tämä merkkijono näytetään vielä koodattuna tietoturvan parantamiseksi. Sivuston tiedostorakenne voidaan nähdä seuraavasta taulukosta. Käyttäjän kirjautuessa web-sivustolle, käytetään vain niitä tiedostoja, joita käyttäjän roolilla voidaan näyttää.

Taulukko 1 Web-sivuston tiedostot ja niiden selitykset

Tiedoston nimi	Selitys
<i>etusivu.php</i>	Kirjautumisen jälkeinen sivu, jossa on yhteystietoja
<i>funktiot.php</i>	Web-sivuston toimintaan liittyvät funktiot
<i>hallitus_tiedostot.php</i>	Hallituksen asiakirjojen hallinnointi
<i>huoltoyhtio_tyot.php</i>	Huoltoyhtiön töiden hallinnointi
<i>index.php</i>	Aloitussivu, sisältää kirjautumisen ja rekisteröitymisen
<i>kirjautuminen.php</i>	Käyttäjän tunnistukseen liittyvät toiminnot
<i>muotoilu.css</i>	Tyylisivu, jossa on ulkoasuun liittyviä asetuksia
<i>omat_tiedot.php</i>	Omat tiedot
<i>ota_yhteytta.php</i>	Yhteydenottolomake
<i>taloyhtiot.php</i>	Kaikkien taloyhtiöiden tiedot
<i>tiedostot.php</i>	Tiedostojen hallinnointi
<i>tiedotteet.php</i>	Tiedotteet, joiden näkyvyyttä pääkäyttäjä voi hallinnoida
<i>tilaukset.php</i>	Tilaukset, joiden näkyvyyttä pääkäyttäjä voi hallinnoida
<i>toiminnot.php</i>	Sivuston keskeisten toimintojen suorittamispaikka
<i>tyyli.css</i>	Tyylisivu, jossa on ulkoasuun liittyviä asetuksia
<i>vikailmoitukset.php</i>	Sivu, jolla asiakkaat voivat jättää vikailmoituksia
<i>yllapito.php</i>	Sivu, jolla pääkäyttäjä voi hallinnoida käyttäjiä

Koska tietokanta on keskeinen osa web-sivuston toimintaa, siihen muodostetaan aina toiminnon vaatiessa yhteys ja se suljetaan toiminnon loputtua. Yhteydenottotapa on samanlainen toiminnosta riippumatta. Tarpeelliset ja usein käytetyt funktiot on sijoitettu keskitetysti yhteen tiedostoon, josta funktioita voidaan kutsua tarpeen vaatiessa. Web-sivuston keskeiset toiminnot on sijoitettu myös yhteen tiedostoon, jossa ne tarpeen vaatiessa suoritetaan. Seuraavassa on esimerkki tietokannan avaamiseen ja sulkemiseen sekä SQL-kyselyiden suorittamiseen liittyvistä funktioista.

```
function Avaa_kanta() //Ensimmäinen funktio
{
    $yht = mysql_connect("mysql3.nebula.fi","x","y");
    mysql_select_db("awioyfi");
}
function Lopeta_yhteys() //Toinen funktio
{
    mysql_close();
}
function Suorita_SQL($SQL) //Kolmas funktio
{
    return mysql_query($SQL);
}
```

Kuva 4 Tietokannan käsittelemiseen liittyviä funktioita

Ensimmäisessä funktiossa *\$yht* yhteysmuuttujaan tallennetaan PHP:n sisäisellä funktiolla *mysql_connect* MySQL-palvelimen sijainti *mysql.nebula.fi*, jossa *x* tarkoittaa käyttäjätunnusta ja *y* salasanaa. Sisäisellä funktiolla *mysql_select_db* valitaan käytettävä tietokanta. Toisessa funktiossa suljetaan tietokantayhteys sisäisellä funktiolla *mysql_close()*. Kolmannessa funktiossa *mysql_query(\$SQL)* palauttaa lähdekoodin suorituskohtaan tietokannasta kyselyn tuottaman tuloksen.

5.4 Testaus ja ohjeistus

Web-sivustoa on testattu kehityksen mukana siten, että esiintyneet virhetilanteet kirjattiin ylös ja korjattiin. Tutkintotyön tilaaja Amurin Willa Isännöinti Oy on myös testannut kehitettyjä uusia ominaisuuksia web-sivuston edistyessä ja raportoinut toimimattomuuksista sekä parannusehdotuksista. Raportointi on tapahtunut sähköpostitse ja uusien ominaisuuksien katselmointien yhteydessä suullisesti. Käyttöönoton yhteydessä pääkäyttäjää eli isännöitsijää opastetaan web-sivuston keskeisten toimintojen käyttämisessä. Muiden käyttäjien osalta ohjeistaminen tapahtuu sähköpostitse.

6 TUTKINTOTYÖN TULOKSET

Tässä luvussa on tarkoituksena kertoa kuvin ja esimerkein web-sivuston ulkoasusta ja toiminnasta. Tietokannan osalta esitellään taulujen rakennetta ja tietokannan taulujen relaatiot. Taulujen luontikäskyt ovat liitteenä.

6.1 Sivuston tietoturva

Tietoturva on huomioitu käyttäjän tunnistamisen ja roolin yhteydessä siten, että käyttäjän kirjautuessa tai rekisteröityessä sivustolle syötetty salasana lähetetään salatussa muodossa. Käyttäjän kirjautuessa sivustolle selaimen osoiterivillä kulkeva tieto roolista näytetään myös salatussa muodossa. Tietoturvan parantamiseksi kirjautumisen yhteydessä käynnistetään sessio, joka estää aiheettoman pääsyn sivustolle ja kirjaa käyttäjän automaattisesti ulos sivustolta, jos käyttäjä ei kirjautuneena tee sivustolla mitään.

Salasanan salaamiseen käytetään `AES_ENCRYPT($salattavatieto,"purkuavain")` ja purkamiseen `AES_DECRYPT($purettava,"purkuavain")` -funktioita. Merkkijono `"purkuavain"` on ohjelmoijan määrittelemä merkkijono, joka tarvitaan salauksen ja purkamisen yhteydessä. Nämä funktiot mahdollistavat tiedon salaamisen sekä purkamisen ja ne käyttävät virallista ja kehittyntä AES-salaamisstandardia (Advanced Encryption Standard). Salaukseen käytetään 128-bittistä avainta, jonka saa tarvittaessa muutettua 256-bittiseksi muokkaamalla funktion lähdekoodia. /11./

Käyttäjän roolia säilytetään selaimen osoiterivillä ja se muunnetaan kuvan 5 funktioilla ymmärrettävästä muodosta koodatuksi ja toisinpäin.

```
$rooli_salaamaton = base64_decode($rooli_salattu);  
$rooli_salattu = base64_encode($rooli_salaamaton);
```

Kuva 5 Roolin salaaminen /10./

Jos pääkäyttäjä on kirjautunut sivustolle, rooli näkyy koodattuna, josta on esimerkkinä `etusivu.php?rooli=aXNhbW5vaXRzaWphX3BhYWtheXR0Ywph`. Merkkijono `aXNhbW5vaXRzaWphX3BhYWtheXR0Ywph` on tässä tapauksessa

muuttujan *\$rooli_salattu* arvo, joka on *base64_encode(\$rooli_salaamaton)* - funktiolla asetettu. Muuttujan *\$rooli_salaamaton* arvo on merkkijono *isannoitsija_paakayttaja*. Muunnos toimii vastaavanlaisesti toiseen suuntaan syöttämällä dekoodaus-funktioon arvot käänteisessä järjestyksessä.

