



SAVONIA

■ OPINNÄYTETYÖ- **AMMATTIKORKEAKOULUTUTKINTO**
TEKNIIKAN JA LIIKENTEEN ALA

ROUTEWORK - AJONHALLINTAJÄRJESTELMÄ

Opinnäytetyö

TEKIJÄ/T: Julius Leppälä

Koulutusala	
Tekniikan ja liikenteen ala	
Koulutusohjelma	
Tietotekniikan koulutusohjelma	
Työn tekijä(t)	
Julius Leppälä	
Työn nimi	
RouteWork -ajonhallintajärjestelmä	
Päiväys	12.12.2014
Sivumäärä/Liitteet	37
Ohjaaja(t)	
Jussi Koistinen - lehtori, Keijo Kuosmanen - lehtori, Timo Pöllänen - toimitusjohtaja	
Toimeksiantaja/Yhteistyökumppanit(t)	
Kuljetukset ja Vuokraus Timo Pöllänen	
Tiivistelmä	
<p>Tämä työ tehtiin yritykselle Kuljetukset ja Vuokraus Timo Pöllänen toteuttamalla verkossa toimiva ajonhallintajärjestelmä, jonne voidaan tallentaa tiedot työntekijöistä, töistä ja hallita tuntikirjanpitoa. Järjestelmän on tarkoitus toimia yrityksen ostamassa web-hotellissa.</p> <p>Työ koostui suunnittelu- ja ohjelmointivaiheesta. Järjestelmän rakenne suunniteltiin kattavasti, jotta sitä voitaisiin jatkossa ylläpitää ja kehittää. Tässä työssä käydään läpi siinä käytetyt tekniikat, järjestelmän rakenne ja käyttöliittymäkuvat.</p> <p>Työn pääohjelmointikieleksi valittiin PHP, koska se on laajalti tuettu ja toimii alustasta riippumatta. Tietokantaohjelmistoksi valittiin MySQL samoista syistä. Lisäksi työssä käytettiin paljon Javascript -ohjelmointikieltä käyttöliittymän toimintojen toteuttamiseen.</p> <p>Tuloksena syntyi pilvipohjainen ajonhallintajärjestelmä, jota täytyy vielä hieman jatkokehittää, jotta sen voisi asentaa tuotantoon työn tilaajalle.</p>	
Avainsanat	
ajonhallintajärjestelmä, RouteWork, web-pohjainen, pilvipohjainen	

Field of Study	
Technology, Communication and Transport	
Degree Programme	
Degree Programme in Information Technology	
Author(s)	
Julius Leppälä	
Title of Thesis	
RouteWork Work Management System	
Date	12 December 2014
Pages/Appendices	37
Supervisor(s)	
Mr. Jussi Koistinen Lecturer, Mr. Keijo Kuosmanen Lecturer, Mr. Timo Pöllänen CEO	
Client Organisation / Partners	
Kuljetukset ja Vuokraus Timo Pöllänen	
Abstract	
<p>This project was carried out for a company called Kuljetukset ja Vuokraus Timo Pöllänen. The purpose of this project was to design and program a work management system running on a web server. The system could be used to save data from employees, their work and their working hours. The system works on a web server the company has bought.</p> <p>The project consisted of the designing and programming phase. The structure of the system was planned extensively so that it could be easily administered and developed in the future. This report covers the techniques, the structure of the system and pictures of the user interface.</p> <p>PHP was chosen as the main programming language in this project because it is widely known and supported on every platform. Based on the same reasons, MySQL was chosen for the database software. Additionally, a lot of Javascript code was used to build the functionality of the user interface.</p> <p>As a result a cloud-based work management system was created. It still needs some development in order to be ready for installation in production for the company.</p>	
Keywords	
Work management system, RouteWork, web-based, cloud-based	

SISÄLTÖ

1	JOHDANTO.....	6
2	KÄYTETYT TERMIT JA LYHENTEET	7
3	VAATIMUSMÄÄRITTELY	8
3.1	Työjärjestykset.....	8
3.2	Tuntilistat	8
3.3	Käyttäjähallinta	8
3.4	Käyttöliittymä	8
3.5	Ylläpito.....	8
4	TEKNIIKAT JA OHJELMISTOT.....	9
4.1	PHP ja PHPStorm	9
4.2	MySQL	9
4.3	Bootstrap	9
4.4	Javascript	10
4.5	jQuery.....	10
4.6	jQueryUI	10
4.7	Git	11
5	TOTEUTUS	12
5.1	Taustatutkimus	12
5.2	Suunnittelu	12
5.3	Ohjelmointi	12
5.4	Käyttöönotto	14
6	JÄRJESTELMÄN RAKENNE.....	15
6.1	Työjärjestykset.....	15
6.2	Tuntilistat	15
6.3	Ilmoitukset	16
6.4	Ylläpito.....	16
7	LUOKKARAKENNE	17

7.1	Yleiset luokat	17
7.2	Työjärjestykset.....	18
7.3	Tuntilistat	18
7.4	Ilmoitukset	19
7.5	Ylläpito.....	19
8	Tietokantarakenne	20
8.1	Käyttäjä	20
8.2	Käyttäjäoikeudet.....	20
8.3	Työt.....	21
8.4	Työjärjestykset.....	22
8.5	Tuntilistat	23
8.6	Ilmoitukset	24
8.7	Loki.....	24
9	KÄYTTÖLIITTYMÄ	25
9.1	Tietokone.....	26
9.2	Mobiililaite.....	33
10	YHTEENVETO	36

1 JOHDANTO

Tämä työ on tehty kuljetusyritykselle Kuljetukset ja Vuokraus Timo Pöllänen, jonka pääpaikka sijaitsee Heinävedellä, mutta yritystoiminnan keskus Varkaudessa. Yritys lukeutuu mikroyrityksiin 10 työntekijän kokoonpanollaan. Kuljetuskalustoa on seitsemän ajoneuvoa, joista yksi on kuorma-auto ja loput pakettiautoja.

Muutama vuosi sitten yrityksen toiminta alkoi laajenemaan sekä työmäärä lisääntymään, jolloin yrityksen toimitusjohtajan kanssa keskusteltiin jonkinlaisen järjestelmän luomisesta, jolla voisi luoda työvuorolistat työntekijöille, ja näin pitää kokonaisuuden hallinnassa. Tuloksena syntyi alkeellinen järjestelmä, jolla saatiin luotua työvuorolistat työntekijöille. Kyseinen järjestelmä ei koskaan tullut täysin valmiiksi, sillä resurssit eivät riittäneet sen loppuunkehittämiseen.

