



SAVONIA

■ OPINNÄYTETYÖ- AMMATTIKORKEAKOULUTUTKINTO
TEKNIIKAN JA LIIKENTEEN ALA

TUNTIVIIHKO

Web-sovellus PHP-kielellä

TEKIJÄ: Ksenia Karhunen

Koulutusala Tekniikan ja liikenteen ala			
Koulutusohjelma Tietotekniikan koulutusohjelma			
Työn tekijä(t) Ksenia Karhunen			
Työn nimi Tuntivihko web-sovellus PHP -kielellä			
Päiväys	07.04.2014	Sivumäärä/Liitteet 39	30.09.2014
Ohjaaja(t) lehtori Keijo Kuosmanen			
Toimeksiantaja/Yhteistyökumppani(t) Yrittäjä Jari Karhunen, Konetyö J. Karhunen Oy			
<p>Tiivistelmä</p> <p>Opinnäytetyön aiheena oli toteuttaa web-sovellus Tuntivihko maanrakennusalan yritykselle Konetyö J. Karhunen Oy. Sovelluksesta piti tulla verkossa oleva järjestelmä, jonka kautta henkilökunta voi tallentaa työtuntivihkoja ja hallita työprosessi alusta laskutukseen asti. Tallennettuja työtuntivihkoja voi muokata, poistaa ja lähettää toiselle käyttäjälle. Lähetetyt työtuntivihkot tulevat kirjanpitäjälle valmiiksi laskupohjaan generoituna. Työn päätavoitteena oli luoda internetin välityksellä toimiva sovellus, joka auttaa ja nopeuttaa työtuntivihkojen päivittäistä täyttämistä.</p> <p>Tässä työssä on esitelty käytetyt tekniikat, työkalut ja fyysiset laitteet yleisellä tasolla sekä tarkemmin järjestelmän rakenne ja visuaalinen käyttöliittymä. Sovelluksen pääohjelmointikieleksi valittiin PHP. Lisäksi projektissa käytettiin JavaScript-kieltä ja jQuery-valmiskomponenttia. Sovelluksen tietokannaksi valittiin MySQL työkalu, yksi maailman suosituimmista tietokannoista ja tukee PHP:tä.</p> <p>Opinnäytetyön tuloksena saatiin valmis versio ja sovelluksen käyttöönotto Konetyö J. Karhunen Oy:n henkilökunnalle. Sovelluksen käyttö on rajoitettu koskemaan vain Konetyö J. Karhunen Oy:n sekä sen yhteistyökumppanin henkilökuntaa. Järjestelmän toteutuksessa opittuja tietoja tullaan hyödyntämään sovelluksen jatkokehityksessä.</p>			
Avainsanat Web-sovellus, Notepad++, PHP, Javascript, työtuntijärjestelmä, tuntivihko, jQuery:n koodikomponentteja			

Field of Study Technology, Communication and Transport			
Degree Programmed Degree Programme in Information Technology			
Author(s) Ksenia Karhunen			
Title of Thesis Web Application for Working Hours			
Date	7 April 2014	Pages/Appendices 39	30 September 2014
Supervisor(s) Mr Keijo Kuosmanen, Lecturer			
Client Organisation /Partners Mr Jari Karhunen, Konetyo J. Karhunen Ltd			
<p>Abstract</p> <p>The purpose of this thesis was to create a Web application to keep data of working hours with the PHP programming language. The application was commissioned by Konetyö J. Karhunen Ltd.</p> <p>The web application created a network system through which employees / managers can save information about working hours in a specific database. The save working hours reports may be edited or deleted and then sent to other end users or managers. The approved working hours reports are made available to the accounting department and can be used to check any final charges to the customers. Approved working hours reports also may also be used to generate the actual invoice form to be sent to the client.</p> <p>The purpose of this application was to give users an easier, more accurate and faster way to record and calculate the working hours of the each employee, which helps the managers create the reports and then to pass on electronically to the accountants.</p> <p>As a result of this thesis a final version of the application was released for use by the employees of Konetyö J. Karhunen Ltd. The application was created only for Konetyö J. Karhunen Ltd's use and the staff of its business partner.</p> <p>Plans for further development of the application will be made only after this project with Konetyö J. Karhunen Ltd. is fully completed.</p>			
Keywords Web application, Notepad++, PHP, Javascript, working hours system, working hours data, elements of jQuery			

SISÄLTÖ

TERMIT JA LYHENTEET	6
1 JOHDANTO	8
2 VAATIMUSMÄÄRITTELY	9
2.1 Toiminnalliset vaatimukset	9
2.2 Ulkoasuun ja käyttökokemukseen liittyvät vaatimukset.....	10
3 SOVELLUKSEN OHJELMONTI	12
3.1 HTML-sivunkuvauskieli.....	12
3.2 CSS-kieli	12
3.3 PHP-ohjelmointikieli ja sen komponentteja	12
3.4 JavaScript.....	13
3.5 jQuery-komponentteja	13
4 KÄYTETYT TYÖKALUT JA LAITTEET	16
4.1 MySQL.....	16
4.2 NotePad++.....	16
4.3 WAMP Server	16
4.4 TELERIK Test Studio.....	17
4.5 Fyysiset laitteet	18
5 TOTEUTUS.....	19
5.1 Suunnittelu	19
5.2 Käyttöliittymä.....	19
5.3 Käyttöönotto ja jälkikehitys	21
5.4 Sovelluksen rakenne.....	21
5.4.1 Kirjautuminen järjestelmään	21
5.4.2 Yleisnäkyä	22
5.4.3 Tuntivihko-sivu	23
5.4.4 Laskut-sivu	25
5.4.5 Selaus-sivu	26
5.4.6 Kansiot-sivu.....	28
5.4.7 Asiakas-sivu.....	29
5.4.8 Analyysi-sivu.....	30

5.4.9	Käyttäjän tietojen päivittäminen	31
5.4.10	Virheilmoitukset.....	31
5.4.11	Ohjeet-näkymä.....	31
6	TIETOKANNAN RAKENNE	32
6.1	Käyttäjät.....	33
6.2	Käyttöoikeudet.....	34
7	TESTAUS JA TYÖKALU TELERIK TEST STUDIO	36
8	YHTEENVETO	37
	LÄHTEET JA TUOTETUT AINEISTOT	38

TERMIT JA LYHENTEET

MYSQL

on tietokantaohjelmisto, jossa luodaan yhteyksiä taulujen välille. MySQL-tietokanta toteutetaan usein Python-, PHP- tai Perl-ohjelmointikielille. MySQL sisältää rajapinnan, joka auttaa käyttämään myös muita ohjelmointikieliä kuin C, C++, C#, Smalltalk, Java, Ruby ja TCL.

MICROSOFT VISUAL STUDIO

on Microsoftin ohjelmistonkehitysympäristö, jonka avulla voi tehdä web-, Windows- ja mobiilisovelluksia käyttämällä erilaisia ohjelmointikieliä kuten C:aa, C++:aa, C#:a ja Visual Basiciä sekä F#:a. Microsoft Visual Studioon voi integroida monia lisätyökaluja, kuten esimerkiksi Testi Studio ja Intel Visual Fortran.

PHP

on ohjelmointikieli, jonka avulla tehdään erityisesti web-sovelluksia tai verkkosivuja. Kyseinen ohjelmointikieli toimii keskenään erilaisten käyttöjärjestelmien sekä alustojen kanssa.

MYSQL WORKBENCH

on MySQL:n relaatiotietokantaohjelmiston täydennystyökalu. Sen tarkoituksena on tehdä visuaalista tietokannan suunnittelua, joka liittyy yhteen kokonaisuuteen suunnittelua, mallinnusta, luomista sekä ylläpitoa MySQL:lle.

Koodikomponentti

on useita koodin pätkiä, jotka yhdistyvät yhteen ympäristöön.

JAVASCRIPT

on ohjelmointikieli, jonka tarkoituksena on kehittää web-sovelluksia ja niiden komponentteja. JavaScript itse asiassa parantaa verkkosivujen toiminnallisuutta (Flanagan 2011, 1).

JQUERY

on avoimen lähdekoodin JavaScript-kielinen kirjasto, joka on tarkoitettu kaikille selaimille. jQuery:n avulla käsitellään ja editoidaan sovelluksen toimintoja, tehdään animaatioita, valitaan DOM-elementtejä sekä toteutetaan Ajax-sovelluksia.

NOTEPAD++

on ohjelmankehitysympäristö, joka pohjautuu avoimeen lähdekoodiin. NotePad++ toimii tekstieditorina Windows-käyttöjärjestelmällä ja tukee erilaisia ohjelmointikieliä, kuten PHP, C, C#, C++, Java, JavaScript. Tämän editorin tarkoituksena on erityisesti web-sovellusten kehittäminen ja niiden luominen.

WAMP SERVER

Windows-kehitysympäristöön soveltuva palvelinohjelmisto WAMP Server tukee Apache, PHP ja MySQL-ohjelmistoja. Sen avulla pystyy luomaan tietokantaa web-sovelluksille sekä testaamaan niiden syntaksia lähdekoodin ja tietokannan välillä. WAMP Serverin käyttö ei vaadi Internetiä.

TELERIK TEST STUDIO

on Windows-pohjainen ohjelmisto, joka toimii testauksen työkaluna. Test Studion kautta voi testata web-, pöytä- ja mobiilisovelluksia. Työkalu integroituu Visual Studioon. Test Studio tukee HTML-, AJAX-, Silverlight-, ASP.NET-, MVC-, JavaScript- ja WPF-pohjaisia sovelluksia.

PDF

on lyhennys sanoista Portable Document Format ja sen siirrettävä tiedostomuoto perustuu PostScript-kieleen. PDF-tiedostomuotoa käytetään erityisesti tulostamiseen ja sähköiseen julkaisemiseen.

1 JOHDANTO

Tämän työn tilaaja on maanrakennusalan yritys Konetyö J. Karhunen Oy. Opinnäytetyön aiheena on kehittää ja toteuttaa web-sovellus PHP-kielillä niin, että sen avulla työntekijät ja yhteistyökumppanit pystyvät sekä selaamaan että luomaan työtuntivihkoja sähköisessä muodossa. Sovelluksen nimeksi valittiin "Tuntivihko".

Konetyö J. Karhunen on karttulalainen yritys, joka on perustettu vuonna 2002. Konetyö J. Karhunen Oy tarjoaa pääasiallisesti maanrakennustöitä ja toimittaa kiviaineksia Pohjois-Savon alueella toimiville yrityksille sekä yksityisille asiakkaille. Yritys kuuluu mikroyrityksiin; sen palveluksessa on vähemmän kuin kymmenen työntekijää. Mikroyrityksia Suomessa on 264 435 eli 93,4 prosenttia.

Monella alalla työntekijät ja esimiehet käyttävät päivittäin työtuntivihkoja, joihin merkitään työaika, asiakkaan tiedot, työnkuvaus, tietoa työtehtävistä sekä esimiehen kuittaukset suoritetuista töistä. Opinnäytetyön aihe syntyi Konetyö J. Karhunen Oy:n tarpeesta saada järjestelmä, joka välttää ongelmatilanteita ja epäselvyyksiä työtuntivihkojen tekemisessä, lähettämässä sekä säilyttämisessä. Lisäksi sovelluksesta piti tulla järjestelmä, joka vähentää työtuntivihkoja paperimuotoisia.

Työn tavoitteena on luoda toimiva web-sovellus. Valmis sovellus otetaan tilaajan käyttöön.

2 VAATIMUSMÄÄRITTELY

Sovellus tuli suunnitella alusta käyttövalmiiseen versioon asti niin, että sen rakenne on selkeä ja helppokäyttöinen sekä tietokanta on looginen.