Seuraavassa kuvassa oleva funktio tarkistaa käyttäjän kirjautumisen jälkeen.

```
function Tarkista_kayttaja()  
{  
    session_start();  
    if(!isset($_SESSION["Kayttaja"]))  
    {  
        header("Location:index.php?info=2");  
    }  
    else{}  
}
```

Kuva 6 Käyttäjän tarkastaminen

Funktiota kutsutaan jokaisen sivun lataamisen alussa. Ensin kutsutaan PHP:n funktiota *session_start()*, joka luo yksilöllisen sessio-id:n. If-else -lohkossa tarkastetaan, että PHP:n globaaliin muuttujaan *\$_SESSION["Kayttaja"]* on asetettu kirjautumisen yhteydessä käyttäjän käyttäjätunnus. Jos tätä ei ole asetettu, tai sessio on vanhentunut, ohjataan käyttäjä aloitussivulle ja annetaan virheilmoitus. /10./

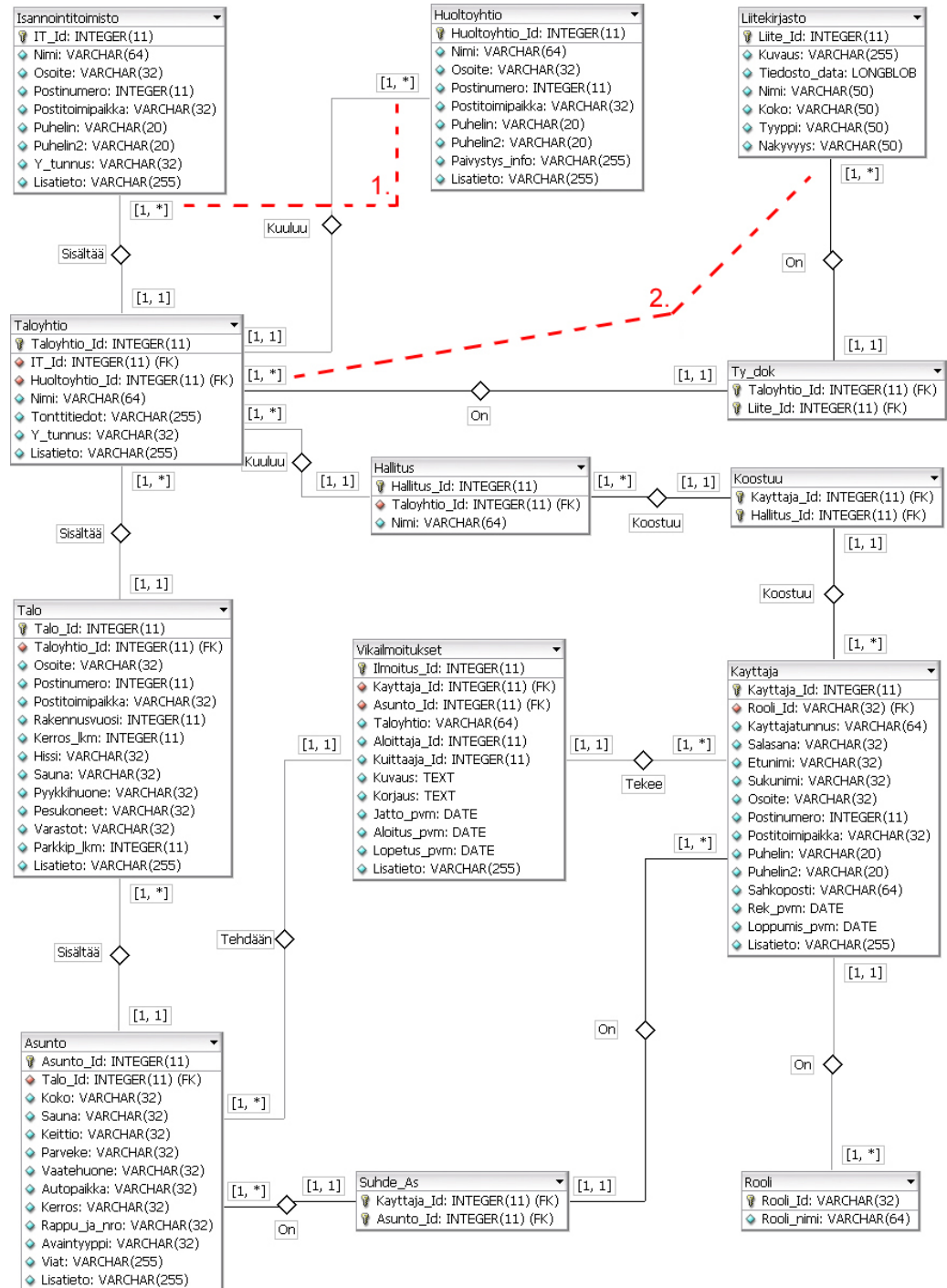
6.2 Tietokanta

Tietokantaa on pyritty kehityksen aikana muokkaamaan sivuston todellisten tarpeiden mukaiseksi ja toteutuksessa on pyritty huomioimaan ylläpidettävyys. Tauluja on yhteensä 13 kpl, jotka voidaan nähdä seuraavasta taulukosta.

Taulukko 2 Tietokannan taulut ja niiden selitykset

Taulun nimi	Selitys
Asunto	Sisältää tietoa taloon kuuluvista asunnoista
Hallitus	Taloyhtiön hallitus
Huoltoyhtio	Sisältää tietoa huoltoyhtiöistä, mm. yhteystiedot
Isannointitoimisto	Sisältää tietoa isännöintitoimistosta, mm. yhteystiedot
Kayttaja	Sisältää käyttäjäinformaation
Koostuu	Käytetään tiettyyn taloyhtiöön kuuluvan hallituksen yksilöimiseen
Liitekirjasto	Web-sivuston dokumenttien säilytyspaikka
Rooli	Käyttäjien roolit
Suhde_As	Käytetään tiettyyn asuntoon kuuluvan käyttäjän yksilöimiseen
Talo	Sisältää tietoa taloyhtiöön kuuluvasta talosta
Taloyhtio	Sisältää tietoa isännöintitoimistoon kuuluvista taloyhtiöistä
Ty_dok	Käytetään tiettyyn taloyhtiöön kuuluvan liitteen yksilöimiseen
Vikailmoitukset	Sisältää asukkaiden jättämiä vikailmoituksia asunnoista

Seuraavaksi esitettävä DBDesigner 4 -ohjelmistolla luotu kaavio helpottaa tietokannan relaatioiden ymmärtämistä.



Kuva 7 Tietokannan taulujen yhteydet

Kaaviota luettaessa on huomattava taulujen välissä olevat suhteet. Suhteet voivat olla erilaiset taulujen mukaisesti. Luettavuuden kannalta selvitän kuvasta kahden eri tyyppiä olevan suhteen toiminnan. Kohdat on numeroitu kuvaan.

1. Kun taulun päässä on merkintä $[1,n]$, se tarkoittaa, että taulu on suhteessa yhdestä moneen toisen taulun kanssa. Kohtaa luettaessa voidaan sanoa, että huoltoyhtiöllä voi olla usea taloyhtiö ja isännöintitoimistolla voi olla usea taloyhtiö. Tämän vuoksi huoltoyhtiön ja isännöintitoimiston id tulee aina taloyhtiön tietoihin.
2. Tässä kohdassa on kyseessä monesta-moneen -suhde kahden taulun välillä ja se ilmenee, kun kahden taulun välissä on molempien suhdetaulujen id:t yksilöivä taulu, tässä kohdassa taulu *Ty_dok*. Tämä tarkoittaa sitä, että yksi liite voi kuulua moneen taloyhtiöön ja yhteen taloyhtiöön voi kuulua monta liitettä. *Ty_dok* -tauluun on yksilöity jokaiseen taloyhtiöön kuuluvat liitteet.