Syksyllä 2013 yrityksen toimitusjohtajan kanssa keskusteltiin uudenlaisen järjestelmän luomisesta opinnäytetyönä. Keskusteluja käytiin monta kertaa ja päädyttiin tietynlaiseen luonnokseen järjestelmästä ja sen vaatimuksista. Työn edetessä suunnitelmat muokkaantuivat työnantajan toiveiden mukaisesti.

Järjestelmä sai nimekseen RouteWork, joka tulee englanninkielien sanoista *route* (tie) ja *work* (työ). Nimi oli aluksi vain kehitysnimi järjestelmälle, mutta myöhemmin se valittiin lopputuotteen nimeksi. Nimeksi haluttiin englanninkielinen nimi siltä varalta, että järjestelmää joskus voitaisiin jatkokehittää myyntiin asti.

Lopputuotteena syntyi toimiva järjestelmä, jota on helppo käyttää niin tietokoneella kuin mobiilipäätelaitteella. Järjestelmää on myös helppo päivittää sen modulaarisuuden takia, mikäli yritys näin tulevaisuudessa haluaa.

2 KÄYTETYT TERMIT JA LYHENTEET

HTML	Hypertext Markup Language. Merkintäkieli web-sivujen luomiseen.
PHP	Hypertext Preprocessor. Palvelinpään ohjelmointikieli, jolla voidaan tuottaa web-sivulle dynaamista sisältöä.
CSS	Cascading Style Sheets. Web-sivulla käytettävä tyylitiedosto, jolla voidaan muokata sivun ulkonäköä.
SQL	Structured Query Language. Tietokantakyselykieli, jolla voidaan suorittaa toimintoja tietokannassa.
MySQL	Avoimen lähdekoodin tietokantaohjelmisto.
Bootstrap	Web-sovelluskehys, joka sisältää monta valmista komponenttia web-sivun käyttöliittymää varten.
Javascript	Asiakaspään web-sivulla suoritettava skriptikieli.
jQuery	Javascript-kielestä kehitetty koodikirjasto.
jQueryUI	Käyttöliittymäkomponentteja sisältävä Javascript-koodikirjasto.
Git	Versionhallintatyökalu
Responsiivisuus	Termi, joka tässä yhteydessä tarkoittaa web-sivun ulkonäön mukautumista sivun leveyden tai käytettävän laitteen mukaan.
Luokka	Olion tyyppi, joka määrittelee olion metodit ja muuttujat
Olio	Luokasta luotu muuttuja, jolla on sen luokan määrittelemät ominaisuudet.

3 VAATIMUSMÄÄRITTELY

3.1 Työjärjestykset

Järjestelmän ensisijaisena vaatimuksena oli työjärjestysten luonti työntekijöille ja niiden hallinnointi. Työjärjestysten luomisen piti olla nopeaa ja yksinkertaista, sekä ne oli saatava verkkoon näkyville työntekijöille. Jokaiselle työntekijälle oli luotava henkilökohtainen työjärjestys sekä esimiehelle lista, jossa näkyy kaikki työt ja niille kohdistetut työntekijät.

3.2 Tuntilistat

Tuntilistojen luonti ja ylläpito oli järjestelmän toissijaisena vaatimuksena. Työntekijöiden oli pystyttävä raportoimaan tehdyt työtunnit verkkoon, minkä mukaan joka kuukauden lopussa maksettaisiin palkka. Tuntilistoille oli luotava tulostusmahdollisuus niiden arkistointia varten.

3.3 Käyttäjähallinta

Järjestelmään oli luotava käyttäjätunnistus, jotta tiedot voitaisiin suojata ja yksilöidä. Käyttäjille oli luotava käyttöoikeudet, joiden perusteella voitaisiin sallia pääsy tiettyihin moduuleihin. Esimerkiksi työjärjestysten luontia ei voida sallia kaikille. Järjestelmään luotiin yksi järjestelmänvalvojatili, jota ei voi poistaa ja jolla on kaikki oikeudet järjestelmässä.

3.4 Käyttöliittymä

Käyttöliittymä piti luoda tietokoneen lisäksi mobiilipäätelaitteelle, sillä osa työntekijöistä käyttää järjestelmää mieluummin älypuhelimella kuin tavallisella tietokoneella. Käyttöliittymän oli oltava selkeä ja helppokäyttöinen molemmilla alustoilla. Mobiilialustalla ladattava tietomäärä oli minimoitava, jotta sovellus toimisi nopeammin. Järjestelmästä ei luotu erillistä natiivisovellusta kullekin mobiilialustalle, vaan sitä oli käytettävä kullakin mobiilialustalla olevalla Internet-selaimella.

3.5 Ylläpito

Järjestelmää on voitava ylläpitää niiden tietojen osalta, joita sinne tallennetaan. Tämä tarkoittaa sitä, että työntekijöitä, reittejä ja keikkoja voidaan lisätä, muokata sekä poistaa tarvittaessa. Ylläpito-oikeuksien on oltava rajoitettuja ja ne on voitava laittaa käyttäjäkohtaisesti.

4. TEKNIIKAT JA OHJELMISTOT

4.1 PHP ja PHPStorm

PHP-kieli on palvelinpäässä suoritettava kieli, jonka alunperin kehitti grönlantilainen Rasmus Lerforf vuonna 1995. Myöhemmin kehittäjäjoukkoon liittyi lisää ohjelmoijia ja muodostui PHP Foundation -yhteisö. PHP mahdollistaa koodin suorituksen ennen, kuin web-sivusto näytetään asiakkaalle. Tällä voidaan esimerkiksi luoda dynaamisesti muuttuvaa sisältöä web-sivustolle. Lisäksi PHP mahdollistaa web-sivuston lomakkeiden käsittelyn ja yhteyden tietokantaan. (History of PHP, 2014-10-26.)

Pääohjelmointityökaluksi valikoitui PHPStorm, jonka ominaisuudet olivat selvästi muita parempia. Toisena vaihtoehtona oli Netbeans, mutta PHPStorm oli varmatoimisempi ja nopeampi. Näiden syiden perusteella PHPStorm valikoitui ensisijaiseksi työkaluksi opinnäytetyötä varten ja sitä varten ostettiin henkilökohtainen lisenssi.