Toimeksiantajan vaatimuksena oli luoda sellainen sovellus, jonka avulla voi nopeuttaa ja helpottaa työtuntivihkojen täyttämistä, niiden lähettämistä työntekijältä esimiehelle ja esimieheltä kirjanpitäjälle Internetin välityksellä. Samalla sovellukseen tulisi tehdä asiakasluettelo sekä generoida tallennettuja työtuntivihkoja laskupohjaan PDF -tiedostona. PDF-tiedostoksi piti tarvittaessa pystyä lähettämään, tulostamaan ja liittämään kirjanpitoon ilman kirjanpitäjän lisäkäsittelyä. Kaikki luodut dokumentit tallennetaan MySQL-tietokantaan.

Sovelluksen toteutuksessa tuli ottaa huomioon jatkokehityksen mahdollisuus. Nykyaikana jatkokehityksen ajankohtainen ratkaisu on mobiiliversio.

2.1 Toiminnalliset vaatimukset

”Tuntivihko” – sovellukseen tuli toteuttaa helposti selattava navigointi eri toiminnallisuuksien välillä sekä rationaalinen, looginen ja selkeä tietokannan rakenne. Sovelluksen kaikki tiedot tallennetaan MySQL-tietokantaan.

Toteutettavat toiminnallisuudet olivat kirjautuminen; työtuntivihkojen lisäys, haku, muokkaus, poisto; asiakkaiden lisäys, haku, muokkaus, poisto; työmaiden lisäys, muokkaus, poisto; työtuntivihkojen hyväksyminen; työtuntivihkojen kirjanpitäjälle lähettäminen; käyttäjän tietojen muokkaus; työtuntivihkojen generointi laskupohjaan PDF-tiedostona.

Sovelluksen kirjautumisessa tuli tutkia käyttäjän syöttämät käyttäjätunnus ja salasana, jotka on tallennettu tietokantaan. Työtuntivihkon tekemisessä on kaksi vaihetta: työvihkon luonti sekä rivin luonti työvihkoon. Työtuntivihkoa täytettäessä jokainen luotu lomake saa ID-numeron; ilman työvihkon ID-numeroa rivin luonti epäonnistuu. Työvihkon, asiakkaan ja työmaan muokkauksessa tietokantaan piti tulla muokauspäivämäärä, joka auttaa kontrolloimaan tietohallintoa.

Käyttäjältä piti pyytää varmistamista työvihkon, työmaan tai asiakkaiden tietojen poistamiseen. Varmistamisella vältetään tärkeiden tietojen poistaminen tietokannasta epähuomiossa. Järjestelmässä piti rajoittaa käyttöoikeuksia, sillä yrityksessä on eri statuksella olevia työntekijöitä. Käyttöoikeuksien hallinta, varmenneratkaisut sekä suostumukseen perustuva tietojen luovutus ovat keinoja, jotka parantavat työntekijöiden ja yrityksen tietoturvaa (Andreasson & Koivisto ym. 2013).

Tilaaajan vaatimusten mukaan jokaiselle työtuntivihkolle piti tulla työtila tai status, jonka avulla käyttäjä voi paremmin kuvata asiakirjan tarkoitusta sekä asiakirjan työvaihetta. Kaikki työtuntivihkot on eritelty työkansioiksi: aloitettu, vahvistettu ja laskutettavaksi.

Käyttäjien oikeuksien erittely tuli toteuttaa seuraavalla tavalla: kirjanpitäjälle tuli näkymään kaikkien käyttäjien laskutettavat työvihkot kansiossa "laskutettavaksi", esimiehelle tuli näkymään kaikki kansiot ja työntekijät voivat nähdä vain kansion "aloitettu" ja omat luodut työtuntivihkot.

Työtuntivihkon luonnin jälkeen tuli siirtyä kansioon "aloitettu". Työvihkoa ei voi lähettää eteenpäin esimiehelle tarkistavaksi ja hyväksyttäväksi, jos siinä ei ole luotuja rivejä. Toisin sanoen tyhjää työvihkoa ei voi siirtää eteenpäin, mutta sen pystyy tallentamaan.

Kun vihko siirretään kansioon "vahvistettu", esimies pystyy tarkistamaan työntekijän tekemää työtä ja täyttämään työtuntivihkoa. Esimies voi hyväksyä työtuntivihkon tai palauttaa sen takaisin työntekijälle. Sen jälkeen työtuntivihko siirtyy esimiehen hyväksyttäväksi. Hyväksymisen jälkeen asiakirja on näkyvässä kansiossa "laskutettavaksi".

Työtuntivihko tulee automaattisesti näkyviin esimiehelle, jonka pitää hyväksyä se painamalla nappia "hyväksy". Napin painamisen jälkeen työtila "aloitettu" muuttuu työtilaksi "hyväksytty". Työtila "hyväksytty" tarkoittaa, että esimies on tarkistanut luodun työtuntivihkon ja kirjanpitäjä voi laskuttaa asiakasta. Työvihko, jonka työtila on "hyväksytty", generoituu laskupohjaan PDF-tiedostoksi. Sen jälkeen valmiit laskut tulevat näkyviin kirjanpitäjälle, jonka seuraavana tehtävänä on tulostaa laskut ja lähettää ne asiakkaille postitse. Laskun tulostamisen jälkeen generoidun tuntivihkon työtilaksi vaihtuu "lähetetty". Jos laskussa on korjattavia virheitä tai koko lasku tulee tarkistaa, voi kirjanpitäjä palauttaa laskun takaisin käsiteltäväksi. Laskun lähettäminen tarkistettavaksi tarkoittaa sitä, että työvihkon status myös muuttuu ja laskun numero poistuu tietokannasta. Palautetun laskun vastaanottaa käyttäjä, joka on sen luonut. Kun lasku on korjattu ja esimies sen tarkistanut, lähtee työvihko kirjanpitäjälle generoituna laskupohjaan PDF-tiedostoksi.

2.2 Ulkoasuun ja käyttökokemukseen liittyvät vaatimukset

Tilaaaja on antanut järjestelmän ulkoasun vaatimuksista selkeän kuvauksen. Sovelluksen käyttöliittymästä piti tulla hyvin looginen ja helppokäyttöinen. Sovelluksen ulkoasussa käytetyt värit ovat kelkainen, harmaa, oranssi, hopea (metalliväri) ja musta. Järjestelmän väriä valitsemalla otettiin huomioon, että suurin osan käyttäjistä on miehiä, joiden työala on kone- ja maanrakennusala. Yleisesti maanrakennusosalalla työkoneiden värit ovat suurelta osin keltaisia tai oransseja. Käyttöliittymän värin valinnassa otettiin huomioon myös tilaajan logo. Logossa on hyvin samoja värejä kuin valittu sovelluksen ulkoasuun.

Väreillä on ihmiseen yllättävän suuri vaikutus. Väri vaikuttaa meihin psykologisesti ja voi antaa hyvän tai huonon vaikutelman tuotteesta. On totta, että värit ja sävyt tehoavat moniin asioihin päätöksenteossa (Koppela & Linjama 2005, 392-397).

Looginen käyttöliittymä, tarkoitetaan hyvin suunniteltua ja rakennettua. Sovelluksessa tuli käyttää sellaisia elementtejä, jotka auttavat käyttäjää paremmin ymmärtämään elementtien tarkoitusta.

Ennen sovelluksen ulkoasun realisointia haastateltiin tilaajan työntekijöitä. Useiden työntekijöiden toivomuksia käyttöliittymän visuaaliseen suunnitteluun olivat: kännykkämäisten kuvanappien tai linkkien käyttäminen, monimutkaisen käyttöliittymän rakenteen välttäminen, iso tekstien fontti, käyttäjälle käyttöohjeet.

3 SOVELLUKSEN OHJELMONTI

Tässä osassa kerrotaan tarkemmin ohjelmointikielistä ja niiden valmiskomponenteista.

3.1 HTML-sivunkuvauskieli

HTML on sivunkuvauskieli, joka perustuu laajoihin tunnisteisiin, jotka määrittelevät web-sivujen rakenteen. HTML-sivuja voidaan näyttää lähes kaikilla tietokoneilla. Käyttämällä vain tekstieditoria ja selainta HTML-tiedoston luonti on hyvin nopeaa ja helppoa.

Tunnisteiden `<html>` ja `</html>` välillä oleva teksti määrittelee web-sivun rakenteen. Siihen laiteetaan sivujen elementtejä, esimerkiksi tekstikenttiä, nappeja, kuvia, tekstiä ja grafiikkaa. (Koppela & Linjama 2005.)

HTML-sivunkuvauskielen tuli toimia Tuntivihko-sovelluksen pohjana. Tälle pohjalle integroitiin muita ohjelmointikomponentteja ja elementtejä.

3.2 CSS-kieli

CSS eli Worldwide Web Consortium (W3C) on ohjelmointikieli, jonka avulla pystyy kuvaamaan ja muotoilemaan verkkosivujen ulkoasuja. CSS on hyvin suosittu kuvauskieli: sillä määritellään värejä, elementtien ja sivun asettelua sekä taustaa web-ratkaisussa. CSS soveltuu erilaisille laitteille ja sen selkeä erottuminen HTML:stä nopeuttaa ja helpottaa sivujen ylläpitoa. (Korpela 2008.)

3.3 PHP-ohjelmointikieli ja sen komponentteja

PHP on yksi tunnetuimmista ja suosituimmista ohjelmointikielistä. PHP auttaa toteuttamaan verkkosivuja ja web-sovelluksia. Kyseinen ohjelmointikieli on kehitetty erityisesti web-sovelluksen ja niiden skriptien rakentamista ja tekemistä varten. Niitä käytetään web-palvelimilla. Kielen syntaksi pohjautuu usein C, Java ja Perl -kieliin. PHP:n syntaksi on hyvin samanlainen kuin C:n ja Perl:n, joten ohjelmistokehittäjälle ei ole vaikea oppia sitä. (Heinisuo 2007.)

PHP:n hyvät puolet ovat korkea suorituskyky, ASPiä nopeampi toiminta, helppokäyttöisyys, ilmaisuus ja se, että PHP-koodia voidaan käyttää NT-ympäristössä ja UNIX-alustassa (Heinisuo 2007).

PHP-skriptit voidaan ajaa palvelimella erillisinä tiedostoina tai integroida HTML-sivulle. PHP onnistuu myös luomaan ja muuttamaan kuvia JPEG, GIF, PNG, PDF ja FLASH -muodoissa. PHP-kieli ja MySQL toimivat myös hyvin keskenään (Heinisuo 2007).

3.4 JavaScript

JavaScript on ohjelmointikieli, joka oli suunniteltu Active Scripting HTML -sivujen tekemistä varten. JavaScript on erittäin tärkeä osa lähes jokaista nettisivua. JavaScript-ohjelmointikieli on upotettu suoraan lähdekoodiin HTML-dokumenttiin. Tämä kieli toimii lähes kaikissa laiteympäristöissä. (Flanagan 2000).

Kielen avulla voidaan toteuttaa helposti esimerkiksi web-sivuja, ARAX, Comet, WebOS, käyttöliittymän elementtejä, sekä mobiili- ja palvelinsovelluksia. Web-sivuilla, jotka reagoivat valintoihin, tarkoitetaan esimerkiksi sitä, että käyttäjä haluaa poistaa jotakin järjestelmästä ja poistamiseen pyydetään varmistus. (Flanagan 2000.)