Suuren tietomäärän tallentaminen tietokantaan on mahdollista monella tapaa. Tietoa voidaan tallettaa tietokantaan numeerisena, tekstimuodossa tai binääridatana. Tässä tutkintotyössä tiedostoja käsitellään tietokannassa binääridatana ja niiden hallinnointi tapahtuu yhden tietokannan taulun kautta. Tietokannassa binääridatan tallentamista varten taulussa tulee olla oma tietotyyppinsä. Tämä tietotyyppi on BLOB (Binary Large Objects) eli suuret binääriset objektit. BLOB-tietotyyppiä on kaikkiaan neljä kappaletta ja ne eroavat toisistaan ainoastaan säilytettävän datan määrässä. BLOB-tietotyypin avulla tietokannassa voidaan säilyttää eri tiedostomuotoja, koska data on binäärimuodossa. Tutkintotyön tietokannassa on käytetty longblob kokoa, jonka koko voidaan nähdä taulukosta 3. Tiedostojen hallinnointiin käytettävä taulu on kuvassa 8. /14./

Field	Type	Null	Key	Default	Extra
Liite_Id	int(11)		PRI	NULL	auto_increment
Kuvaus	varchar(255)	YES		NULL	
Tiedosto_data	longblob	YES		NULL	
Nimi	varchar(50)				
Koko	varchar(50)				
Tyyppi	varchar(50)				
Nakyvyys	varchar(50)	YES		NULL	

Kuva 8 Tietokannan Liitekirjasto-taulu

Taulukko 3 BLOB-tietotyypit /14./

Tyyppi	Koko/[tavu]
Longblob	4 294 967 295
Mediumblob	16 777 215
Blob	65 535
Tinyblob	255

Taulun kenttä *Tiedosto_data* on tiedoston vientikenttään sijoitetun tiedoston binääritieto. Tiedoston vieminen kantaan tapahtuu sql-kyselyllä, johon sijoitetaan selaimelta saadut arvot binääritiedosta, nimestä, koosta ja tyyppistä. Vietäessä selaimen kautta tiedostoja tietokantaan, selain pystyy päättelemään vietävän tiedoston päätteen perusteella tiedoston tyyppin.

```
$binaaridata = fread(fopen($binaaritiedosto, "r"), filesize($binaaritiedosto));
```

Kuva 9 Binääritiedon muodostaminen

Kuvassa 9 on esimerkki binääritiedon muodostumisesta. *\$binaaritiedosto* -muuttuja on selaimen tiedoston vientikenttään syötetty tieto. PHP:n metodilla *filesize()* määritellään syötetyn tiedoston koko tavuina ja metodi *fopen()* avaa tiedoston käsittelyä varten, parametri *r* tarkoittaa, että tiedosto on ainoastaan luettavissa. Metodilla *fread()* luetaan tavujen mukainen määrä tietoa muuttujaan *\$binaaridata*.

Vietäessä selaimen kautta tiedostoja tietokantaan, selain pystyy päättelemään vietävän tiedoston päätteen perusteella tiedoston tyyppin. Ladattaessa tiedostoja, selain päättelee tyyppin mukaan, mitä tiedostolle tehdään latausvaiheessa. Tiedoston tyyppejä ovat mm. application/msword, image/jpg ja text/plain. Selaimen kautta ladattaessa tiedostoja käytetään seuraavaa ohjelmakoodia, joka on kuvassa 10

```
$sql = "SELECT Tiedosto_data, Tyyppi, Nimi, ";  
$sql .= "Koko FROM Liitekirjasto WHERE Liite_Id=$id";  
$sqlkysely = suorita_sql($sql);  
$data = @mysql_result($sqlkysely, 0, "Tiedosto_data");  
$nimi = @mysql_result($sqlkysely, 0, "Nimi");  
$koko = @mysql_result($sqlkysely, 0, "Koko");  
$tyyppi = @mysql_result($sqlkysely, 0, "Tyyppi");  
header("Cache-Control: must-revalidate, post-check=0, pre-check=0");  
header("Content-type: $tyyppi");  
header("Content-length: $koko");  
header("Content-Disposition: attachment; filename=$nimi");  
header("Content-Description: PHP Generated Data");  
echo $data;
```

Kuva 10 Binääritiedon lataaminen

Alussa luodaan sql-kysely, jossa haetaan ladattavan tiedoston tiedot tietokannasta valitun id:n mukaisesti. Kyselyn tulokset pilkotaan muuttujiin, joita käytetään selaimen tiedoston lataamisessa. Ohjelmakoodin viimeinen rivi sisältää binääritiedon tiedostosta, jonka selain osaa assosoida oikeaan ohjelmaan tyypin mukaisesti.

6.3 Sivusto

Sivuston rakennetta on alettu muokata valmiin pohjan avulla. /5./ Yläpalkki, sen alla oleva navigointipalkki, vasen navigointipalkki, sisältöosuus ja alapalkki ovat kaikki omia elementtejään, joiden ominaisuudet on määritelty CSS-tiedostossa. Elementti määritetään sivulle *div*-tagin avulla ja elementti saa yksilöllisen nimen. Sivun lataamisen alussa on ohjelmakoodi, joka hakee jokaiselle elementille määritellyt ominaisuudet. Koodi on kuvassa 11.

```
<link rel="stylesheet" type="text/css" href="tyyli.css" />  
<link rel="stylesheet" type="text/css" href="muotoilu.css" />
```

Kuva 11 Tyylitiedostojen lataaminen sivustolle /3, 4./

Tyyli.css sisältää sivuston rakennemäärittelyitä ja *muotoilu.css* sisältää tekstin ulkoasuun liittyviä muotoilumäärittelyitä. Kuvassa 12 on esimerkki *tyyli.css*-tiedostosta, jossa määritellään vasen sarake ja yläpalkki.

```
#l-col {  
  float:left;  
  background:#94C8F4;  
  width:150px;}  
  
#hdr {  
  height:80px;  
  background:#94C8F4;}
```

Kuva 12 Esimerkki *tyyli.css* tiedostosta /3, 4./

#l-col ja *#hdr* ovat sivuston lähdekoodiin määriteltyjen elementtien yksilölliset tunnisteet, joiden avulla tyylitiedostosta tiedetään ladata oikeat ominaisuudet elementille. *float*-komento asettaa sarakkeen sijainniksi vasen ja *width*-komento kertoo sarakkeen leveyden. *background*-komennolla voidaan määrittää elementin taustaväri ja *height*-komento asettaa navigointipalkin korkeuden. Kuvassa 13 on sivuston yläpalkkielementin lähdekoodi.

```
<div id="hdr" align="center">  
  
</div>
```

Kuva 13 Yläpalkkielementti

Ensimmäisellä rivillä elementillä on yksilöllinen id *hdr*, jonka avulla ominaisuuksien lataaminen tyylitiedostosta onnistuu kyseiselle elementille. Elementissä oleva sisältö on sijoitettu keskelle *align="center"*-komennolla. *img src*-komennolla ladataan yläpalkkiin kuva *awioy.jpg*.

Web-sivustolle saavuttaessa näkymä on kuvan 14 mukainen.

Amurin
Willa - Isännöinti Oy

TÄNÄÄN ON 6.3.2005

Sisäänkirjautuminen

1. Käyttäjätunnus:
Salasana:

Kirjaudu vieraana jättämällä käyttäjätunnus ja salasana tyhjiksi.