4.2 MySQL

MySQL on relaatiotietokantaohjelmisto, jonka ensimmäisen version kehittivät suomalainen Michael "Monty" Widenius ja ruotsalainen David Axmark vuonna 1995. Heillä oli yritys MySQL AB aina vuoteen 2008 asti, jolloin he myivät yrityksensä Sun Microsystems:lle miljardin dollarin hintaan. Vuonna 2008 Oracle Corporation osti Sun Microsystemsin ja on MySQL-tietokantaohjelmiston nykyinen omistaja ja kehittäjä. (History of MySQL, 2014.)

MySQL valittiin tietokannaksi, koska kyseessä on avoimen lähdekoodin tietokanta, joka toimii kaikilla alustoilla. Se on myös helppokäyttöinen ja Internetistä löytyy kattava tukiverkosto sekä paljon ohjeita sen käyttämiseksi.

4.3 Bootstrap

Bootstrap-sovelluskehiksen kehittivät alunperin amerikkalaiset Mark Otto ja Jacob Thornton ollessaan Twitterillä töissä. He huomasivat, että moni ohjelmoija käytti erilaisia koodikirjastoja käyttöliittymiä luodessaan, mikä johti suureen ylläpitokuormaan ja muihin vaikeuksiin. Tämän takia he kokosivat pienen kehittäjäporukan ja ajattelivat kehittää yhteisen koodikirjaston. Tämä projekti sai kuitenkin kuuluisuutta ja siitä julkaistiinkin avoimen lähdekoodin versio vuonna 2011. (Otto, 2012.)

Opinnäytetyössä haluttiin panostaa ulkonäköön, mutta koska ei ollut aika itse luoda näyttävää ulkoasua, Bootstrap oli tähän hyvä valinta. Siinä on valmiiksi tehdyt tyylitiedostot ja teemat, joten se nopeutti ulkoasun luontia. Toiseksi Bootstrap on jo valmiiksi tehty mobiilialustoja ajatellen, joten myös tämän takia Bootstrap tuli valituksi järjestelmän graafiseksi alustaksi.

4.4 Javascript

Javascript-kielen kehitti amerikkalainen Brendan Eich vain kymmenessä päivässä. Hänet kutsuttiin töihin Netscape Communication Corporation -yritykseen, jotta hän laittaisi heidän silloiseen web-selaimeen Scheme-ohjelmointikielen tuen. Sen sijaan hän lähti kehittämään uutta ohjelmointikieltä ja syntyi Javascript. (A Short History of JavaScript, 2012.)

Javascript on nykyään ehdoton valinta web-sovelluskehityksessä, jos haluaa liittää sivustoon käyttöliittymän toimintoja. Tämä oli myös syy valita Javascript tähän projektiin.

4.5 jQuery

jQuery on ilmainen, avoimen lähdekoodin Javascript-kielen koodikirjasto, jonka tarkoitus on yksinkertaistaa HTML-sivulla tapahtuvaa asiakaspään koodin suoritusta. Ensimmäinen versio jQuerystä julkaistiin tammikuussa 2006. Koodikirjaston silloinen kehittäjä ja ensimmäisen version julkaisija, John Resig, halusi yksinkertaistaa Javascript-kielen käyttöä ja DOM-elementtien käsittelyä HTML-sivulla. Kehittäessään jQueryä hän otti mallia Dean Edwardsin kehittämästä cssQuery -koodikirjastosta. Nykypäivänä jQueryä ylläpidetään monen kehittäjän voimin. Heitä johtaa Dave Methvin jQuery Foundation -yhteisössä. (York 2012, 28.)

```

1 //Javascript
2 var element = document.getElementById('elementti');
3 element.innerHTML = 'Hei maailma!';
4
5 //jQuery
6 $('#elementti').html('Hei maailma!')
```

KUVA 1. Tavallinen Javascript verrattuna jQuery-koodiin

jQuery valittiin myös opinnäytetyöhön toteuttamaan järjestelmän toiminnallisuuksia, sillä se yksinkertaistaisi monia käytännön asioita. Järjestelmässä lähes kaikki sivujen lataukset tapahtuvat Ajaxin kautta käyttäen jQueryn valmiita funktioita. Lisäksi kaikkien painikkeiden toiminnallisuudet ovat ohjelmoitu käyttäen jQueryä.

4.6 jQueryUI

jQueryUI on ilmainen avoimen lähdekoodin jQuery-koodikirjaston laajennus, joka sisältää paljon valmiiksi ohjelmoitua käyttöliittymäkomponentteja, animaatiotehosteita sekä teematiedostoja. Ensimmäinen versio julkaistiin syyskuussa 2007. Nykypäivänä sitä ylläpitää myös jQuery Foundation oman kehittäjäjoukkonsa kanssa. (Resig, 2007.)

jQueryUI otettiin opinnäytetyöhön mukaan, koska haluttiin helpottaa tiettyjä toiminnallisuuksia järjestelmässä. Näitä toiminnallisuuksia ovat muun muassa työntekijöiden valitseminen reiteille työjärjestyssuunnittelussa sekä reittien järjestyksen vaihtaminen työjärjestysnäkyvässä.

4.7 Git

Git on versionhallintaohjelmisto, jonka kehitti Linus Torvalds vuonna 2005 kehittäessään Linux-käyttöjärjestelmää. Hän ei ollut tyytyväinen silloisiin versiohallintaohjelmistoihin ja päätti kehittää omansa Linux-projektiansa varten. Git on ilmainen avoimen lähdekoodin ohjelmisto. (McMillian, 2005.)

Git valittiin opinnäytetyöhön muutamien vertailujen kautta. Ensiksi kokeiltiin Subversion-nimistä versionhallintaohjelmistoa, mutta se oli välillä epävakaa, jolloin päädyttiin Gitiin.

5 TOTEUTUS

5.1 Taustatutkimus

Opinnäytetyön alussa suoritettiin taustatutkimus, jotta saataisiin selville, minkälainen sovellus konkreettisesti tarvitaan. Työnantajan kanssa selviteltiin erilaisia vaihtoehtoja ja mietittiin sovelluksen vaatimuksia. Lopuksi päädyttiin web-sovellukseen, joka toimisi verkossa käyttäjätunnistuksen takana, jolloin se olisi saatavilla paikasta ja ajasta riippumatta.