Tässä työssä käytettiin JavaScript-kieltä, esimerkiksi "Confirm dialog"-ikkunoita, print()-, goBack()- ja arlet()-funktioita. JavaScriptin funktiot on esitetty kuvassa 1 ja kuvassa 2.

```
<script>
function goBack ()
{
  window.history.back()
}
</script>
```

KUVA 1. JavaScript goBack()-funktio

```
<script>
function Print () {
  window.print ();
}
</script>
```

KUVA 2. JavaScript Print()-funktion

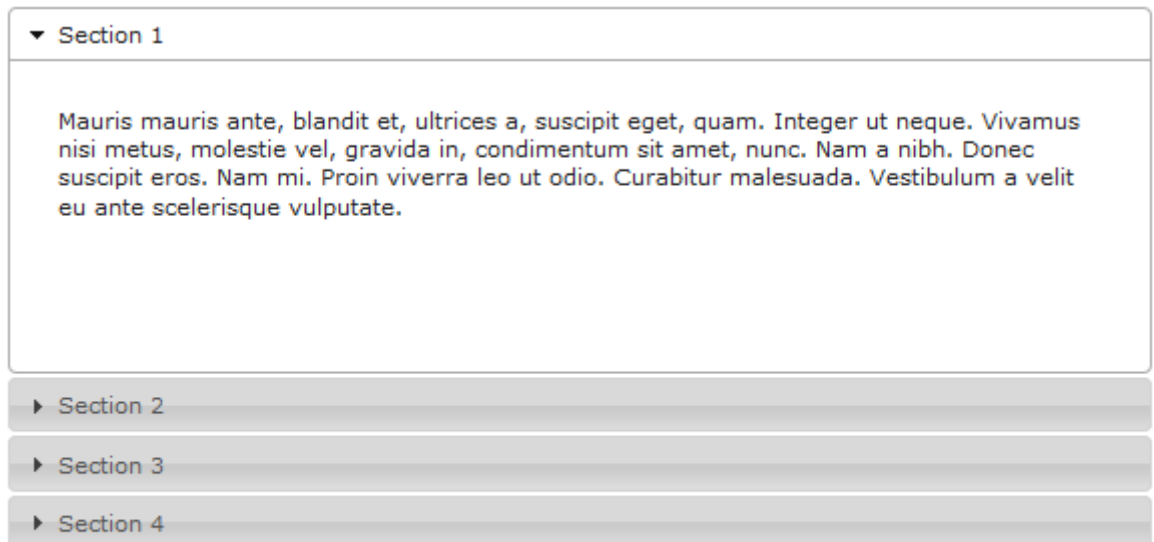
3.5 jQuery-komponentteja

Työssä käytettiin valmiina olevia JavaScript-kielen kirjastoja.

Kuvassa 3 ja 4 on esitetty Accordion-komponentti. Valitsemalla luettelopaneelistä, esimerkiksi Selection 1 tai Selection 2, käyttäjä pystyy tuomaan näkyville piilossa olevan paneelin sisällön. Klikkaamalla samalle luettelopaneelille käyttäjä piilottaa näkymästä sisällön. Projektissa Accordion-komponentti on käytetty Kansiot -sivulla.

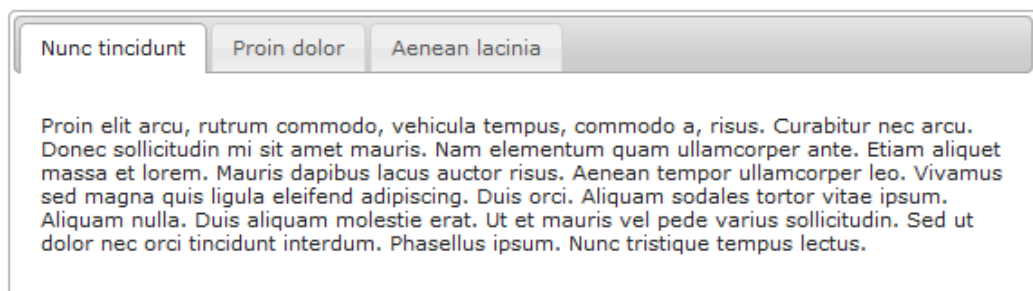
```
<script>
$(function() {
  $( document ).tooltip();
});
</script>
<style>
label {
  display: inline-block;
  width: 5em;
}
</style>
```

KUVA 3. jQuery-komponentti nimeltä Accordion (<http://jqueryui.com/>)



KUVA 4. jQuery-komponentti nimeltä Accordion Insert Caption-toiminnolla (<http://jqueryui.com/>)

Samassa kohdassa käytettiin vielä yksi jQuery-komponentti nimeltä Tabs joka kuvattu kuvassa 3. Sen pääideana on välilehtien vaihtaminen klikkaamalla niiden nimien päälle.

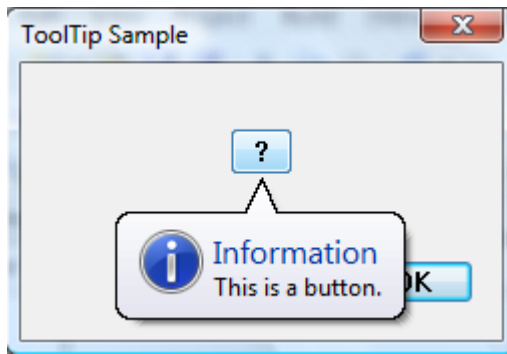


KUVA 5. jQuery-komponentti nimeltä Tabs

Hyvänä apuvälineenä oli käytetty Tooltip-komponenttia, joka ilmoittaa linkin tai elementtien tarkoituksen käyttäjälle. Tämä on hyvä tapa tehdä web-sovelluksesta helppokäyttöinen ja selkeä käyttäjälle. Se oli yksi monista tilaajan vaatimuksista. Kuvassa 6 on esitetty Tooltip-komponentin koodi ja kuvassa 7 on kuvattu sen komponentin käyttöliittymä.

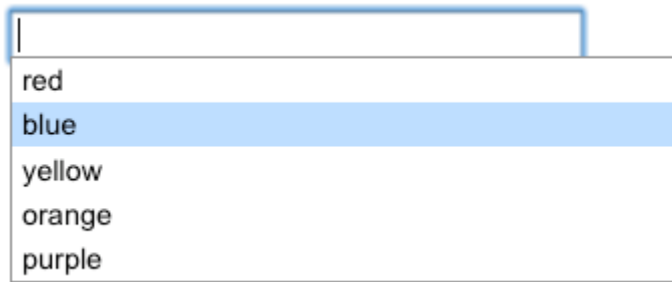
```
<script>
$(function() {
  $( document ).tooltip();
});
</script>
```

KUVA 6. jQuery-komponentti nimeltä Tooltip koodi



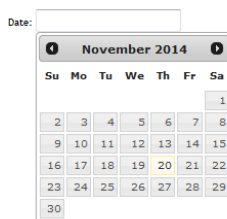
KUVA 7. jQuery-komponentti nimeltä Tooltip

Tässä projektissa JavaScript-kielistä kirjastoa nimeltä Autocomplete käytettiin monta kertaa. Sen avulla käyttäjät voivat nopeasti löytää ja valita valmiista arvoluettelosta täydentämällä vain ensimmäiset sanan kirjaimet. Tämä tapa on esitetty kuvassa 8. Työssä sain muokata tämän valmiin komponentin niin että käyttäjä voi valita tietoja suoraan tietokannasta.



KUVA 8. jQuery-komponentti nimeltä Autocomplete

Datapicker-komponentti oli sidottu vakiomuotoisen lomakkeen syöttökenttään, jossa on tarvetta täyttää päivämäärä. Kuvassa 9 on esitetty tilanne, jossa käyttäjä valitsee päivämäärän napsauttamalla syöttökenttään ja hänen näkyviinsä tulee visuaalinen kalenteri.



KUVA 9. jQuery-komponentti nimeltä Datapicker

Visuaalinen kalenteri tulee näkyviin, kun käyttäjä klikkaa kalenteri-syöttökenttään ja syöttökenttä palautuu samaan muotoon heti, kun käyttäjä on valinnut päivämäärän. Tällainen syöttötapa nopeuttaa lomakkeen täyttämistä.

4 KÄYTETYT TYÖKALUT JA LAITTEET

Tässä luvussa on käyty läpi työssä käytetyt tekniikat, työkalut ja laitteet yleisellä tasolla. Työkalujen ja laitteiden valinta on merkittävä asia työn tekemisessä, koska se vaikuttaa työtekoon, ajankäyttöön ja sovelluksen toteuttamiseen. Ennen ohjelmointityön aloittamista on hyvä miettiä, mitä työkaluja ja laitteita käytetään kehitysympäristössä. Työkalujen ja laitteiden piti olla helppoja käyttää ja sovelluksen kehittäjälle tuttuja.

4.1 MySQL

MySQL on yksi maailman suosituimmista avoimeen lähdekoodiin perustuvista tietokannoista. Se on yksi useista tietokannoista, joka tukee PHP:tä. Kyseinen työkalu on ilmainen ja riittävän tehokas ratkaisemaan todellisia ongelmia.

MySQL on palvelin, jonka avulla voi yhdistää käyttäjät etäyhteyden välityksellä. MySQL:stä löytyy web-kehittäjälle helppokäyttöinen työkalu Denwer phpMyAdmin. Tätä työvälinettä käytettäessä voidaan luoda uusi tietokanta tai taulu, täyttää taulukko, jonka tietoja voi päivittää, poistaa tai lisätä. MySQL tunnistaa tietotyyppejä, esimerkiksi numeerinen, päivämäärä, kellonaika. Kukin näistä ryhmistä on jaettu useisiin tyypeihin.

Kun tietokanta on luotu ja tarpeeksi kehitetty phpMyAdmin:n avulla, voidaan se liittää ulkoiseen web-käyttöliittymään. Web-sovellus "Tuntivihko" tehtiin pääosin PHP-kielellä, minkä vuoksi työskenteleminen MySQL:lla oli helppoa.

4.2 NotePad++

NotePad++ on tekstieditori, joka pohjautuu avoimeen lähdekoodiin. Sitä voi käyttää Windows-käyttöjärjestelmän ympäristössä. NotePad++ tukee erilaisia ohjelmointikieliä ja sitä käyttäessä voi päivittää useita tekstejä avaamalla yhtäaikaaisesti välilehtiä.

Windowsille on valtava määrä tekstieditoreja. Kevyistä tekstieditoreista suositeltavin on nykyään Notepad++, joka on muun muassa helppokäyttöinen, tehokas ja nopea.

4.3 WAMP Server

WampServer on web-kehitysympäristö, joka tulee ohjelmointikehittäjälle avuksi web-sovelluksen tekemisessä. Se mahdollistaa web-sovelluksien luonnin Apachen, PHP:n ja MySQL -tietokannan avulla. Se sisältää myös PHPMyAdminin, jolla voi helposti hallita tietokantoja. WampServer asentuu automaattisesti asennusohjelman avulla ja sen käyttö on hyvin intuitiivista. WampServer on pakettiratkaisu, joka auttaa luomaan omaa tuotantopalvelinta. Kun WampServer on asennettu, siihen on mahdollisuus lisätä uusia Apache, MySQL ja PHP -julkaisuja. Asennuksen jälkeen työpöydälle tulee myös pikakuvake, jolla voi hallita palvelinta ja sen asetuksia.

WampServer on ilmainen ja valmis kokonaispaketti, joka on tuttu opiskeluajoilta jokaiselle IT-alan ammattilaiselle. Opinnäytetyössä WampServer toimii virtuaalisena palvelimena portfoliosivustolle. Kyseinen web-kehitysympäristö asennettiin kannettavalle Samsung-tietokoneelle Windows-käyttöjärjestelmän kanssa.