2. Kirjaudu

3. Rekisteröidy

Amurin Willa Isännöinti Oy

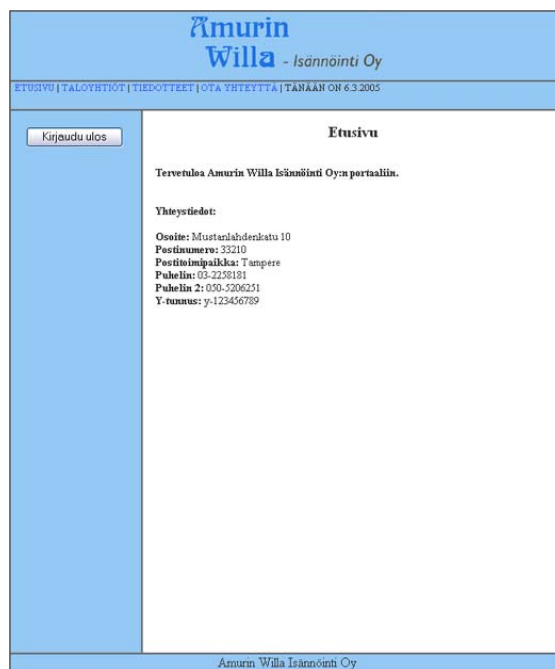
Kuva 14 Aloitussivu

1. Kohdassa voi syöttää käyttäjätunnuksen ja salasanan. Jos käyttäjätunnusta ei ole olemassa tai salasana syötetään väärin, annetaan virheilmoitus. Etusivu avautuu tunnuksen ollessa oikea.
2. Jos painiketta painetaan ilman, että on syötetty käyttäjätunnusta ja salasanaa, voidaan kirjautua vieraana sivustolle, jolloin näytetään vain rajoittunut osa web-sivustoa.

3. Jos valitaan rekisteröinti, linkki ohjaa tietojen täyttämiseen tarkoitettuun osaan. Tässä kohdassa täytetään tarpeelliset yhteystiedot ja kirjoitetaan haluttu käyttäjätunnus ja salasana. Jos tunnus on jo olemassa tai syötetyt salasanat eivät täsmää, annetaan virheilmoitus.

6.3.1 Vieras

Kun on kirjaututtu web-sivustolle ilman käyttäjätunnusta ja salasanaa, näytetään ensin kuvan 15 mukainen sivu, jossa on isännöintitoimiston yhteystiedot. Jokaisella sivulla on vasemmassa navigointipalkissa uloskirjautumispainike.



Kuva 15 Etusivu



The screenshot shows the 'Taloyhtiön tietoja' (Apartment Information) page. The header includes the Amurin Willa logo and navigation links. The main content area is divided into two columns. The left column contains a 'Kirjeudu ulos' button and a list of apartment buildings: Peräniemen taloitus, Tammonkadun taloitus, Järvenkadun taloitus, and Pedronkadun taloitus. Below this is a section for 'Taloyhtiön talot' (Apartment Buildings) listing Peräniemi 79, Peräniemi 76, and Peräniemi 80. The right column features a 'Taloyhtiön tietoja' section with a heading and a paragraph: 'Voit selata tietoja valitsemalla ensin taloyhtiön, jonka jälkeen ilmestyvästi talolistasta voit valita kyseisen taloyhtiön talon.' This is followed by detailed information for Peräniemi 79, including postal code 33900, location Tampere, construction year 1990, 4 floors, lift, sauna, and parking spaces.

Kuva 16 Taloyhtiöiden tietoja


Valittaessa navigointipalkista taloyhtiö -linkki, näytetään kuvan 16 mukainen sivu. Sivulla on tietoja isännöintitoimiston kaikista taloyhtiöistä. Tietoja selattaessa ensin valitaan vasemmasta navigointipalkista taloyhtiö, jonka jälkeen taloyhtiön talot listautuvat alle. Valitun talon tiedot listautuvat keskelle.



The screenshot shows the 'Tiedotteet' (Announcements) page. The header is identical to the previous screenshot. The main content area has a 'Kirjeudu ulos' button and a heading 'Tiedotteet'. Below the heading is a paragraph: 'Alla olevasta listasta voit valita haluamasi tiedotteen. Tiedotetta klikkaamalla avautuu ikkuna, josta voit valita avauksen tai tallennuksen.' A single bullet point links to 'Linkkivalkoima.doc'.

Kuva 17 Tiedotteet

Valitsemalla tiedotteet -linkin, näytetään kuvan 17 mukainen sivu. Sivulla on pääkäyttäjän julkaisemia tiedotteita, joista valitsemalla avautuu lataamisikkuna. Tästä ikkunasta voidaan valita tiedoston avaaminen tai tallentaminen haluttuun paikkaan tietokoneen kovalevyille.



The screenshot shows a web browser window displaying the 'Ota yhteyttä' (Contact Us) page of Amurin Willa - Isännöinti Oy. The page has a blue header with the company logo and navigation links: ETUSIVU | TALOVHITOT | TIEDOTTEET | OTA YHTEYTTÄ | TÄNÄÄN ON 6.3.2005. A 'Kirjautuu ulos' button is visible in the top left. The main content area is titled 'Ota yhteyttä' and contains the text: 'Voit ottaa yhteyttä Amurin Willa Isännöintiin alla olevalla lomakkeella.' Below this are input fields for 'Nimi:' (Name), 'Sähköposti:' (Email), and 'Palautus:' (Message). There is a radio button for 'Halustako vastauksen?' (Do you want a response?) with 'Kyllä' (Yes) selected and 'En' (No) as an option. At the bottom of the form are two buttons: 'Lähetä palaute' (Send message) and 'Tyhjennä' (Clear). The footer of the page reads 'Amurin Willa Isännöinti Oy'.

Kuva 18 Ota yhteyttä

Kun halutaan ottaa yhteyttä isännöintitoimistoon tai tässä tapauksessa isännöitsijään, pitää täyttää kuvan 18 mukaiset tiedot. Nimen, sähköpostin ja palautteen lisäksi tulee valita vastauksen pyytäminen. Palaute lähetetään painamalla Lähetä palaute -painiketta. Tyhjän palautteen lähettäminen ei ole mahdollista, vaan siitä annetaan virheilmoitus. Palaute ohjataan ohjelmakoodiin lisättyyn sähköpostiin, josta isännöitsijä voi halutessaan vastata palautteisiin.

6.3.2 Asukas ja Osakas

Käyttäjän roolin ollessa asukas tai osakas, navigointipalkista on mahdollisuus valita vikailmoituksen jättäminen tai omien tietojen päivittäminen.

Vikailmoituksen jättämislomake on kuvassa 19 ja omien tietojen päivittäminen on kuvassa 20.

The screenshot shows the 'Vikailmoitus' (Complaint) form in the Amurin Willa system. The page header includes the logo 'Amurin Willa - Isännöinti Oy' and navigation links: 'ETUSIVU | TALOYHTIÖT | TIEDOTTEET | OTA YHTEYTTÄ | TÄNÄÄN ON 6.3.2005'. Below the header, there are links for 'VIKAILMOITUS' and 'OMAT TIEDOT'. A 'Kirjeudu ulos' button is located in the top left corner. The main content area is titled 'Vikailmoitus' and contains the following text: 'Vikailmoitukseen lisätään automaattisesti taloyhtiön nimi ja yhteystietosi huoltoyhtiön yhteydenottoa varten.' and 'Jos taloyhtiössä ei ole huoltoyhtiötä, soita vika-asioissa ensin isännöitsijälle numeroon 03-2258181 tai 050-5206251'. Below this is a text input field with the label 'Kirjoita vikailmoitus alla olevaan tekstikenttään.' and two buttons: 'Jätä vikailmoitus' and 'Tyhjennä'. The footer of the page reads 'Amurin Willa Isännöinti Oy'.