Tämän jälkeen tutkittiin erilaisia sovellustekniikoita ja pohdittiin, mikä niistä olisi paras juuri opinnäytetyön tekemiseen. Lisäksi piti ottaa huomioon siihen asti kertynyt tietotaito. Päädyttiin PHP/MySQL -yhdistelmään, sillä kyseiset tekniikat ovat laajalti tuettuja verkkopalveluja tarjoavissa yrityksissä sekä ovat suhteellisen helppoja työstää. Tämän lisäksi web-sovelluksessa tarvittaisiin Javascript-koodia käyttöliittymän toiminnallisuuksien toteuttamiseen, jolloin ohjelmoinnin helpottamiseksi jQuery-koodikirjasto otettiin myös käyttöön.

5.2 Suunnittelu

Suunnittelu aloitettiin ensiksi hyvin karkealta tasolta. Mietittiin esimerkiksi, minkälaisia näkymiä järjestelmässä tarvitaan, miltä käyttöliittymän pitää näyttää, ja millä alustoilla järjestelmän tulee toimia. Kun päädyttiin siihen, että järjestelmän on toimittava sekä mobiilialustoilla että tietokoneella, käytettäviin tekniikoihin liitettiin Bootstrap-sovellusalausta. Jo alussa päädyttiin siihen, että järjestelmän tulee olla modulaarinen, jotta sen eri komponentteja voitaisiin helposti ja suhteellisen vaivattomasti päivittää, ottaa tai poistaa käytöstä sekä lisätä niitä.

Alustavan suunnittelun jälkeen oli vuorossa tietokantarakenne. Ensimmäiseksi piti miettiä, miten pieniin yksiköihin järjestelmä on kannattavaa jakaa. Päädyttiin siihen, että järjestelmä koostuisi kolmesta perusyksiköstä: työntekijöistä, reiteistä ja keikoista. Nämä olisivat järjestelmän päätaulut, joita yhdistelemällä voisi muodostaa muita tauluja, kuten työjärjestykset ja tuntilistat.

Tämän jälkeen alettiin miettimään luokka- sekä tiedostorakennetta. Päädyttiin sellaiseen rakenteeseen, että kaikille yhteiset luokat ja tiedostot tulevat juurihakemiston alle omiin kansioihin. Tämän lisäksi moduulit oli hyvä erotella omiin kansiorakenteisiinsa. Järjestelmälle luotiin yhteinen konfigurointitiedosto, jossa olisi järjestelmän kannalta välttämättömiä tietoja, joita kaikki tiedostot tarvitsevat.

5.3 Ohjelmointi

Ohjelmointityössä toteutettiin ensiksi kaikki luokat, jotka syntyivät suunnittelutyön tuloksena. Ensiksi tehtiin yleiset luokat, jotka ovat kaikille moduuleille pakollisia. Esimerkiksi ilman luokkaa DatabaseConnectivity ei mikään luokka voi yhdistää tietokantaan.

```

1  <?php
2  abstract class DatabaseConnectivity {
3      protected function openConnection() { //Yhteyden luonti }
4
5      protected function prepare() { //Kyselyn valmistelu }
6
7      protected function bind() { //Parametrin liittäminen }
8
9      protected function execute() { //Kyselyn suoritus }
10
11     protected function getResult() { //Tuloksen haku }
12
13     protected function closeConnection() { //Yhteyden sulkeminen }
14
15 }
16 ?>

```

KUVA 2. Hahmotelma DatabaseConnectivity-luokasta

Kun yleiset luokat oli tehty, siirryttiin luomaan moduulikohtaisia luokkia. Luokkien ohjelmointia helpotti huomattavasti kattava luokkasuunnitelma, jota pystyi seuraamaan. Kuitenkin tästä huolimatta suunnitelmaan tuli muutoksia, sillä muutamia asioita ei pystyttykään toteuttamaan niin kuin oli aluksi ajateltu. Esimerkiksi keikkojen oli alunperin oletettu käyttäytyvän samalla tavalla kuin reittien, mikä olisi helpottanut ohjelmointia, mutta näin ei ollutkaan, joten se vaati toteutukseen suurehkon muutoksen.

Luokkien valmistuttua siirryttiin hahmottelemaan käyttöliittymää pitäen koko ajan mielessä myös mobiililustoja. Käyttöliittymän ulkonäön ja toiminnallisuuden ohjelmointi oli kaikista työläin osa, sillä jokaiseen toimintoon kuului kuvion 1 mukaiset vaiheet, jotta se toimisi. Lisäksi jotkin asiat vaativat vielä enemmän työtä.

Ohjelmointityön keskivaiheilla otettiin käyttöön versiohallinta, jotta voitaisiin säilyttää lähdekoodit turvassa varmuuskopioituina. Sitä ennen lähdekoodit olivat tallessa ulkoisella kovalevyllä.

Loppupuolella huomattiin, että aika loppuu kesken, eikä kaikkia toimintoja ehditä opinnäytetyön aikana tehdä, joten ne karsittiin kokonaan sen sisällöstä pois. Näitä olivat muunmuassa raportointi ja viestintä -moduuli. Nämä moduulit toteutetaan myöhemmin, jos ne ovat tarpeellisia ja löytyy resursseja niiden tekemiseen.



KUVIO 1. Toiminnon ohjelmoinnin eri vaiheet

5.4 Käyttöönotto

Käyttöönotto on suunniteltu alustavasti vuoden 2015 alkupuolelle. Järjestelmästä puuttuu vielä muutamia lisättäviä osia, joita tarvitaan lopulliseen versioon. Ennen sitä pitää vielä suorittaa riittävän kattava testaus, jotta järjestelmän käyttöä estävät virheet saadaan poistettua. Tämän lisäksi pitää vielä tehdä niin sanottu konversio, eli vanhasta järjestelmästä siirretään kaikki tärkeä tieto uuteen järjestelmään. Tällainen prosessi vaatii oman työkalunsa, joka pitää vielä ohjelmoida.

6 JÄRJESTELMÄN RAKENNE

RouteWork-järjestelmä koostuu erillisistä moduleista, jotka voidaan tarvittaessa kytkeä päälle tai pois. Tosin jotkut moduulit ovat riippuvaisia toisista moduuleista. Jokaiseen moduuliin liittyvät omat luokat ja kooditiedostot, joita kutsutaan tarvittaessa.