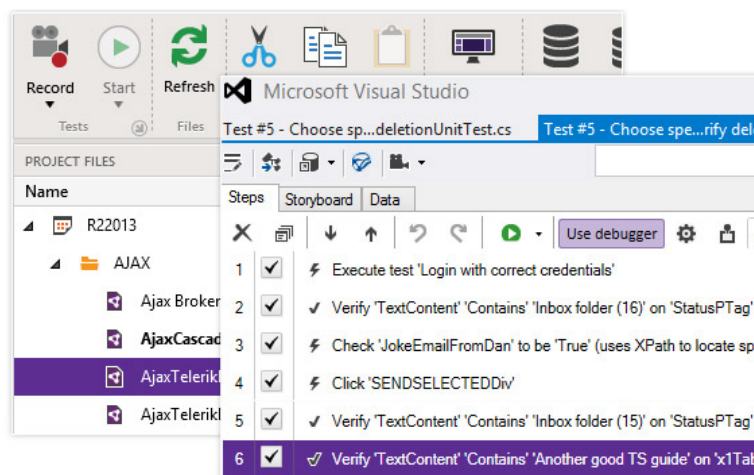
WampServer on hyvä työkalu sovelluksen testauksessa. Testauksen jälkeen SQL-lauseelle saatiin siirrettyä valmis tietokanta palvelimelle verkkoon.

4.4 TELERIK Test Studio

Test Studio on uusi ja helppokäyttöinen työkalu web-sovelluksen testauksessa. Test Studio on TELERIK-yrityksen kehittämä testiohjelmisto. Test Studio toimii Visual Studio 2012:n yhteydessä. Ilman Visual Studio – ympäristöä Test Studion käyttäminen ei ole mahdollista. Kuvassa 10 on esitetty testin ajaminen Test Studiolla.

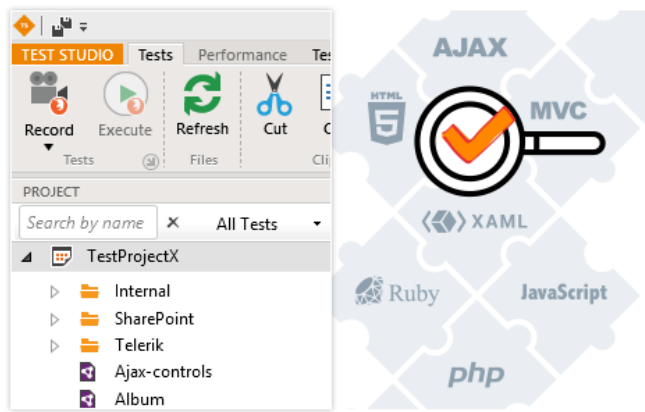
Test Studio antaa mahdollisuuden testata samaa sovellusta eri selaimilla. Tuettuja selaimia ovat:

- Internet Explorer 6+
- Mozilla Firefox 3.6+
- Google Chrome 20+
- Opera 10+
- Safari 4+.



KUVA 10. TELERIK Test Studion käyttöliittymä

TELERIK Test Studio pystyy automatisoimaan testit web- ja pöytä-sovelluksille. Test Studio tukee HTML5- , AJAX- , JavaScript- , Silverlight- , WPF- , MVC- , Ruby- ja PHP-kieliä, jotka esitetty kuvassa 11. Ohjelmointikehittäjä voi halutessaan käyttää tavallisia ohjelmointikieliä.



KUVA 11. Test Studion projekti

Test Studiossa käyttäjä perustaa testitapahtumia ja testitapauksia, jotka on mahdollista käyttää uudelleen upottamalla ne eriin testeihin, niin kuin kuvassa 11 on kuvattu. Työaikaa säästyy, koska testiä voi luoda kerran ja käyttää uudelleen moneen kertaan.

Test Studiolla on mahdollista suorittaa testit useissa koneissa samanaikaisesti sekä hyödyntää koko testausympäristöä ajoittamalla ja suorittamalla koeajoja rinnakkain. TELERIK-työkalun avulla pystyy pitämään kaikki testit ajan tasalla sovellukselle sekä tunnistaa ongelmat.

4.5 Fyysiset laitteet

Työn kehittämisessä ja testauksessa käytettiin sekä fyysisiä että virtuaalisia laitteita.

Tuntivihko-järjestelmässä käytetyt fyysiset laitteet:

- Pöytätietokone 27" Apple iMac Book -käyttöjärjestelmä, Full HD IPS LED-näyttö, Intel i5 - prosessori / 4 GHz, 8 GB RAM, 500 GB kiintolevy
- kannettava tietokone Samsung 15.6" HD LED -näyttö, 1.7 GHz, Windows7-käyttöjärjestelmänä
- Taulutietokone Apple iPad 4 Retina 16 GB
- Älypuhelin Samsung Galaxy S4 (Android-versio 4.2, API 17).

Tärkeä merkitys oli Samsungin kannettavalla tietokoneella ja Windows-käyttöjärjestelmällä, koska työn toteutuksessa käytettiin MySQL-, WAMP Server-, Visual Studio-, Telerik Test Studio- ja Note-Pad++ -työkaluja.

Applen pöytätietokoneen rooli oli merkittävä näytön suuren koon vuoksi. Kyseistä tietokonetta käytettiin eniten raportin kirjoitukseen ja materiaalien lukemiseen.

5 TOTEUTUS

5.1 Suunnittelu

Tilaaajalla oli vaatimuksena, että sovelluksen pitäisi toimia Internetin kautta. Tilaajan päätavoitteena oli tehdä sellainen verkossa oleva sovellus, jota voisi käyttää paikasta ja ajasta riippumatta.

Työn alussa tutkittiin ja mietittiin tilaajan kanssa, minkä näköinen sovelluksesta pitäisi projektin lopuksi tulla. Suunnitteluvaiheessa saatiin määriteltyä pakolliset vaatimukset ulkoasulle ja toiminnallisuudelle.

Seuraavana asiana oli päättää ja pohtia sovelluksessa käytettävät työkalut, tekniikat ja laitteet. Tämä oli yksi tärkeimmistä ratkaisuista sovelluksen toteuttamisessa. Päädyttiin tekemään sovellus pääosin PHP-kielellä ja käyttämällä jQuery-komponentteja, JavaScript- ja CSS- ohjelmointikieliä. Sovelluksen tietokannaksi valittiin MySQL.

Työajan säästämiseksi päätettiin käyttää valmiita jQuery-komponentteja. Tekniikoita ja työkaluja valittaessa otettiin huomioon niiden helppokäyttöisyys ja saatavuus.

Sovelluksen piirteiden ja kuvauksen selkiytyttyä, siirryttiin suunnitteluvaiheesta sovelluksen toteutukseen.

5.2 Käyttöliittymä

Toteutus aloitettiin domainin perustamisesta. Samassa paketissa oli mukana jo MySQL-tietokanta ja muisti. Käyttöliittymän tekeminen alkoi ulkoasun koodaamisesta. Jaettiin sivu yleiselementeiksi alku (header), jossa on yleensä navigointivalikko (navigation bar), sisältö (content) ja loppuosa (footer) kuvan 12 mukaisesti. Alku- ja loppuosassa ovat sovelluksen yleisnäkymät. Sisältö muuttuu, kun käyttäjä valitsee navigointivalikosta jonkin linkin.

HTML:n ja CSS:n avulla saatiin tehtyä sovelluksen mallipohja, joka on toiminut yleisnäköisenä. Toisin sanottuna visuaalinen näkymä oli samantyylinen kaikilla sivuilla. Kuvassa 12 on esitetty sovelluksen käyttöliittymä (Smith & Negrino 2007, 263 – 270.)

```

<body>
<!-- CONTAINER-->
<div id="container">
  <!-- HEADER-->
  <div id="header">
    <!-- LOGO-->
    <div id="logo"></div>
    <!-- TOP INFO-->
    <div id="top_info"></div>
    <!-- NAVBAR-->
    <div id="navbar"></div>
  <!-- END HEADER-->
</div>
<!-- CONTENT AREA-->
<div id="content_area">
  <!-- BANNER-->
  <div id="banner"></div>
  <!-- LEFT COL-->
  <div id="left_col"></div>
  <!-- RIGH COL-->
  <div id="righ_col"></div>

  <!-- END CONTENT AREA-->
</div>
<!-- FOOTER-->
<div id="foOter">

  <!-- END FOOTER-->
</div>

<!-- END CONTAINER-->
</div>

```

KUVA 12. Web-sovelluksen käyttöliittymä

Kun mallipohja tuli valmiiksi, siirryttiin seuraavaan vaiheeseen, toiminnallisuuteen.

Sitten siirryttiin tekemään funktiota järjestelmän visuaaliseen käyttöliittymään, joka on kuvattu kuvassa 13.



KUVA 13. Web-sovellus

5.3 Käyttöönotto ja jälkikehitys

Työn alussa suunniteltiin, että valmis tuote tulee lopputestauksen jälkeen tilaajan käyttöön. Projektin kehittyessä saatiin kuitenkin uusia ideoita sovellukselle ja päätettiin toteuttaa ne kehitystyössä, joka jatkuu vielä hieman opinnäytetyön suorituksen jälkeenkin.

5.4 Sovelluksen rakenne

Tuntivihko-sovellus koostuu taulukon 1 mukaisista osista.

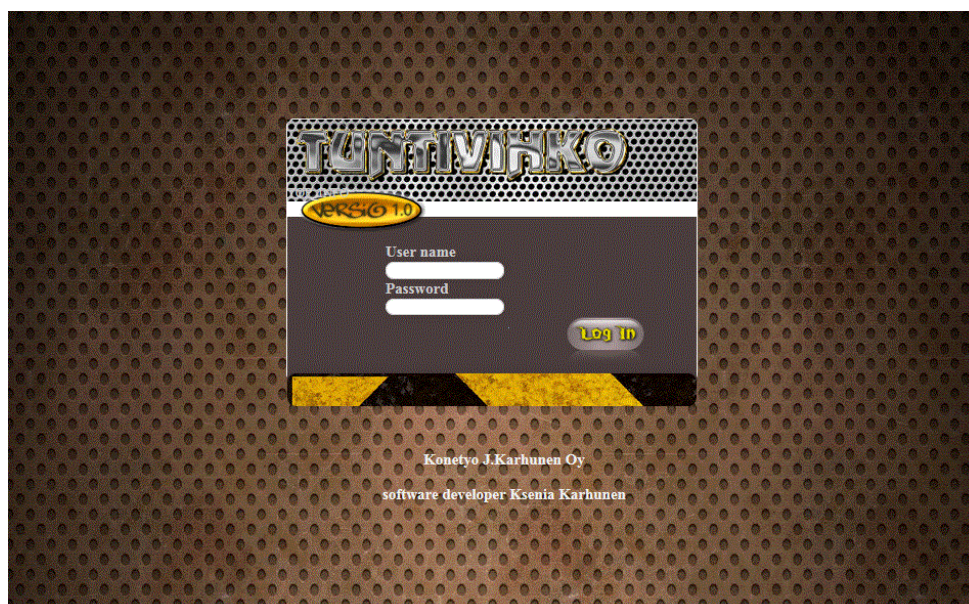
TAULUKKO 1. Tuntivihko-sovelluksen osat

Kirjautumisen sivu
Yleisnäkömä tai tausta-sivu:
Tuntivihko-sivu
Laskut-sivu
Selaus-sivu
Kansiot-sivu
Asiakas-sivu
Anaslyysi-sivu
Asetukset-sivu
Tietokanta MySQL

Järjestelmän perusrakenteeseen voisi laittaa seuraavat osat: tuntivihko, kansiot ja laskut.

5.4.1 Kirjautuminen järjestelmään

Kirjautumissivulle tuli näkömä, joka on esitetty kuvassa 14.