Kuva 19 Vikailmoituksen jättäminen

Vikailmoitus kirjoitetaan sille varattuun tekstikenttään ja se lähetään painamalla kentän alapuolella olevaa Jätä vikailmoitus -painiketta. Vikailmoitukseen lisätään automaattisesti ilmoittajan yhteystiedot huoltoyhtiön yhteydenottoa varten.

The screenshot shows the 'Omat tiedot' (My information) form in the Amurin Willa system. The page header is identical to the previous screenshot. Below the header, there are links for 'VIKAILMOITUS' and 'OMAT TIEDOT'. A 'Kirjeudu ulos' button is located in the top left corner. The main content area is titled 'Omat tiedot' and contains the text: 'Voit muuttaa omia tietojasi muokkaamalla seuraavia tietoja:'. Below this is a list of input fields for personal information: Etunimi (Testi), Sukunimi (Teppo), Osoite (Testikatu 3), Postinumero (33900), Postitoimipaikka (Testila), Puhelin (0400-123456), Puhelin 2 (03-1234567), Sähköposti (testi@testi.com), and Salasan (testi2). A 'Päivitä tiedot' button is located at the bottom of the form. The footer of the page reads 'Amurin Willa Isännöinti Oy'.

Kuva 20 Omien tietojen päivittäminen

Omia tietoja voi päivittää muuttamalla yhteystietoja ja salasanaansa voi muuttaa haluamukseen. Tietojen päivitys tapahtuu painamalla tekstikenttien alapuolella olevaa Päivitä tiedot -painiketta.

6.3.3 Välitystoimiston käyttäjä

Välitystoimiston käyttäjällä on mahdollisuus muokata omia tietojään ja tilata esitteitä valitsemastaan taloyhtiöstä. Esitteiden tilaamiseen käytettävä sivu on kuvassa 21.

Ämurin
Willa - Isännöinti Oy

ETUSIVU | TALOYHTIÖT | TIEDOTTEET | OTA YHTEYTTÄ | TÄNÄÄN ON 6.3.2005
TILAUKSET | OMAT TIEDOT

Kirjeudu ulos

Valitse toiminto

Tilaa esite
Selaa esitteitä

Tilaukset

Valitse taloyhtiön esite:
Järmonkatie 72

Lisätieto:

Pyydä esite Tyhjennä

Ämurin Willa Isännöinti Oy

Kuva 21 Esitteiden tilaaminen

Esitettä tilattaessa ensin valitaan ylhäällä olevasta alasvetovalikosta isännöintitoimistoon kuuluva taloyhtiö ja tarvittaessa kirjoitetaan tilauksen yhteyteen lisätietoja. Tilaus lähetetään painamalla lisätietokentän alla olevaa Pyydä esite -painiketta. Esitteitä selattaessa keskellä on lista tilatuista esitteistä, jotka pääkäyttäjä on julkaissut. Esitteiden lukumäärän ollessa enemmän kuin kymmenen, esitteet sivutetaan siten, että näytetään vain 10 esitettä kerralla. Valitsemalla halutun esitteen avautuu tiedostojen lataamisikkuna, josta voi valita tiedoston avaamisen tai tallentamisen tietokoneelle.

6.3.4 Huoltoyhtiön käyttäjä

Huoltoyhtiön käyttäjällä on mahdollisuus avata vikailmoitukseen liittyvä työ ja sulkea se. Työt listautuvat vasempaan navigointipalkkiin siten, että ylempänä ovat aloittamattomat työt ja alle listautuvat aloitetut työt päivämäärittäin. Aloittamattomien ja aloitettujen töiden päivämäärien perään listautuu kyseisenä päivämääränä jätettyjen vikailmoitusten lukumäärä.

Työtä aloitettaessa työlle on lisättävä aloituspäivämäärä ja jos päivämäärä jätetään antamatta, siitä annetaan virheilmoitus. Työtä lopetettaessa työlle on kirjoitettava pakollinen korjausinformaatio sekä mahdollinen lisätieto ja työn lopetuspäivämäärä. Jos edellä mainittuja tietoja ei anneta tai ne täytetään puutteellisesti työn lopetuksen yhteydessä, annetaan virheilmoitus. Isännöitsijälle lähetetään tietoa sähköpostitse työn aloittamisesta ja lopetuksesta. Huoltoyhtiön käyttäjällä on myös mahdollisuus päivittää omia tietoja. Huoltoyhtiön aloitettu työ näkyy kuvassa 22.

The screenshot shows a web interface for 'Amurin Willa - Isännöinti Oy'. The main content area is titled 'Vikailmoitukset' (Incident Reports). On the left, there is a sidebar with a 'Kirjaudu ulos' (Logout) button and a list of work items under 'Aloittamattomat työt' (Unstarted work) and 'Aloitettut työt' (Started work). The main form contains the following fields:

- Nimi:** Testi Teppo
- Yhteystiedot:** Testitatu 3, 33900 Testilä, 0400-123456, testi@testi.com
- Taloyhtiö:** Perhekeskuden talohtiö
- Vian kuvaus:** Vihaa asunnossa
- Vikailmoituksen jättämispvm (vvv-kk-pp):** 2005-03-06
- Korjauksen aloituspvm:** 2005-03-06
- Korjauksen kuvaus:** (Empty text field)
- Lisätieto:** (Empty text field)
- Korjauksen lopetuspvm:** (Dropdown menu for month, dropdown menu for day, and a 'Lopeta työ' button)

At the bottom of the page, the text 'Amurin Willa Isännöinti Oy' is visible.

Kuva 22 Huoltoyhtiön työnäkymä

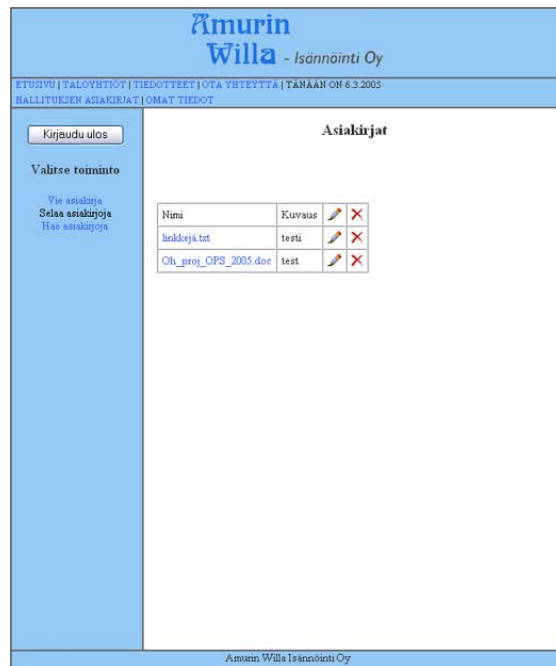
6.3.5 Hallituksen jäsen

Taloyhtiöiden hallitusten jäsenillä on mahdollisuus taloyhtiökohtaisesti lukea ja selata oman hallituksensa asiakirjoja. Käyttäjät voivat viedä tietokantaan uusia dokumentteja, jotka näkyvät vain oman hallituksen jäsenille. Hallituksen jäsenet voivat myös päivittää omia tietojaan. Asiakirjojen vienti tietokantaan tapahtuu seuraavan sivun avulla, joka näkyy kuvasta 23.