TAULUKKO 1. Järjestelmän moduulit ja riippuvuudet

Moduuli	Riippuvuudet
Työjärjestykset	-
Tuntilistat	Työjärjestykset
Ilmoitukset	-
Ylläpito	Pakollinen

Järjestelmän perusrakenteeseen kuuluu reitit, keikat ja työntekijät

6.1 Työjärjestykset

RouteWork-järjestelmässä on kahdenlaisia töitä: reittejä ja keikkoja. Reitit ovat viikottain toistuvia ajoja, joille voi asettaa vakiotunteja. Tästä eroaa keikat, jotka tarkoittavat yksittäistä työkeikkaa tietyssä päivänä.

Työjärjestykset ovat olennaisin osa järjestelmää, sillä sitä vartenhan se alunperin on suunniteltu. Tämä moduuli sisältää työjärjestyksien tarkasteluun sekä luomiseen liittyvät toiminnot. Työjärjestyksissä voi valita näkyville vain omat työt tai sitten kaikki työt. Oletuksena näytettävä aikaväli on viikko, mutta omia töitä voi saada näkyville neljäksi viikoksi.

Työjärjestyksien luominen vaatii tarvittavia käyttöoikeuksia, jotka järjestelmän ylläpitäjä voi antaa. Yhdelle reitille tai keikalle voi laittaa miten monta työntekijää tahansa, mutta ei kuitenkaan kahta samaa työntekijää.

6.2 Tuntilistat

Tuntilistamoduulilla voidaan seurata työntekijöiden tekemiä työtunteja ja maksaa niiden perusteella palkka. Työntekijät kirjaavat tehdyt työtuntinsa järjestelmään ja kuukauden lopussa tuntilistat tulostetaan, lukitaan ja lähetetään palkanmaksuun. Kun työntekijä on tehnyt kuukauden listansa valmiiksi, hän kuittaa sen tehdyksi. Tämän jälkeen työnantaja tulostaa tuntilistan ja merkitsee sen lukituksi, jotta muutoksia ei enää voida tehdä, koska tunnit ovat lähteneet laskutukseen.

6.3 Ilmoitukset

Järjestelmän etusivulle tulee näkyville ilmoituksia, jotka ovat järjestelmässä. Niillä voidaan ilmoittaa kaikille käyttäjille haluttuja asioita. Ilmoituksia pääsee lisäämään ja muokkaamaan, jos käyttäjällä on riittävät oikeudet järjestelmään.

6.4 Ylläpito

Ylläpitomoduulissa voidaan hallita työntekijöitä, reittejä, keikkoja ja tunti listoja. Jokainen erillinen hallintamahdollisuus on oman käyttäjäoikeuden takana, joten esimerkiksi keikkoja voi päästä hallitsemaan, mutta reittejä ei. Järjestelmä sisältää yhden ylläpitokäyttäjän, joka ei näy listoilla eikä sitä näin ollen voi poistaa.

7 LUOKKARAKENNE

7.1 Yleiset luokat

Järjestelmä sisältää luokkia, jotka ovat kaikille moduuleille yhteisiä. Nämä on järjestelmän toiminnan kannalta välttämättömiä, koska niitä käytetään kaikkialla. Tämän lisäksi tähän ryhmään kuuluu myös luokka Span, jota käytetään vain tietyissä työjärjestyksissä ja tuntilistat-moduuleissa yhteisesti.

TAULUKKO 2. Yleiset luokat

Nimi	Tyyppi	Tehtävä
Authentication	Luokka	Käyttäjätunnistuksena ja salasanan vaihdon toiminnallisuus
Session	Luokka	Istunnon ja keksien tallentaminen ja lukeminen
DatabaseConnectivity	Abstrakti luokka	Tietokannan käsittelyyn vaadittavat metodit
DatabaseMethods	Rajapinta	Pakolliset metodit, jotka tarvitaan tietojen välittämiseen tietokantaan
SelectBuilder	Luokka	SELECT-kyselyn helpompi luonti
Log	Luokka	Järjestelmään tehtyjen muutoksien tallentaminen lokiin
Input	Luokka	POST ja GET-tilaukista arvojen helpompi hakeminen
ModalBuilder	Luokka	Bootstrap-modaali-ikkunoiden helpompi luominen
Module	Luokka	Moduulin lataamisen tarvittavat metodit
ModuleLibrary	Luokka	Järjestelmän moduuleiden lataaminen ja käyttöönotto
Permission	Luokka	Käyttöoikeuksien hallinnointi ja tarkistus
Person	Luokka	Työntekijän tietojen käsittely
People	Luokka	Mönen työntekijän yhtäaikainen käsittely
Route	Luokka	Reitin tietojen käsittely
Routes	Luokka	Mönen reitin yhtäaikainen käsittely
Gig	Luokka	Keikan käsittely
Gigs	Luokka	Mönen keikan yhtäaikainen käsittely
Span	Abstrakti luokka	Aikavälin hallintaan liittyviä yhteisiä apumetodeja

7.2 Työjärjestykset

Työjärjestys-moduulin luokkien rakenne muodostuu hierarkisesti pienemmästä isompaan. Toisin sanoen järjestyksessä isompi luokka sisältää aina pienemmän luokan tiedot. Tällainen jaottelu tehtiin, jotta tiedon käsittely voitaisiin mahdollisimman pitkälle hoitaa luokan sisällä. Tähtäimessä oli, että näkyvää sivua luodessa tarvitsisi ainoastaan käydä tiedot läpi silmukassa ja tulostaa. Tämän lisäksi työt oli oltava mahdollista näyttää työntekijän, reitin tai keikan mukaan. Luokat WorkspanPerson ja WorkspanRoute perivät luokan Span yleisistä luokista.