KUVA 14. Kirjautumisen sivu

Sovellukseen voi kirjautua vain oikealla tunnuksella ja salasanalla, jotka on kryptattu tietokantaan. Ohjelmointikoodi on rakennettu niin, että ilman LogIn-sivun kautta kirjautumista siirtyminen sovellukseen ei onnistu. Järjestelmän jokaisella sivulla tutkitaan, onko käyttäjä kirjautunut omilla tunnukillaan ja heitetään käyttäjä LogIn-sivulle, jos kirjautumista ei ollut suoritettu. Jos kirjautuminen epäonnistuu, käyttäjälle tulee dialog-ikkunaan ilmoitus virheestä. (Jorgensen 2012.)

Kun sisäänkirjautuminen sovellukseen onnistuu, käyttäjälle tulee esille järjestelmän yleisnäköy. Sivun yläpuolelle tulee käyttäjän nimi, jota klikkaamalla voidaan muokata käyttäjän tietoja.

Käyttäjä voi lopettaa työn kirjautumalla ulos järjestelmästä. Järjestelmästä ulos kirjautuminen tapahtuu navigointivalikosta löytyvää "Kirjaudu ulos" -nappia painamalla.

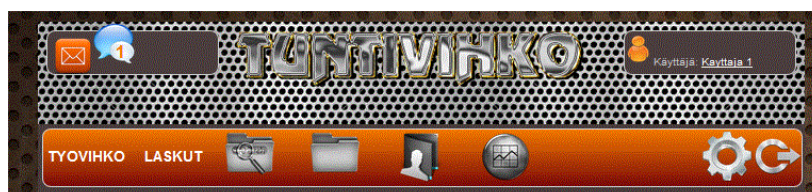
5.4.2 Yleisnäköy

Yleisnäköyillä tarkoitetaan näköyää tai taustasivua, joka on aina näkyvässä käyttäjälle muiden sivujen vaihtamisesta riippumatta. Tuntivihko-sovelluksessa sellaiseksi voi nimittää taulukossa 2 esitettyjä elementtejä.

TAULUKKO 2. Sovelluksen yleisnäköyän elementit

Elementti	Tarkoitus	Ohjelmointi
Päivämäärä – elementti	Näyttää paikallisen päivämäärä käyttäjälle.	Valmis jQuery-komponentti
Kelloaika - elementti	Näyttää paikallisen kellonajan käyttäjälle.	PHP, CSS
Kalenteri - elementti	Näyttää kalenteri käyttäjälle.	Valmis jQuery-komponentti
Navigointi valikko	Navigointi valikosta löytyy linkkejä joiden avulla pystytään vaihtamaan sovelluksen sisältöä. Kuvassa 15 ja taulukossa 3 on hyvin kuvattu navigointi valikkoa.	PHP, CSS

avassa kuvassa 15 on esitetty visuaalinen sovelluksen navigointi valikko.



KUVA 15. Sovelluksen navigointi valikko

Taulukossa 3 on kuvattu sovelluksen navigointi valikossa olevat linkit ja niiden tarkoitukset.

TAULUKKO 3. Sovelluksen navigointi valikon linkkejä

Linkin nimi	Tarkoitus
Työvihko	Työviikkojen luonti
Laskut	Laskujen tarkistaminen ja tulostaminen
Selaus	Tietojen selaus
Kansiot	On työviikkojen selaus ja työprosessin hallinta sekä asiakirjojen vaihto yrityksen välillä tai organisaatiossa.
Asiakas	Asiakkaan lisäys
Analyysi	Työn ja työtuntiviikkojen luonnin analyysia
Asetukset	Järjestelmän datan käsittely
Ulos kirjautuminen	Ulos kirjautuminen järjestelmästä ja siirtäminen LogIn-sivulle

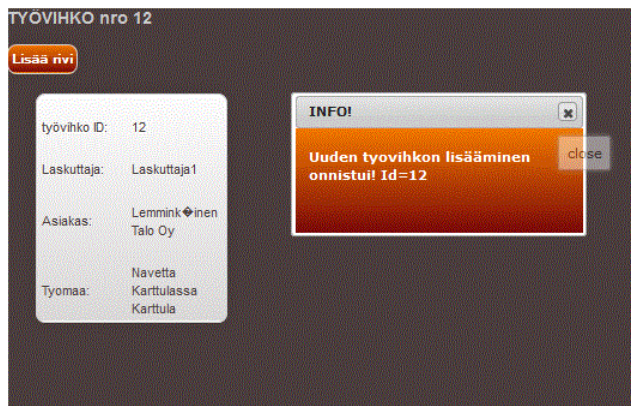
5.4.3 Tuntiviikko-sivu

Työtuntiviikon lisäys tapahtuu Työvihko-sivulla. Käyttäjälle tulee täyttämiseksi esille ”työtuntiviikon lisäys”-lomakkeen. Kyseisestä lomakkeesta löytyy tietoja taulukosta 4.

TAULUKKO 4. Lomakkeen elementtien nimet Tuntiviikko-sivulla

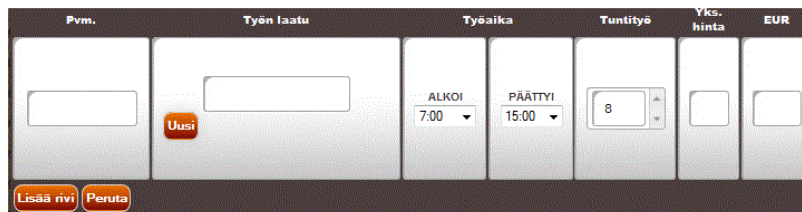
Elementin nimi	Tyyppi	Ohjelmointi ja tarkoitus
Laskuttajan nimi	TextField	”Autocomplete”-komponentti JavaScript-kielinen kirjasto
Asiakkaan nimi	TextField	”Autocomplete”-komponentti JavaScript-kielinen kirjasto
Työmaan nimi ja osoite	TextField	”Autocomplete”-komponentti JavaScript-kielinen kirjasto
Koneen nimi	TextField	”Autocomplete”-komponentti JavaScript-kielinen kirjasto
Luonnin päivämäärä	TextField	Tietokenttä on hidden-tyypellä ja päivämäärä tallentuu MySQL-tietokantaan.
Tallenna	Button	Tallentaa tietoja MySQL-tietokantaan ja siirtää seuraavaan vaihteeseen rivin luomiseen
Perua	Button	goBack-functio JavaScript:illa
Uusi	Button	Lisää uusia tietoja MySQL-tietokantaan

Kaikki työvihkon täytettävät tiedot löytyvät valitsemalla tietokannasta. Laskuttajan, asiakkaan, suorittajan, koneen ja työmaan kentissä käytettiin Autocomplete-komponentti, joka on yksi JavaScript-kielinen kirjastosta. Käyttäjä valitsee tietoja suoraan tietokannasta täydentämällä vain sanojen ensimmäisiä kirjaimia. Käyttäjällä on mahdollisuus lisätä uusia tietoja heti painamalla ”uusi”-nappia. Tallentamisen jälkeen tiedot tulevat esille käyttäjälle kenttään.



KUVA 16. Sovelluksen siirtymä kun työtuntivihko on luotu

Kuvassa 16 on esitetty sovelluksen siirtymä, kun työtuntivihko on luotu. Kun työtuntivihkon tallennus onnistuu, käyttäjä pystyy luomaan samalle työtuntivihkolle rivejä painamalla ”lisää rivi”-nappia, mikä jälkeen esille tulee kuvan 17 mukainen näkymä.



KUVA 17. työtuntivihkon rivin lisääminen

Rivin luonti tehdään täyttämällä tietoja lomakkeeseen. Kyseinen lomake on kuvattu kuvassa 17. Taulukossa 5 ”rivin luonti”-lomakkeessa on eritelty elementeiksi.

TAULUKKO 5. ”Rivin luonti”-lomake

Elementin nimi	Tyyppi	Ohjelmointi ja tarkoitus
Työn päivämäärä	TextField	”Datapicker”-komponentti JavaScript-kielinen kirjasto
Työtehtävän nimike	TextField	”Autocomplete”-komponentti JavaScript-kielinen kirjasto. Jos täytettävistä työtehtävistä ei löydy sopivaa käyttäjälle, hänellä on mahdollisuus lisätä uusi painaamalla ”uusi”-nappia

Työ alkaa klo	TextField	"Autocomplete"-komponentti JavaScript-kielinen kirjasto
Työ loppuu klo	TextField	"Autocomplete"-komponentti JavaScript-kielinen kirjasto
Tuntimäärä	TextField	Laitettiin valmiiksi työajaksi 8 tuntia (vakio). Kun käyttäjä muuttaa työajan alkamista ja loppumista, työtuntimäärä muuttuu ajan erotuksen mukaisesti
Tuntihinta	TextField	Käyttäjä itse syöttää tuntihinnan
Yhteensä työtunti	TextField	Vakio 8 tuntia. Käyttäjä voi muuttaa vakiota isommaksi tai pienemmäksi
Yhteishinta	TextField	sovellus laskee käyttäjälle valmiiksi yhteishinnan työtuntien ja tuntihinnan mukana. Yhteishinnan laskeminen tapahtuu yhtälöllä (yhteishinta = tuntimäärä x tuntihinta) kun käyttäjä täyttää tuntihintakentän.
Tallenna	Button	Tallentaa tietoja MySQL-tietokantaan ja siirtää seuraavaan vaihteeseen "Kansiot"-sivulle kun työtuntivihkolla vähemmän kuin yksi luotu rivi.
Perua	Button	goBack-funktio JavaScript:illa
Uusi	Button	Lisää uusia tietoja MySQL-tietokantaan

5.4.4 Laskut-sivu

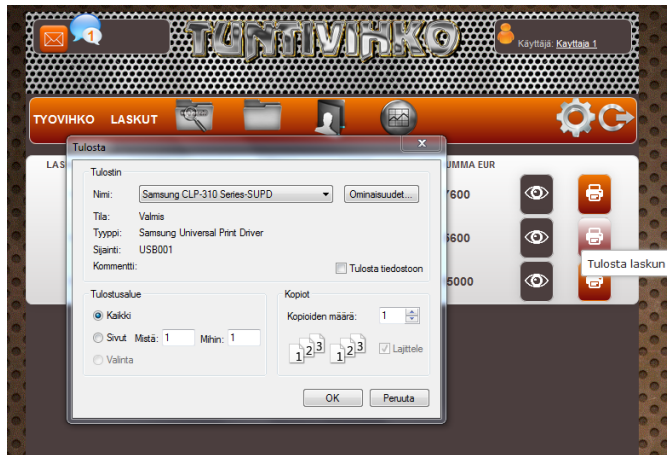
Kaikki esimiehen hyväksymät työtuntivihkot generoituvat PDF-tiedostoiksi. Laskut-sivulla käyttäjä voi esikatsella ja tulostaa tarvittavan laskun. Laskut-sivu on esitetty kuvassa 18.

LASKUN NRO	LASKUTTAJA	ASIAKAS	TYÖMAA	OSOITE	YHT SUMMA EUR
161	Laskuttaja 2	Asiakas 1	Navetta Karttussa	Karttula	7600
162	Laskuttaja1	Asiakas 1	Navetta Karttussa	Karttula	5600
163	Laskuttaja1	Asiakas 1	Työmaa 1	73200	15000

KUVA 18. Sovelluksen Laskut-sivu

Laskun generointi PDF-tiedostoksi oli toteutettu PDF generator FPDF-kirjaston avulla. Kyseinen kirjasto on ilmainen ja helppokäyttöinen. PHP-sivusta pystyttiin FPDF-kirjastoa käyttämällä toteuttamaan oikeannäköinen laskupohja PDF-tiedostoksi.

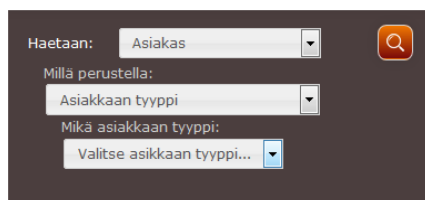
JavaScript-ohjelmointikielen Print- ja goBack-funktioita (kuvassa 19) käytettiin laskujen tulostuskäytössä.