Kuva 23 Asiakirjojen vieminen

Jokaiselle asiakirjalle on annettava kuvausteksti, joka on mahdollisimman tarkka kuvaus vietävästä asiakirjasta. Asiakirja valitaan painamalla Selaa -painiketta, joka näyttää oman tietokoneen tiedostorakenteen. Hakemalla tiedoston sen sijaintipaikasta, tiedoston sijaintipolku ilmestyy kuvauskentän alapuolelle. Hallituksen nimi haetaan automaattisesti tietokannasta. Asiakirjan voi viedä tietokantaan painamalla Lisää asiakirja -painiketta. Kun asiakirja on viety tietokantaan, viedystä asiakirjasta näytetään vielä sen nimi, koko ja tyyppi. Jos asiakirjan viennin yhteydessä ei ole täytetty kaikkia kenttiä, annetaan virheilmoitus.



Kuva 24 Asiakirjojen selaaminen

Taloyhtiön hallituskohtaisia asiakirjoja voi selata valitsemalla vasemmasta navigointipalkista Selaa asiakirjoja -linkin, joka on kuvassa 24. Asiakirjat listautuvat keskelle taulukkoon. Jos asiakirjoja on enemmän kuin kymmenen kappaletta, asiakirjat sivutetaan siten, että näytetään kymmenen asiakirjaa kerrallaan. Valitsemalla asiakirjan taulukosta, avautuu tiedoston lataamisikkuna, josta voi valita tiedoston avaamisen tai tallentamisen tietokoneelle.

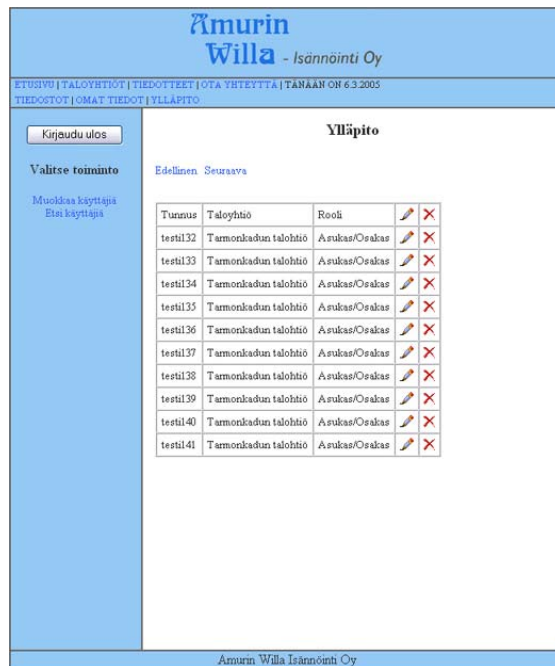
Asiakirjan voi poistaa punaisesta ruksista ja poistettaessa asiakirjaa kysytään aina varmistus asiakirjan poistosta. Jokaisen asiakirjan kuvausta voi muokata painamalla kynän kuvaa asiakirjan kohdalla. Tietyn asiakirjan etsiminen tapahtuu valitsemalla vasemmasta navigointipalkista Hae asiakirja -linkki, joka näyttää tekstikentän, johon voi kirjoittaa etsittävän asiakirjan nimen. Hakutulokset näytetään taulukoituna vastaavanlaisesti kuin selatessa asiakirjoja. Hallituksen jäsenellä on myös mahdollisuus muokata ja päivittää omia tietojaan.

6.3.6 Pääkäyttäjä

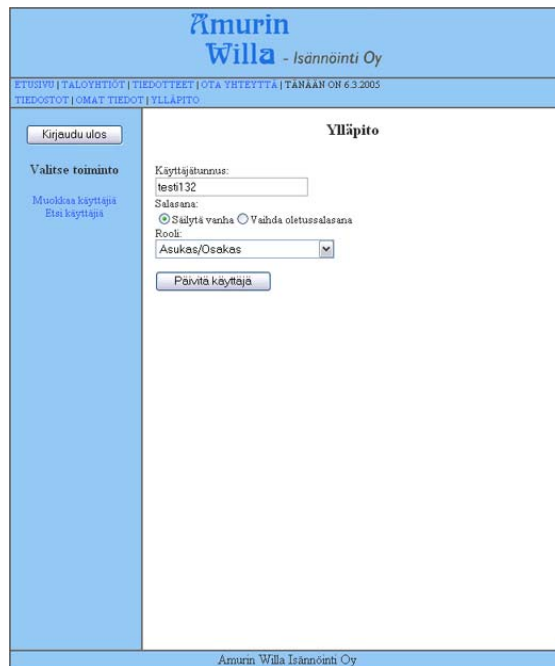
Web-sivuston pääkäyttäjällä on mahdollisuus julkaista dokumentteja ja määrittää dokumenttikohtainen näkyvyys. Näkyvyys tarkoittaa tässä tiedoston julkaisemista tiedotteet -sivulle tai välitystoimiston tilaukset -sivulle. Tiedoston vieminen tietokantaan ja niiden selaaminen sekä hakeminen tietokannasta tapahtuvat vastaavanlaisesti kuin hallituksen jäsenillä. Tiedoston muokkauksessa on ainoastaan eroa, sillä muokatessa pystyy muuttamaan tiedoston näkyvyyttä.

Pääkäyttäjä voi selata ja hakea web-sivuston käyttäjiä. Selatessa käyttäjiä, käyttäjät listataan taulukkoon, josta ilmenee mahdollinen käyttäjän taloyhtiö ja rooli. Käyttäjän voi poistaa punaisesta ruksista ja poiston yhteydessä kysytään varmistus ennen poistoa. Käyttäjiä voi myös hakea tietokannasta valitsemalla vasemmasta navigointipalkista Etsi käyttäjiä -linkki, josta avautuu tekstikenttä haettavaa käyttäjätunnusta varten. Hakutulokset listautuvat keskelle vastaavanlaisesti, kuin selatessa tiedostoja. Pääkäyttäjä voi myös poistaa taloyhtiöitä ja niihin kuuluvat talot pois tietokannasta ja hän voi myös päivittää omia tietojaan.

Pääkäyttäjällä on oikeus muuttaa jokaisen käyttäjän roolia. Rekisteröityneen käyttäjän rooli on oletuksena vierailija, joten rekisteröitymisen yhteydessä lähetetystä sähköpostista pääkäyttäjälle on tarkastettava rekisteröityneen käyttäjän tiedot mahdollisen roolin muutoksen vuoksi. Pääkäyttäjä voi myös muuttaa käyttäjien salasanan oletussalasanaksi mahdollisen unohdustilanteen sattuessa. Jos pääkäyttäjä vaihtaa oletussalasanan käyttäjälle, tieto salasanan muutoksesta lähetetään käyttäjälle sähköpostitse. Kuvat käyttäjien selaamisesta ja muokkaamisesta on kuvissa 25 ja 26.



Kuva 25 Käyttäjien selaaminen



Kuva 21 Käyttäjän muokkaaminen

6.4 Alkuasetukset ja ylläpito

Web-sivuston käyttöönoton yhteydessä tietokantaan syötetään isännöinti-toimiston piiriin kuuluvat tiedot tekstitiedoston avulla. Näitä tietoja ovat taloyhtiöiden, talojen, asuntojen ja käyttäjien (tässä tapauksessa asukkaiden) tiedot, mitkä ovat ennen olleet vanhassa järjestelmässä. Tässä järjestelmässä on salattu Microsoft Access -tietokanta, jonka avaaminen onnistuu ainoastaan vanhalla Access 95 -tietokantaohjelmistolla. Vanhasta tietokannasta käydään rivi riviltä tietueet läpi ja kirjataan saadut tiedot tekstitiedostoon.