TAULUKKO 3. Työjärjestys-moduulin luokat

Nimi	Tyyppi	Tehtävä
WorkRoute	Luokka	Sisältää yhden työntekijän, yhden reitin ja tietyn päivän liitoksen
WorkdayRoute	Luokka	Sisältää yhden reitin, tietyn päivän ja sen kaikkien työntekijöiden liitoksen
WorkspanRoute	Luokka	Sisältää tietyllä aikavälillä yhden reitin ja sen kaikkien työntekijöiden tiedot
WorkspanAllRoutes	Luokka	Sisältää tietyllä aikavälillä kaikkien reittien ja niiden työntekijöiden tiedot
WorkGig	Luokka	Sisältää yhden työntekijän ja yhden keikan liitoksen
WorkdayGig	Luokka	Sisältää monen keikan tiedot tietyltä päivästä joko henkilön tai keikan mukaan
WorkspanGig	Luokka	Sisältää monen keikan ja niihin liitettyjen työntekijöiden tiedot tietyltä aikaväliltä
WorkdayPerson	Luokka	Sisältää yhden työntekijän kaikki yhden päivän työt reiteiltä ja keikoilta
WorkspanPerson	Luokka	Sisältää yhden työntekijän työt reiteiltä ja keikoilta tietyllä aikavälillä
WorklistVisibility	Luokka	Työjärjestyksen näkyvyyden hallinnointi

7.3 Tuntilistat

Tuntilistat-moduulien luokkien hierarkia muodostuu samalla tavalla kuin työjärjestykset-moduulin, eli ylempi luokka sisältää alemman luokan tiedot. Luokka HourSpan perii luokan Span yleisistä luokista, jotta sille saadaan aikavälin laskentaan tarvittavat metodit.

TAULUKKO 4. Tuntilista-moduulin luokat

Nimi	Tyyppi	Tehtävä
HourRoute	Luokka	Sisältää yhden työntekijän, yhden reitin työtuntien ja tietyn päivän liitoksen
HourGig	Luokka	Sisältää yhden työntekijän, yhden keikan työtuntien ja tietyn päivän liitoksen
HourDay	Luokka	Sisältää yhden työntekijän kaikkien reittien ja keikkojen työtunnit tietyltä päivältä
HourSpan	Luokka	Sisältää yhden työntekijän kaikkien reittien ja keikkojen työtunnit tietyltä aikaväliltä
HourStatus	Luokka	Tuntilistan tilan käsittely

7.4 Ilmoitukset

Ilmoitukset-moduuli sisältää ainoastaan kaksi luokkaa. Announcement-luokka sisältää tarvittavat metodit yhden ilmoituksen käsittelyyn ja AnnouncementCollection-luokka sisältää tarvittavat metodit monen ilmoituksen käsittelyyn ja niiden tulostamiseen näytölle.

TAULUKKO 5. Ilmoitukset-moduulin luokat

Nimi	Tyyppi	Tehtävä
Announcement	Luokka	Yhden ilmoituksen käsittely
AnnouncementCollection	Luokka	Kaikkien ilmoitusten yhtäaikainen käsittely ja niiden tulostaminen näytölle

7.5 Ylläpito

Ylläpito-moduuli koostuu vain toiminnoista, joilla käsitellään järjestelmän sisältöä, kuten reittejä, keikkoja ja työntekijöitä. Näihin toimintoihin ei tarvita erillisiä luokkia, vaan tämä moduuli käyttää suoraan yleisiä luokkia, jotka mainittiin kappaleessa 6.1.

8 TIETOKANTARAKENNE

Tietokanta koostuu 14 taulusta. Nämä on jaettu tarkoituksensa perusteella seitsemään osakokonaisuuteen, jotka sisältävät yhden tai useamman taulun. Jako selkeyttää järjestelmän rakenteen ymmärtämistä ja hahmottamista. Vierasavainviittaukset ovat merkitty tähdellä (*) jokaisen kentän nimessä.

8.1 Käyttäjä

Tämä osakokonaisuus sisältää kaikkien järjestelmän käyttäjien tiedot. Person-taulu sisältää työntekijän yleisiä tietoja ja Authentication-taulu sisältää kirjautumiseen vaadittavat tiedot.

TAULUKKO 6. Authentication-taulu.

Authentication	
Kenttä	Tarkoitus
PersonID*	ID-numero
Username	Käyttäjänimi
Password	Salasana

TAULUKKO 7. Person-taulu.

Person	
Kenttä	Tarkoitus
PersonID	ID-numero
FirstName	Etunimi
LastName	Sukunimi
Email	Sähköpostiosoite
Phone	Puhelinnumero
Available	Henkilön tila (käytössä, ei-käytössä, poistettu)
Visibility	Henkilön näkyvyys työjärjestelyssä
LastLogon	Päiväys viimeisestä kirjautumisesta
Type	Käyttäjän tyyppi
Created	Päiväys, milloin käyttäjä luotiin
CreatedBy*	Käyttäjän luojan ID
Modified	Päiväys, milloin tietoja muokattiin
ModifiedBy*	Käyttäjän muokkaajan ID

8.2 Käyttäjäoikeudet

Käyttäjäoikeudet ovat jaettu kahteen tauluun. Permission-taulu sisältää käyttöoikeuden kuvauksen ja ID-numeron. Tämän lisäksi on UserPermission-taulu, joka sisältää tiedot siitä, kenellä työntekijällä on mihinkin toimintoon käyttäjäoikeudet. Jos UserPermission-taulussa ei löydy tietystä toiminnosta merkintää käyttäjälle, hänellä ei ole oikeuksia siihen.

TAULUKKO 8. Permission-taulu.

Permission	
Kenttä	Tarkoitus
PermissionID	ID-numero
Content	Etunimi

TAULUKKO 9. UserPermission-taulu.

UserPermission	
Kenttä	Tarkoitus
PermissionID*	ID-numero
PersonID*	Käyttäjän PersonID-numero

8.3 Työt

Kuten kappaleessa 5.1 todettiin, järjestelmä pitää sisällään kahdenlaisia töitä: reittejä ja keikkoja. Tämän osakokonaisuuden taulut pitävät sisällään tiedot kaikista järjestelmässä olevista reiteistä ja keikoista. Tarvittavilla käyttöoikeuksilla niitä on mahdollista päästä luomaan, muokkaamaan sekä poistamaan.

TAULUKKO 10. Route-taulu.