KUVA 19. Laskun tulostamisesta Laskut-sivulla

5.4.5 Selaus-sivu

Sovelluksella pystyy hakemaan sekä työtuntivihkojen että asiakkaiden tietoja. Hakutulokset tehtiin listaksi, koska tämä käyttötapa helpottaa tietojen lukemista. Tietojen haku Selaus-sivulla on kuvattu kuvassa 20.



KUVA 20. Selaus-sivulla tietojen haku

Hakutulosten listaan tuli vain 10 viimeistä käyttäjän luomaa tuntivihkoa (kuva 21).



KUVA 21. Selaus-sivulla hakutulokset

Tarvittaessa käyttäjä voisi ladata lisää tuntivihkoja seuraavaa sivua painamalla. Kaikki työtuntivihkot jaettiin taulukon 6 mukaisiin kategorioihin.

TAULUKKO 6. Työtuntivihkon kategoriat

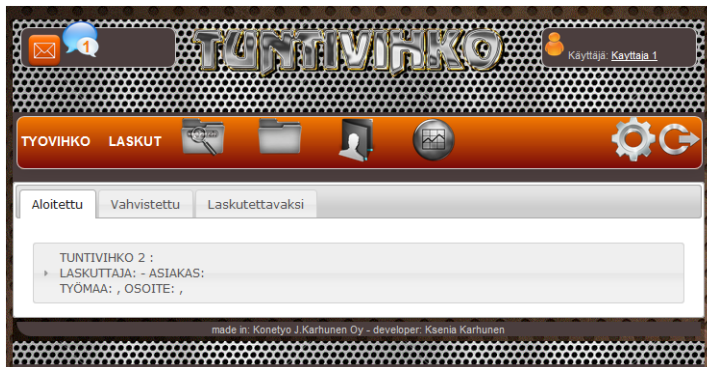
Kategorian nimi	Tarkoitus
Aloitettu	Työtuntivihko, joka on vasta luotu ja jolla on enimmän kuin yksi rivi
Palautettu	Työtuntivihko, joka on lähetetty esimieheltä takaisin tekijälle korjattavaksi.
Poistettut	Poistettu työtuntivihkot
Hyväksytyt	Työtuntivihko, jonka esimies on tarkistanut ja hyväksynyt
Lähetetty	Työtuntivihko, joka esimies on tarkistanut ja sen jälkeen lähettänyt kirjanpitäjälle laskutettavaksi
Laskutettut	Työtuntivihko, joka on generoitu PDF-tiedostoksi ja tehty valmis lasku
Virheelliset	Työtuntivihko, josta on tehty virheellinen lasku. Lasku peruutaan silloin työtuntivihko tulee takaisin esimiehelle tarkistettavaksi.

Kun käyttäjä haluaa poistaa tai muokata tietyn työviHKON hakutuloksen listassa, työtuntivihkon kategorian tai statuksen pitää olla "hyväksytyt".

Selaus-sivu auttaa käyttäjää etsimään jo tallenneltuja tietoja. Käyttäjä voi etsiä asiakkaita tai työtuntivihkoja erilaisten kriteerien perustella. Tämä asia on esitetty kuvassa 20. Asiakkaita voi etsiä esimerkiksi nimen, ID-tunnuksen tai asiakkaan tyyppin perusteella. Asiakkaan tyyppillä tarkoitetaan yksityinen tai yritys.

5.4.6 Kansiot-sivu

Työtuntien hallinta toteutettiin jakamalla kaikki työtuntivihkot aloitettu-, vahvistettu- ja laskutettavaksi –kansioiksi.



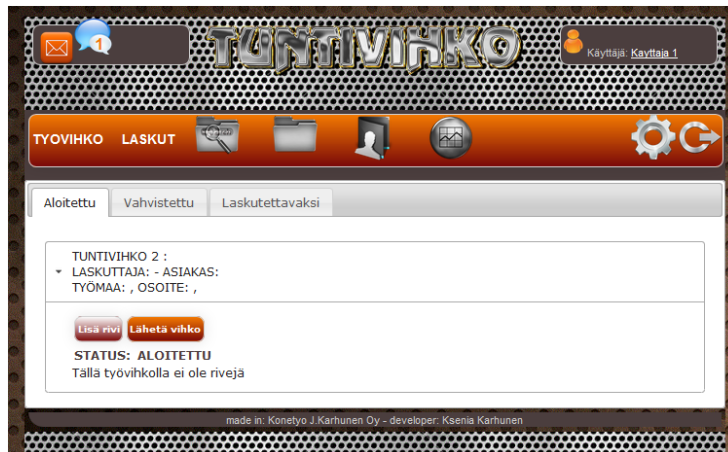
KUVA 22. Sovelluksen Kansiot-sivu

Kansiot näkyvät käyttäjälle samanaikaisesti, mutta käyttöoikeudet vaihtelevat kansioittain. Kansiot -sivun käyttöliittymän toteuttamisessa käytettiin Tabs-komponenttia JavaScript-kielisestä kirjastosta. Kuvassa 22 on kuvattu Tabs-komponentti. Jokaisessa kansiossa kaikki työtuntivihkot on esitetty listana Accordion-komponentin avulla. Kansio-sivun erittely kansioiksi on esitetty taulukossa 7.

TAULUKKO 7. Työtuntien hallinta Kansiot-sivulla

Kansion nimi	Tarkoitus	Käyttöoikeudet	Statuksen nimi
Aloitettu	Kansiossa näkyvissä ovat työtuntivihkot, jotka ei ole vielä lähetetty esimiehelle tai ovat esimiehen hyväksymättömiä.	Työntekijä, esimies	Aloitettu Palautettu
Vahvistettu	Kansiossa näkyvissä ovat esimiehen hyväksymiä ja kirjanpitäjälle lähettämiä työtuntivihkoja kirjanpitäjälle sekä virheellisiä työtuntivihkoja.	Esimies	Hyväksytty Lähetetty Virheelliset
Laskutettavaksi	Kansiossa näkyvissä on työtuntivihkoja, jotka lähetetty laskutettavaksi sekä tuntivihkoja joista on jo laskut tehty	Kirjanpitäjä, esimies	Laskutetut Lähetetty

Työtuntivihkole tulee "aloitettu"-status luomisen jälkeen, vaikka siinä ei olisi yhtään riviä. Kyseinen näkymä on esitetty kuvassa 23. Kaikkien statuksien työtuntivihkoja tai niiden tietoja voi muokata ja päivittää tarvittaessa. Sovellus säilyttää kaikki luodut ja poistetut työtuntivihkot.



KUVA 23. Sovelluksen Kansiot-sivu

5.4.7 Asiakas-sivu

Asiakas-sivun kautta käyttäjä voi päivittää ja lisätä asiakkaan tietoja asiakasrekisteriin.

Uuden asiakkaan luonnissa käyttäjä täyttää lomakkeen asiakkaan tiedoilla. Sovellus ilmoittaa käyttäjälle uusien tietojen onnistuneesta tallentumisesta tietokantaan. Kuvassa 24 lomakkeella asiakkaan lisäyksen pakolliset kentät.

ASIAKKAIDEN TIETOKANTA:

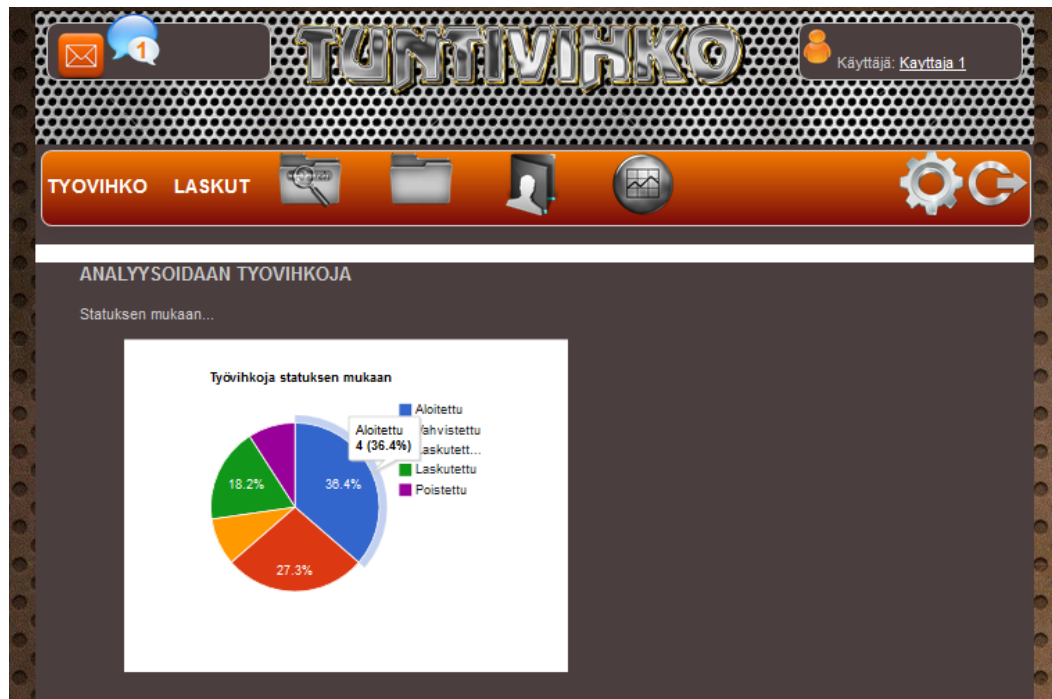
Asiakas	Asiakas 3
Yritys	<input type="radio"/>
Yksityinen	<input checked="" type="radio"/>
KÄYNTIOSOITE	
Osoite	Kurkikatu 356
Postipaikka	Tervo
Postinumero	74990
	<input checked="" type="checkbox"/> Laskutus osoite sama kuin käyntiosoite
LASKUTUSOSOITE	
Osoite	Kurkikatu 356
Postipaikka	Tervo
Postinumero	74990
Puhelin	045566777
E-mail	test@kiiu.fi
Yhteishenkilö	Nora Kurika
	<input type="button" value="Tallenna"/> <input type="button" value="Peruta"/>

KUVA 24. Asiakkaan lisäys sovelluksessa

Käyttäjällä voi päivittää tallennettuja tietoja asiakasrekisterissä. Ensimmäinen täytyy löytää asiakkaan tiedot Selaus-sivulta ja sitten tehdä päivitys. Tietojen muokkauksessa Selaus-sivulta tulee siirtyä Asiakas-sivulle.

5.4.8 Analyysi-sivu

Analyysi-sivu antaa käyttäjälle visuaalisen kuvauksen työtuntivihkojen tilanteesta. Toisin sanottuna näkymällä selkeästi eritellään työtuntivihkoja statuksien perusteella. Esimies voi tarkistaa, miten paljon työtuntivihkoja on laskutettu, montako vielä on kesken sekä mikä määrä niistä on poistettu.

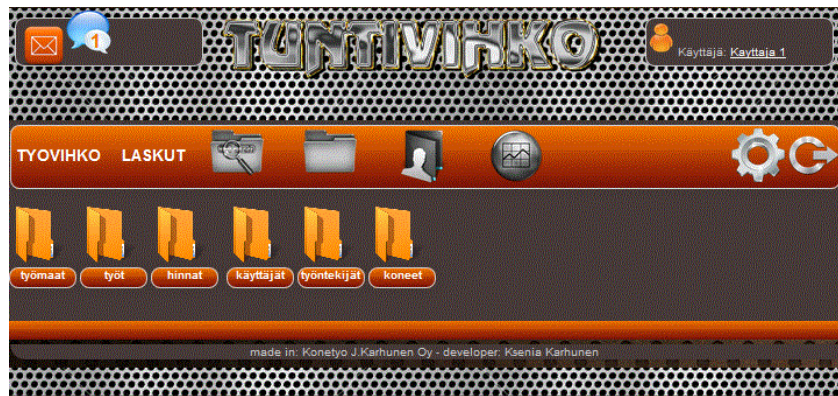


KUVA 25. Analyysi-sivu

Näkymä kuvassa 25 toteutettiin valmiista koodielementistä, joka löytyi developer.google- verkkosivulta [<https://googledevelopers.appspot.com/chart/interactive/docs/gallery/piechart#donut>]. Tämä näkymä ei ollut opinnäytetyön suunnitelmassa. Sivua tehtiin työn aikana ja jäätettiin jatkokehitykseen.