Tietojen siirtämistä harkittiin myös tietojen siirtoon tarkoitetun ohjelman tekemisellä, mutta idea hylättiin, koska uusi ja vanha tietokanta eroavat suuresti toisistaan ja ohjelman tekemiseen olisi kulunut liian paljon aikaa. Tarvittava tiedot sisältävä tekstitiedosto ajetaan tietokantaan SSH Secure Shell -ohjelmiston terminaaali-ikkunan avulla ja tekstitiedostoon voidaan samalla määrittää myös pääkäyttäjän käyttäjätunnus.

Sivuston ylläpidossa tarvitaan hieman asiantuntemusta, vaikka sivustolla tarvittava ylläpito on pyritty toteuttamaan ja suunnittelemaan vähäiseksi. Tarvittavan ylläpidon hoitaa sivuston suunnittelija. Ylläpitoa helpottamaan web-sivuston yhteyteen asennetaan ilmainen tietokannanhallintaohjelmisto phpMyAdmin, joka on vapaan lähdekoodin ohjelmisto.

7 POHDINTA

Tutkintotyö on valmistunut suunnitellussa aikataulussa, vaikka työn alussa käytettyjen kehitysympäristöjen tunteminen ei ollut vahvalla tasolla. PHP:n perusasioiden ja MySQL:n käsittelemiseen liittyvien toimintojen opettelu vaati työn alkuvaiheessa paljon aikaa ja kärsivällisyyttä. Myös sivuston rakenteen toteutuksessa ja suunnittelussa on käytetty aikaa, koska tyyli tiedostojen käyttö oli kokonaan uusi asia. Tyyli tiedostoja käytettäessä huomioitiin myös sivuston visuaalinen muokattavuus tulevaisuuden kannalta ajateltuna.

Web-sivusto otetaan käyttöön kesällä 2005 osoitteessa <http://www.awioy.fi>. Sivuston kehittäminen on ollut vaivatonta valmiin Internet-osoitteen ansiosta, koska kehitys on tapahtunut kokonaisuudessa varsinaisen osoitteen alikansiossa <http://www.awioy.fi/kehitysversio>, josta on ollut helppoa testata sivustolle kehitettyjä ominaisuuksia. Työn kustannuksia ovat olleet ainoastaan Internet-osoitteesta menevä kuukausimaksu palveluntarjoajalle, jolloin on säästyty palvelimen ja sen ohjelmistojen hankkimiseen liittyviltä kustannuksilta.

Kehityksen yhteydessä työn tilaajalta saadun palautteen mukaan web-sivustosta on onnistuttu luomaan selkeä ja helppolukuinen kokonaisuus. Työn tilaajan vaatimukseen on ollut helppoa mukautua ja vaatimuksia on pystytty työn aikana muokkaamaan järkeviksi. Kokonaisuutena ajatellen tutkintotyön toteuttaminen on onnistunut lähes moitteetta ja vaikka kehitystyökalut ja ympäristöt ovat olleet ilmaisia vapaan lähdekoodin tuotteita, ei kehityksessä ja toteutuksessa ole ilmaantunut ylitsepääsemättömiä ongelmia. Kehitetyn web-sivuston jatkokehitykseen on jo olemassa uusia suunnitelmia, joiden suunnittelu ja toteutus jäävät myöhempään ajankohtaan. Loppukommentiksi voin sanoa, että tutkintotyön tekeminen oli haastavaa ja mielenkiintoista uusien asioiden kannalta ajateltuna ja on oltava erittäin tyytyväinen saatuun lopputulokseen.

LÄHTEET

1. Heinisuo Rami, PHP ja MySQL tietokantapohjaiset verkkopalvelut, Satku – Kauppakaari, 2001, sivumäärä 212
2. Haikala Ilkka, Märijärvi Jukka, Ohjelmistotuotanto, Satku – Kauppakaari, 2001, sivumäärä 386
3. Laine Niklas, Karjalainen Erkki, WWW-sivut HTML-kielellä REALbasic ohjelmointi, Helsinki Media – Hakapaino Oy, 2001, sivumäärä 116
4. Tizag Tutorials, [www-sivu], Saatavissa: <http://www.tizag.com/>
5. The SSI Developer, [www-sivu], Saatavissa: <http://www.ssi-developer.net/>
6. ITK215 WWW-sovellukset – kevät 2004, [www-sivu], Saatavissa: <http://www.cs.jyu.fi/~kolli/ITK215/luennot.html>
7. Maguma's modular PHP IDE/Editor, [www-sivu], Saatavissa: <http://www.maguma.com/>
8. SSH Communication Security, [www-sivu], Saatavissa: <http://www.ssh.com>
9. fabFORCE.net, [www-sivu], Saatavissa: <http://www.fabforce.net/dbdesigner4>
10. PHP: Hypertext Preprocessor, [www-sivu], Saatavissa: <http://www.php.net/>
11. MySQL: The World's Most Popular Open Source Database, [www-sivu], Saatavissa: <http://www.mysql.com/>
12. Opera Web Browser, [www-sivu], Saatavissa: <http://www.opera.com/>
13. Firefox – Rediscover the web, [www-sivu], Saatavissa: <http://www.mozilla.org/produ>
14. Blobbing data with PHP and MySQL, [www-sivu], Saatavissa: <http://www.devarticles.com/c/a/MySQL/Blobbing-Data-With-PHP-and-MySQLcts/firefox/>

***** Isännöintitoimiston tietoja varten oleva taulu *****

```
CREATE TABLE Isannointitoimisto(  
IT_Id                INTEGER NOT NULL AUTO_INCREMENT,  
Nimi                 VARCHAR(64),  
Osoite               VARCHAR(32),  
Postinumero          INTEGER,  
Postitoimipaikka    VARCHAR(32),  
Puhelin              VARCHAR(20),  
Puhelin2             VARCHAR(20),  
Y_tunnus             VARCHAR(32),  
Lisatieto            VARCHAR(255),  
PRIMARY KEY (IT_Id)  
);
```

***** Rooli -taulu *****

```
CREATE TABLE Rooli(  
Rooli_Id             VARCHAR(32) NOT NULL,  
Rooli_nimi           VARCHAR(64),  
PRIMARY KEY (Rooli_Id)  
);
```

***** Käyttäjä -taulu *****

```
CREATE TABLE Kayttaja(  
Kayttaja_Id         INTEGER NOT NULL AUTO_INCREMENT,  
Rooli_Id             VARCHAR(32) NOT NULL,  
Kayttajatunnus      VARCHAR(64),  
Salasana             VARCHAR(32),  
Etunimi              VARCHAR(32),  
Sukunimi             VARCHAR(32),  
Osoite               VARCHAR(32),  
Postinumero          INTEGER,  
Postitoimipaikka    VARCHAR(32),  
Puhelin              VARCHAR(20),  
Puhelin2             VARCHAR(20),  
Sahkoposti           VARCHAR(64),  
Rek_pvm              DATE NOT NULL,  
Loppumis_pvm         DATE,  
Lisatieto            VARCHAR(255),  
PRIMARY KEY (Kayttaja_Id),  
FOREIGN KEY (Rooli_Id)  
REFERENCES Rooli (Rooli_Id)  
);
```

***** Huoltoyhtiö -taulu *****

```
CREATE TABLE Huoltoyhtio(  
Huoltoyhtio_Id      INTEGER NOT NULL AUTO_INCREMENT,  
Nimi                 VARCHAR(64),  
Osoite               VARCHAR(32),  
Postinumero          INTEGER,  
Postitoimipaikka    VARCHAR(32),  
Puhelin              VARCHAR(20),  
Puhelin2             VARCHAR(20),  
Paivystys_info       VARCHAR(255),  
Lisatieto            VARCHAR(255),  
PRIMARY KEY (Huoltoyhtio_id)  
);
```