Route	
Kenttä	Tarkoitus
RouteID	ID-numero
Name	Nimi
Description	Kuvaus reitistä
Days	Reitin ajopäivät
Order	Reitin järjestys työjärjestys-näkymässä
Hours_0	Maanantain vakiotunnit
Hours_1	Tiistain vakiotunnit
Hours_2	Keskiviikon vakiotunnit
Hours_3	Torstain vakiotunnit
Hours_4	Perjantain vakiotunnit
Hours_5	Lauantain vakiotunnit
Hours_6	Sunnuntain vakiotunnit
Status	Reitin tila (käytössä, piiloitettu, poisteettu)
Created	Päiväys reitin luomisesta
CreatedBy*	Reitin luojan PersonID
Modified	Päiväys reitin muokkaamisesta
ModifiedBy*	Reitin muokkaajan PersonID

TAULUKKO 11. Gig-taulu.

Gig	
Kenttä	Tarkoitus
GigID	ID-numero
Name	Käyttäjän PersonID-numero
Description	Keikan kuvaus
Date	Keikan päivämäärä
Hours_0	Arvioitu tuntimäärä keikasta
Status	Keikan tila (käytössä, piiloitettu, poistettu)
Created	Päiväys reitin luomisesta
CreatedBy*	Reitin luojaan PersonID
Modified	Päiväys reitin muokkaamisesta
ModifiedBy*	Reitin muokkaajan PersonID

8.4 Työjärjestykset

Järjestelmän olennaisin tehtävä on mahdollistaa työjärjestyksen ylläpito. Seuraavat kolme taulua on luotu tähän tarkoitukseen ja ne pitävät sisällään niihin liittyvät tiedot.

WorklistVisibility-taulu sisältää tiedot työjärjestyksen tilasta: onko se julkaistu vai vielä piilossa. Tämä mahdollistaa työjärjestyksen luomisen ja suunnittelun ilman, että kukaan käyttäjistä vielä näkee sitä.

GigSchedule-taulut sisältävät tiedot keikoista ja niihin kohdistetuista työntekijöistä. Tässä taulussa ei tarvita päivämäärätietoa, koska se on jo sisällytetty Gig-tauluun. Vastaavasti RouteSchedule-taulu sisältää tiedot reiteille kohdistetuista työntekijöistä tietyinä päivinä.

TAULUKKO 12. WorklistVisibility-taulu.

WorklistVisibility	
Kenttä	Tarkoitus
Week	Viikon numero
Year	Vuosi
Status	Työjärjestyksen tila (näkyvillä/piilossa)
LastChanged	Muutospäiväys

TAULUKKO 13. GigSchedule-taulu.

GigSchedule	
Kenttä	Tarkoitus
GigID*	Keikan GigID-numero
PersonID*	Käyttäjän PersonID-numero

TAULUKKO 14. RouteSchedule-taulu.

RouteSchedule	
Kenttä	Tarkoitus
PersonID*	Käyttäjän PersonID-numero
RouteID*	Reitin RouteID-numero
Date	Päiväys, milloin käyttäjä on kohdistettu reitille

8.5 Tuntilistat

Tuntilistojen ylläpito vaatii kolme erillistä taulua tietojen tallentamiseen. HourlistRoutes-taulussa tallennetaan tunnintiedot siitä työstä, jota työntekijä on tietyllä reitillä tietyinä päivinä tehnyt. Keikoilla tehdyt työt löytyvät omasta HourlistGigs-taulusta. Viimeisenä tauluna on HourlistStatus, joka sisältää tiedot tuntilistan tilasta: onko se kesken, valmis vai lukittu. Tuntilista lukitaan, kun tunnintiedot lähtevät palkanmaksuun.

TAULUKKO 15. HourlistRoutes-taulu.

HourlistRoutes	
Kenttä	Tarkoitus
PersonID*	Käyttäjän PersonID-numero
RouteID*	Reitin RouteID-numero
Date	Päiväys tehdystä työstä reitillä
Hours	Tehty tuntimäärä
Extra	Ylimääräinen lisätieto tehdystä työstä reitillä

TAULUKKO 16. HourlistGigs-taulu.

HourlistGigs	
Kenttä	Tarkoitus
PersonID*	Käyttäjän PersonID-numero
GigID*	Keikan GigID-numero
Date	Päiväys tehdystä työstä keikalla
Hours	Keikan päivämäärä
Extra	Ylimääräinen lisätieto tehdystä työstä keikalla

TAULUKKO 17. HourlistStatus-taulu.

HourlistStatus	
Kenttä	Tarkoitus
PersonID*	Käyttäjän PersonID-numero
Month*	Kuukausi
Year	Vuosi
Status	Tuntilistan tila (kesken/valmis/lukittu)

8.6 Ilmoitukset

Ilmoituksia varten on varattu yksi taulu, joka sisältää kaikki niihin liittyvät tiedot. Ilmoituksella on tila (Status), jonka avulla se voi olla joko piilossa tai näkyvillä.

TAULUKKO 18. Announcement-taulu.

Announcement	
Kenttä	Tarkoitus
ItemID	ID-numero
Time	Ilmoituksen julkaisupäivämäärä
Content	Ilmoituksen sisältö
Creator*	Ilmoituksen luojan ID-numero
Status	Ilmoituksen tila

8.7 Loki

Log-tauluun pitää yllä järjestelmälokiä, jonne tallentuu tiedot esimerkiksi sisään- ja uloskirjautumisista, tuntilistamerkinnoista, omien tietojen muokkauksista sekä työvuorosuunnittelusta.

TAULUKKO 19. Log-taulu.

Log	
Kenttä	Tarkoitus
LogID	ID-numero
Type	Lokityyppi (ylläpito, kirjautuminen, yleinen, järjestelmävirhe tai moduulikohtainen koodi)
Content	Keikan kuvaus
Date	Keikan päivämäärä
PersonID	Lokimerkintään liittyvän henkilön ID-numero
RemoteHost	IP-osoite, josta lokikirjaus tehtiin

9 KÄYTTÖLIITTYMÄ

RouteWork-järjestelmän käytössä huomioitiin myös mobiilikäyttäjät, jotka nykypäivänä ovat yhä kasvavampi osuus käyttäjistä. Yrityksessä työskentelevät työntekijät myös käyttävät paljon Internetiä älypuhelimellaan.