5.4.9 Asetukset-sivu

Järjestelmällä on mahdollisuus muokata tietokannan tietoja ja vaihtaa jonkinlaiset kriteerit helpottamaan järjestelmän käyttöä ja lisäämään työprosessiin tehokkuutta. Sen vuoksi päätettiin tehdä erillinen sivu, jolla voi lisätä tai päivittää dataa tietokannasta. Kuvassa 26 näkyy Asetukset-sivun kansioiksi eritelty data.



KUVA 26. Asetukset-sivu

Kaikki tiedot, jotka tarvitaan työtuntivihkon luomiseen ja työtuntien hallintaan eritellään kansioiksi. Jokaisessa kansiossa tiedot ovat esitetty listana "Accordion"-komponentin avulla.

5.4.9 Käyttäjän tietojen päivittäminen

Käyttäjä voi muokata omia tietojaan, sekä uusia käyttäjätunnuksen ja salasanan. Tälle näkymälle siirrytään käyttäjä -kuvaketta tai käyttäjän nimeä klikkaamalla. Käyttäjän kuvake sekä nimi ovat hyvin esitetty kuvassa 26 oikeassa yläkulmassa.

5.4.10 Virheilmoitukset

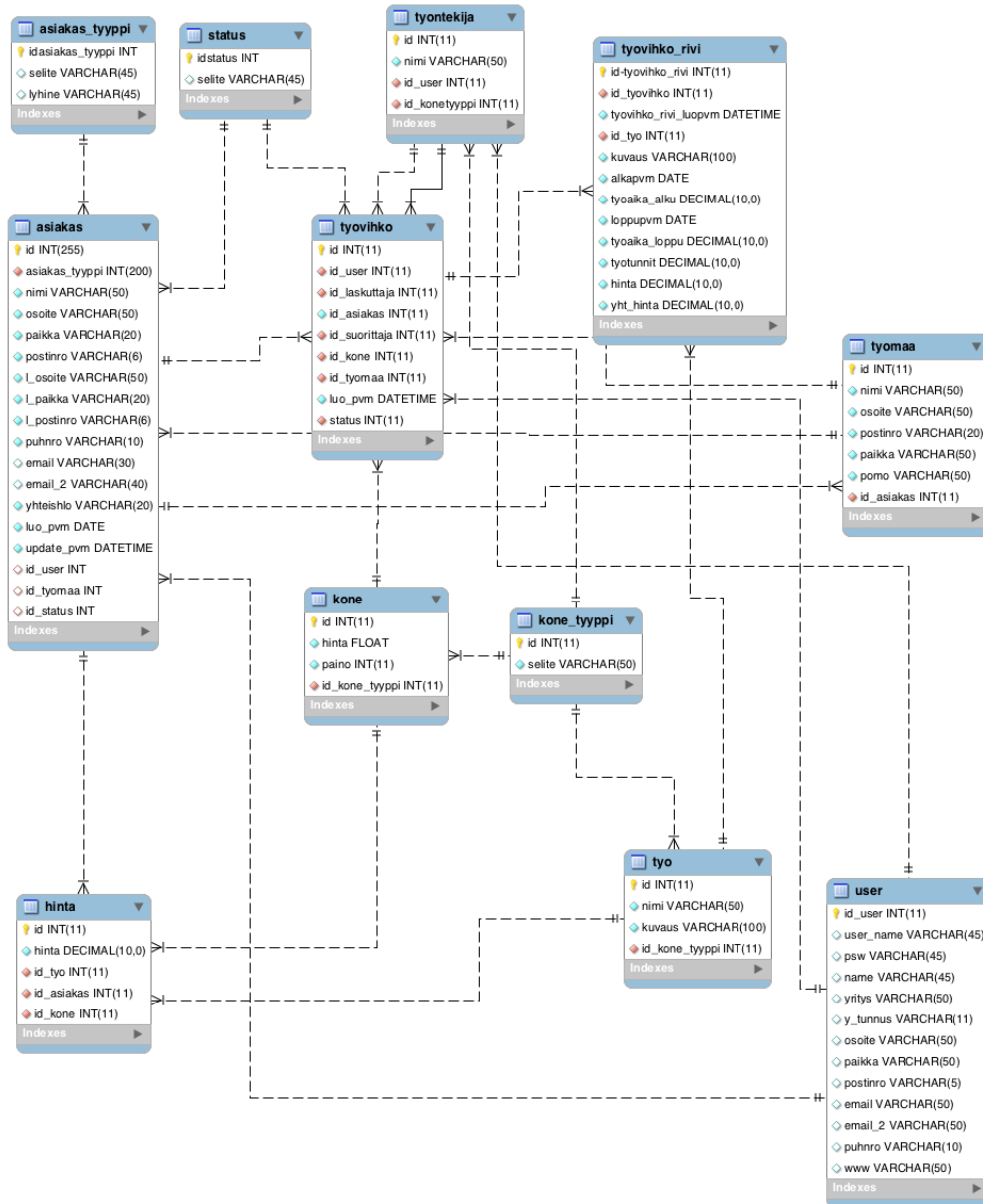
Virhe- ja infoilmoituksia toteutettiin dialogi-ikkunoiden avulla. Niiden ohjelmoinnissa käytettiin kahta eri tapaa Javascriptillä ja jQueryllä.

5.4.11 Ohjeet-näkymä

Ohjeet-näkymäksi kutsutaan tässä työssä sovelluksen apuvälinettä, jossa linkin tai elementin tarkoitus ilmoitetaan käyttäjälle viemällä hiiri linkin tai elementin päälle. Kyseinen kohta saatiin toteutettua käyttämällä "Tooltip"-komponenttia JavaScript-kielisestä kirjastosta.

6 TIETOKANNAN RAKENNE

Kuvassa 27 on esitetty sovelluksen tietokannan käsitemalli.



KUVA 27. Sovelluksen tietokannan käsitemalli

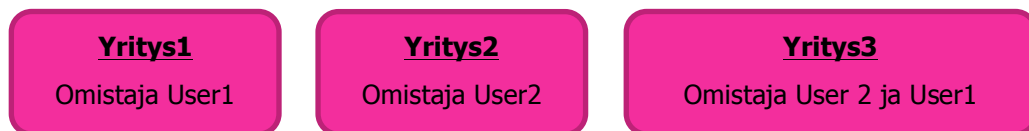
Kun käyttäjä luo uuden työtuntivihkon, se tallentuu ”työvihko”-tauluun. Uudelle työtuntivihkolle annetaan ID numero (Gilmore 2005, 618).

Seuraavana käyttäjän tulee luoda enemmän kuin yksi rivi. Jokaista riviä lisättäessä tietoja tallentuu ”työvihko_rivi”-tauluun. Jos työtuntivihkolla ei ole yhtään luotua riviä, käyttäjä ei voi lähettää sitä eteenpäin esimiehelle. (Heinisuo & Rauta 2007.)

6.1 Käyttäjät

Tuotteessa eli työn sovelluksessa tuli pystyä käsittelemään kolmen yrityksen tietoja **Yritys1, Yritys2, Yritys3**. Kyseistä web-sovellusta tuli käyttää kahden omistajan (**User1, User2**) ja kirjanpitäjän välillä (**User3**). **Yritys1**:llä ja **Yritys2**:lla on työntekijöitä, jotka tulevat käyttämään tätä sovellusta.

Tästä järjestelmästä tuli kaksimoduulinen: yksi moduuli, joka toimii työntekijän, esimiehen ja kirjanpitäjän välillä ja toinen web-ratkaisu, joka yhdistää kolmen yrityksen toimintaa ja helpottaa asiakirjan vaihtoa.



User1 henkilö, joka on yksi **Yritys3:n** omistajista ja **Yritys1:n omistaja**

User2 henkilö, joka on yksi **Yritys3:n** omistajista ja **Yritys2:n omistaja**

User3 kirjanpitäjä, joka hoitaa **Yritys1:n** ja **Yritys3:n** laskutusasiat eteenpäin.

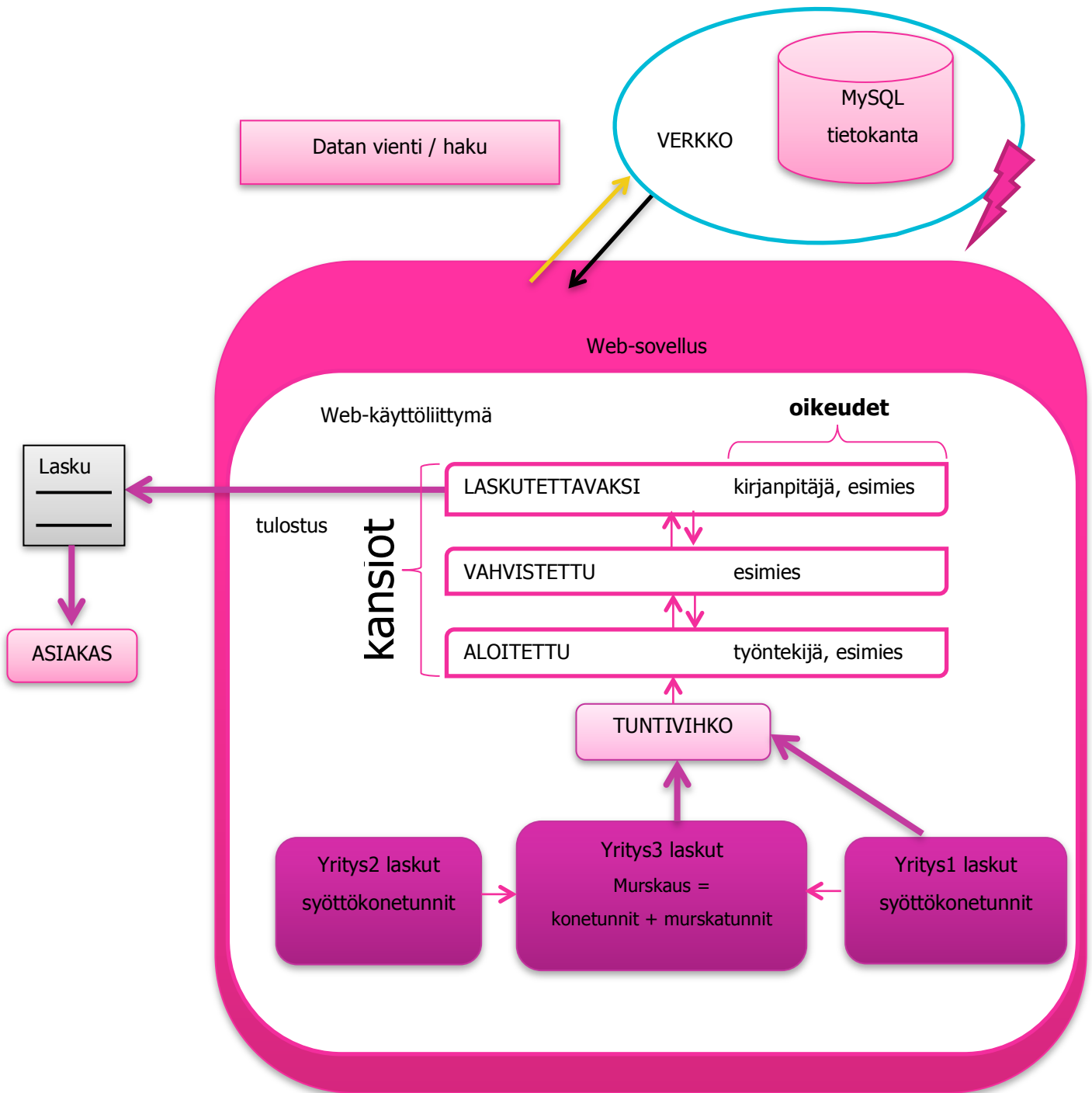
UserN **Yritys1:n** ja **Yritys3:n** työntekijät.