***** Taloyhtiö -taulu *****

```
CREATE TABLE Taloyhtio(  
Taloyhtio_Id        INTEGER NOT NULL AUTO_INCREMENT,  
IT_Id                INTEGER NOT NULL,  
Huoltoyhtio_Id      INTEGER NOT NULL,  
Nimi                 VARCHAR(64),  
Tonttitiedot         VARCHAR(255),  
Y_tunnus             VARCHAR(32),  
Lisatieto            VARCHAR(255),  
PRIMARY KEY (Taloyhtio_Id),  
FOREIGN KEY (IT_Id)  
REFERENCES Isannointitoimisto (IT_Id),  
FOREIGN KEY (Huoltoyhtio_Id)  
REFERENCES Huoltoyhtio (Huoltoyhtio_Id)  
);
```

***** Talo -taulu *****

```
CREATE TABLE Talo(  
Talo_Id INTEGER NOT NULL AUTO_INCREMENT,  
Taloyhtio_Id INTEGER NOT NULL,  
Osoite VARCHAR(32),  
Postinumero INTEGER,  
Postitoimipaikka VARCHAR(32),  
Rakennusvuosi INTEGER,  
Kerros_lkm INTEGER,  
Hissi VARCHAR(32),  
Sauna VARCHAR(32),  
Pyykkihuone VARCHAR(32),  
Pesukoneet VARCHAR(32),  
Varastot VARCHAR(32),  
Parkkip_lkm INTEGER,  
Lisatieto VARCHAR(255),  
PRIMARY KEY (Talo_Id),  
FOREIGN KEY (Taloyhtio_Id)  
REFERENCES Taloyhtio (Taloyhtio_Id)  
);
```

***** Asunto -taulu *****

```
CREATE TABLE Asunto(  
Asunto_Id INTEGER NOT NULL AUTO_INCREMENT,  
Talo_Id INTEGER NOT NULL,  
Koko VARCHAR(32),  
Sauna VARCHAR(32),  
Keittio VARCHAR(32),  
Parveke VARCHAR(32),  
Vaatehuone VARCHAR(32),  
Autopaikka VARCHAR(32),  
Kerros VARCHAR(32),  
Rappu_ja_nro VARCHAR(32),  
Avaintyyppi VARCHAR(32),  
Viat VARCHAR(255),  
Lisatieto VARCHAR(255),  
PRIMARY KEY (Asunto_Id),  
FOREIGN KEY (Talo_Id)  
REFERENCES Talo (Talo_Id)  
);
```

***** Suhde_As -taulu *****

```
CREATE TABLE Suhde_As(  
Kayttaja_Id INTEGER NOT NULL,  
Asunto_Id INTEGER NOT NULL,  
PRIMARY KEY (Kayttaja_Id, Asunto_Id),  
FOREIGN KEY (Kayttaja_Id)  
REFERENCES Kayttaja (Kayttaja_Id),  
FOREIGN KEY (Asunto_Id)  
REFERENCES Asunto (Asunto_Id)  
);
```

***** Hallitus -taulu *****

```
CREATE TABLE Hallitus(  
Hallitus_Id INTEGER NOT NULL,  
Taloyhtio_Id INTEGER NOT NULL,  
Nimi VARCHAR(64) NOT NULL,  
PRIMARY KEY (Hallitus_Id),  
FOREIGN KEY (Taloyhtio_Id)  
REFERENCES Taloyhtio (Taloyhtio_Id)  
);
```

***** Koostuu -taulu *****

```
CREATE TABLE Koostuu(  
Kayttaja_Id INTEGER NOT NULL,  
Hallitus_Id INTEGER NOT NULL,  
PRIMARY KEY (Kayttaja_Id, Hallitus_Id),  
FOREIGN KEY (Kayttaja_Id)  
REFERENCES Kayttaja (Kayttaja_Id),  
FOREIGN KEY (Hallitus_Id)  
REFERENCES Hallitus (Hallitus_Id)  
);
```

***** Liitekirjasto -taulu *****

```
CREATE TABLE Liitekirjasto(  
  Liite_Id          INTEGER NOT NULL AUTO_INCREMENT,  
  Kuvaus           VARCHAR(255),  
  Tiedosto_data    LONGBLOB,  
  Nimi             VARCHAR(50) NOT NULL,  
  Koko             VARCHAR(50) NOT NULL,  
  Tyyppi           VARCHAR(50) NOT NULL,  
  
  Nakyyvyys       VARCHAR(50),  
  PRIMARY KEY (Liite_Id),  
  UNIQUE ID(Liite_Id)  
);
```

***** Ty_dok -taulu *****

```
CREATE TABLE Ty_dok(  
  Taloyhtio_Id     INTEGER NOT NULL,  
  Liite_Id         INTEGER NOT NULL,  
  PRIMARY KEY (Taloyhtio_Id, Liite_Id),  
  FOREIGN KEY (Taloyhtio_Id)  
  REFERENCES Taloyhtio (Taloyhtio_Id),  
  FOREIGN KEY (Liite_Id)  
  REFERENCES Liitekirjasto (Liite_Id)  
);
```

***** Vikailmoitukset -taulu *****

```
CREATE TABLE Vikailmoitukset(  
  Ilmoitus_Id     INTEGER NOT NULL AUTO_INCREMENT,  
  Kayttaja_Id     INTEGER NOT NULL,  
  Asunto_Id       INTEGER NOT NULL,  
  Taloyhtio       VARCHAR(64),  
  Aloittaja_Id   INTEGER,  
  Kuittaaja_Id   INTEGER,  
  Kuvaus          TEXT,  
  Korjaus        TEXT,  
  Jatto_pvm      DATE NOT NULL,  
  Aloitus_pvm    DATE DEFAULT NULL,  
  Lopetus_pvm    DATE DEFAULT NULL,  
  Lisatieto      VARCHAR(255),  
  PRIMARY KEY (Ilmoitus_Id),  
  FOREIGN KEY (Kayttaja_Id)  
  REFERENCES Kayttaja (Kayttaja_Id),  
  FOREIGN KEY (Asunto_Id)  
  REFERENCES Aaunto (Asunto_Id),  
);
```

***** Alkuasetukset isännöintitoimistolle *****

```
INSERT INTO Isannointitoimisto VALUES('1','Amurin Willa -Isännöinti Oy','Mustanlahdenkatu  
10','33210','Tampere','03-2258181','050-5206251','y-123456789','');
```

***** Alkuasetukset rooleille *****

```
INSERT INTO Rooli VALUES('vierailija','Vieras');  
INSERT INTO Rooli VALUES('asukas_osakas','Asukas&Osakas');  
INSERT INTO Rooli VALUES('valitystoimiston_kayttaja','Välitystoimiston käyttäjä');  
INSERT INTO Rooli VALUES('huoltoyhtion_kayttaja','Huoltoyhtiön käyttäjä');  
INSERT INTO Rooli VALUES('hallituksen_jasen','Hallituksen jäsen');  
INSERT INTO Rooli VALUES('isannoitsija_paakayttaja','Isännöintitoimiston pääkäyttäjä');
```

***** Asetetaan pääkäyttäjä *****

```
INSERT INTO Kayttaja  
VALUES('1','isannoitsija_paakayttaja','admin',AES_ENCRYPT('admin','purkuavain'),'Mika','Mannin  
en','Perkiönkatu 79 a 3','33900','Tampere','0400-940012','-  
' , 'mika.manninen@pp4.inet.fi','2004-4-11','','Sivujen ylläpitäjä: mika.manninen (ät)  
pp4.inet.fi');
```