User1 ja User2 voivat olla sekä työntekijän että esimiehen roolissa.

Tuotteen tarkoituksena oli organisaation tuntivihkojen tekeminen, säilyttäminen sekä tuntivihkojen lähettäminen käyttäjältä toiselle. Kun **User1** ja **User2** hyväksyvät työtuntivihkoja, ne siirtyvät kirjanpitäjälle näkyviin laskutusta varten.

Yritys1 ja **Yritys2** ovat maanrakennusalan yrityksiä, jotka tarjoavat **Yritys3**:lle työkoneita (syöttökoneita) ja tarvittaessa kuorma-auton ajot sekä muut palvelut. Sopimuksen mukaan **Yritys1** ja **Yritys2** vuokraa **Yritys3**:lle aina kaksi syöttökoneita, eli kummaltakin yksi työkone **Yritys3**:lle. **Yritys1**:llä ja **Yritys2**:llä voivat olla muut asiakkaat paitsi Yritys3.

Yritys3 on yritys, jolla on murskauskone. **Yritys3** laskuttaa asiakkailta murskauksesta. Murskaustyö sisältää syöttökoneen ja murskauskoneen hinnan. Kuvassa 28 on esitetty sovelluksen kokonaisuus pohjapiirustuksessa.



KUVA 28. Sovelluksen pohjapiirustus

6.2 Käyttöoikeudet

Järjestelmässä kaikki käyttöoikeudet on jaettu käyttäjän roolin perusteella. Roolilla tarkoitetaan henkilön paikkaa yrityksen hierarkiassa. Roolit ovat työntekijä, esimies ja kirjanpitäjä. Järjestelmän oikeudet ja roolit on kuvattu taulukossa 8.

TAULUKKO 8. Oikeudet ja roolit järjestelmässä

Käyttäjä	Rooli	Henkilön määrä	Oikeudet
User1	Päärooli on esimies, joskus voi olla työntekijän roolissa	1	Kaikki sovelluksen näkymät ovat saatavissa. Pystyy näkemään muiden käyttäjien tekemät työtuntivihkot UserN , User1 , User3 .
User2	Päärooli on esimies, joskus voi olla työntekijä	1	Kaikki sovelluksen näkymät ovat saatavissa paitsi Yritys1 :n tietokannan tietoja ja työtuntivihkoja ulkopuolelle asiakkaille. Sallittu käsitellä vain omia, Yritys3 :n kaikki tietoja sekä niitä Yritys1 :n työtuntivihkoja joilla asiakkaana on Yritys3 ja työvihkon status on "hyväksyty".
User3	Kirjanpitäjä	1	Saatavissa kaikki yleisnäkymät ja kaikki sivut paitsi Kansiot-sivulla on rajoituksia. Rajoitukset koskevat "aloitettu"- ja "vahvistettu"-kansioita, jotka ovat pois näkymästä.
UserN		1-50	Saatavissa kaikki yleisnäkymät ja kaikki sovelluksen sivut paitsi Laskut-sivulle. Kansiot- ja Selaus-sivuilla on rajoituksia. Rajoitukset koskevat Kansiot-sivulla "vahvistettu"- ja "laskutettavaksi"-kansiot, jotka ovat pois näkymästä. Selaus-sivulla on asiakkaita ja työtuntivihkoja joilla status on "aloitettu" ja "palautettu".

7 TESTAUS JA TYÖKALU TELERIK TEST STUDIO

Sovelluksen toteutusvaiheessa käytettiin WAMP-palvelinta. WAMP:n avulla testattiin sovelluksen jokaisen vaiheen toiminnallisuus heti luomisen jälkeen. Tämä käytötapa säästi huomattavasti aikaa sovelluksen lopputestauksessa ja havaittavien virheiden korjaamisessa.

Testausvaihe suoritettiin järjestelmän ollessa lähes valmis. Järjestelmä testattiin ensin fyysisillä laitteilla. Sen jälkeen testattiin TELERIK Test Studio-työkalun toimiminen yhdessä Visual Studio 2013 kanssa (Jorgensen 2008).

Testauksen alussa perustettiin projekti nimellä ”työvihko” Testi Studiossa. Seuraavaksi lisättiin siihen testitapauksia, jotka toteutettiin järjestelmää käyttämällä. Tämä työväline nauhoittaa sovelluksessa kaikki käyttäjän klikkaukset ja siirtymät, jotka sitten Test Studio generoi koodiksi tai koodikomponenteiksi. Valmiit koodikomponentit liittyvät automaattisesti kyseiseen testitapaukseen tai testitapahtumaan.

Kaikki sovelluksen kohdat testattiin. Taulukossa 9 on esitetty testien tulokset Test Studiolla.

TAULUKKO 9. Sovelluksen testien tulokset

Testin nimi	Tarkoitus	Explorer	Safari	Chrome	Firefox
LogIn	Kirjautuminen sisään ja ulos, niiden siirtymät ja näkymät.	✓	✓	✓	✓
PanoramaView	Yleisnäkymät	✓	✓	✓	✓
PagesView	Kaikki sivut, näkymät ja toiminnallisuudet	✓	✓	✓	✓
UserUpdateView	Käyttäjän tietojen päivitys, siirtymä ja näkymät	✓	✓	✓	✓

Tämä testaustyökalu on erittäin yksinkertainen, säästää työaikaa ja on helppo käyttää. Kaikki testit ja testitapaukset voi myös tarvittaessa generoida raporttiin. Raportissa esitellään testien ja testitapausten nimet, testivaiheet ja niiden onnistumiset sekä kuvat web-sovelluksesta, joten merkattu testattava kohta. Kun testaus Testi Studion avulla päättyi, valmis tuote annettiin tilaajalle testiin.

8 YHTEENVETO

Opinnäytetyön tavoitteena oli toteuttaa tilaajalle, maanrakennusalan yritys Konetyö J. Karhunen Oy:lle internetin välityksellä toimiva Tuntivihko-sovellus PHP-kielillä. Järjestelmän tarkoitus oli nopeuttaa yrityksen laskuttamista sekä helpottaa asiakirjojen vaihtoa ja niiden tallentamista. Tuotteen piti toimia pääosin työntekijän, esimiehen ja kirjanpitäjän välillä. Tuntivihko-sovelluksesta saatiin tehtyä looginen web-ratkaisu, jossa oli realisoitu kaikki tilaajan vaatimukset niin käyttöliittymän visuaalisuudesta kuin toiminnallisuudestakin.

Työn toteutuksessa eniten aikaa vei suunnitteluvaihe, mutta se olikin kaikista haasteellisinta ja opettavaisinta. Projektin aikana käytettiin aihetta koskevaa englannin-, venäjän- ja suomenkielistä materiaalia. Suunnittelun jälkeen työ oli valmis siirtymään ohjelmointivaiheeseen.

Ohjelmointi aloitettiin käyttöliittymän rakentamisesta, jossa halusimme panostaa tilaajan visuaalisiin vaatimuksiin. Tuotteesta piti tulla selkeä, helppokäyttöinen ja looginen. Sen jälkeen siirryttiin sovelluksen toiminnallisuuden toteuttamiseen. Eniten työaikaa ohjelmoinnista kuului käyttöliittymän tekemiseen. Ohjelmointivaiheessa panostin erityisesti JavaScript-kieliseen kirjastoon ja sen valmiskomponenttien tutkimiseen ja oppimiseen, koska jQuery:llä työskentelystä ei ollut. Seuraavana työvaiheena oli projektin testaus, jossa käytettiin uutta ja innovatiivista testaustyökalua TELERIK Test Studio.

Tämän projektin tekeminen oli mielenkiintoinen, opettava ja haastava. Työn toteutuksessa ja sovelluksen suunnittelussa auttoi erityisesti oma työkokemus. Maanrakennusalan tuntemus ja sen toiminnan tietojen hyödyntäminen auttoivat paremmin kuvaamaan tämän työn toteuttamista.

Asiat, jotka projektin aikana opin, antavat tulevaisuudessa varmuutta uusien web-ratkaisujen toteutukseen ja auttavat kehittämään erityisesti ohjelmointitaitoja. Tässä työssä opin paljon uutta web-ratkaisun realisoinnista sekä vahvistin koodaustaitoa.

Tuotteesta tuli valmis ja toimiva ja kaikki tilaajan vaatimukset täyttyivät. Tilaaja oli tyytyväinen lopputulokseen. Järjestelmä odottaa jatkokehitystä, ennen kuin se julkistetaan.

LÄHTEET JA TUOTETUT AINEISTOT

ANDREASSON, Ari, KOIVUSTO, Juha & YLIPARTANEN, Arto 2013. Tietosuoja – vastaavan käsikirja. Helsinki: Tietosanoma Oy.

FLANAGAN, David 2000. JavaScript. Tehokäyttäjän opas, toinen panos. Suomentanut Pasi Matilainen. Espoo: Suomen Atk-kustannus Oy. Englanninkielinen alkuteos 1997.

FLANAGAN, David 2011. JavaScript – The Definitive Guide, Sixth Edition. Sebastopol: O`Reilly Media.

JORGENSEN, Paul C. 2012. Arjen tietoturva. Vinkit ja ratkaisut. Jyväskylä: Docendo.

JORGENSEN, Paul C. 2008. Software testing. A craftsman`s approach. New York: Taylor & Francis Group.

GOLMORE, Jason W. 2005. PHP5 & MySQL. Tehokas hallinta. Suomentanut Arto Kuvaja. Helsinki: Readme.fi. Englanninkielinen alkuteos 2005.

KERÄNEN, Vesa, LAMBERG, Niko & PENTTINEN, Jukka 2005. Web-julkauseminen & multimedia. Jyväskylä: Docendo Finland Oy.

KORPELA, Jukka K 2008. CSS. Verkkosivujen muotoilussa. Jyväskylä: WSOYpro/Docendo-tuotteet.

KORPELA, Jukka K & LINJAMA, Tero 2005. Web-suunnittelu. Jyväskylä: Docendo Finland Oy.

HEINISUO, Rami & RAUTA, Ilkka 2007. PHP ja MySQL. Tietokantapohjaiset verkkopalvelut. Helsinki: Talentum Oy.

SMITH, Dori & NEGRINO, Tom 2007. JavaScript. Tehokas hallinta, panos 1. Suomentanut Mikko Kamppila. Helsinki: Readme.fi. Englanninkielinen alkuteos 2007.

JQUERY – WRITE LESS, DO MORE. Ei päivitystä. [WWW-artikkeli]. [Viitattu 2014-08-30]. Saatavissa: <http://jqueryui.com/>.

TELERIK – PROGRESS COMPANY. Ei päivitystä. US. [WWW-artikkeli]. [Viitattu 2014-08-20]. Saatavissa: <http://www.telerik.com/download/teststudio>.

W3-SCHOOLS. JQUERY. Ei päivitystä [Viitattu 2014-07-08] Saatavissa: http://www.w3schools.com/jquery/jquery_intro.asp