Nykypäivän trendi web-sivun käyttöliittymää suunniteltaessa on responsiivisuus. Tätä ominaisuutta on myös hyödynnetty RouteWork-järjestelmässä, jotta saataisiin kaikille näyttöko'oilte sopiva käyttöliittymä. Responsiivisuus tässä yhteydessä tarkoittaa sitä, että käyttöliittymä muuttaa muotoaan joko käytettävän laitteen perusteella tai selaimen leveyden mukaan. Tarvittaessa hyödynnetään molempia ominaisuuksia. Mobiililaitteilla näytön leveys voi olla esimerkiksi 1080 pikseliä, mutta selainikkuna luokitellaan pienemmäksi, koska muuten tekstiä olisi mahdoton lukea. Jotta mobiililaitte osaisi tehdä tällaisen skaalauksen, täytyy web-sivulle asettaa *head*-tagiin seuraavassa kuvassa esitelty asetus.

```

1 <head>
2   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
3 </head>

```

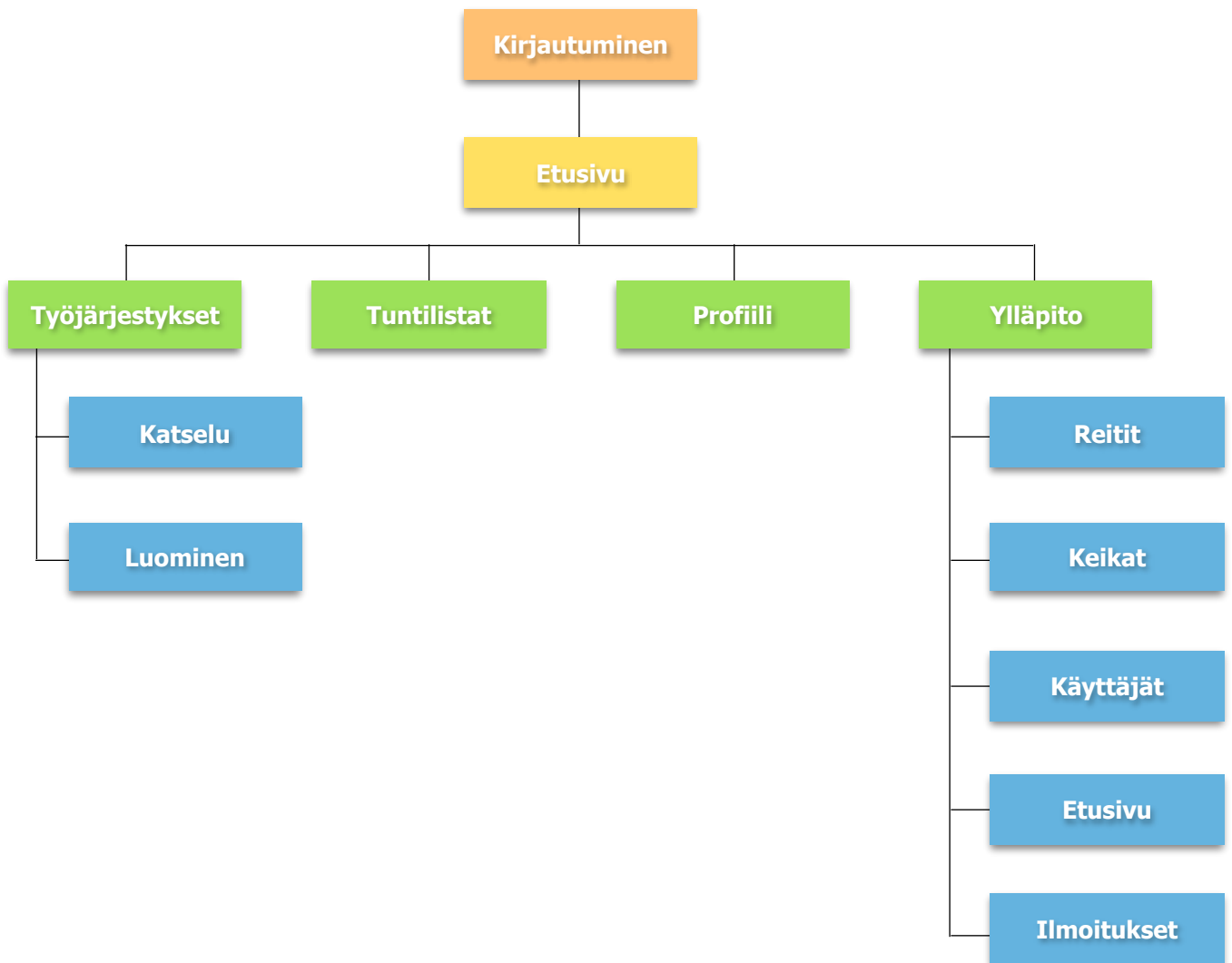
KUVA 3. Mobiilikäyttöliittymälle tarvittava HTML-koodi

RouteWork-järjestelmässä käyttöliittymä skaalautuu ja muotoutuu sen perusteella, kuinka leveä selainikkuna on. Skaalauksen katkaisukohtia on neljä, mutta järjestelmässä ajatellaan olevan vain kaksi käyttöliittymämuotoa: tietokone ja mobiililaitte. Taulukko 20 kuvaa tarkemmin käyttöliittymän skaalautuvuuden.

TAULUKKO 20. Käyttöliittymän skaalautuvuuden katkaisukohtat.

Numero	Näytön leveys d (pikseleissä)	Luokittelu
1	$d < 768$	Mobiililaitte
2	$768 \leq d < 992$	Mobiililaitte
3	$992 \leq d < 1200$	Mobiililaitte
4	$d \geq 1200$	Tietokone

Käyttöliittymän perusrakenne muodostuu kirjautumis- ja etusivusta sekä jokaisen moduulin alisivuista, jotka tulevat ilmi kuviosta 2. Mobiilikäyttöliittymään liittyy kuitenkin tiettyjä rajoituksia, joista kerrotaan enemmän omassa kappaleessaan.

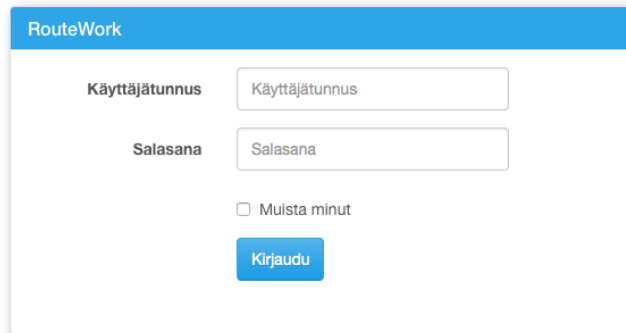


KUVIO 2. Järjestelmän sivukartta

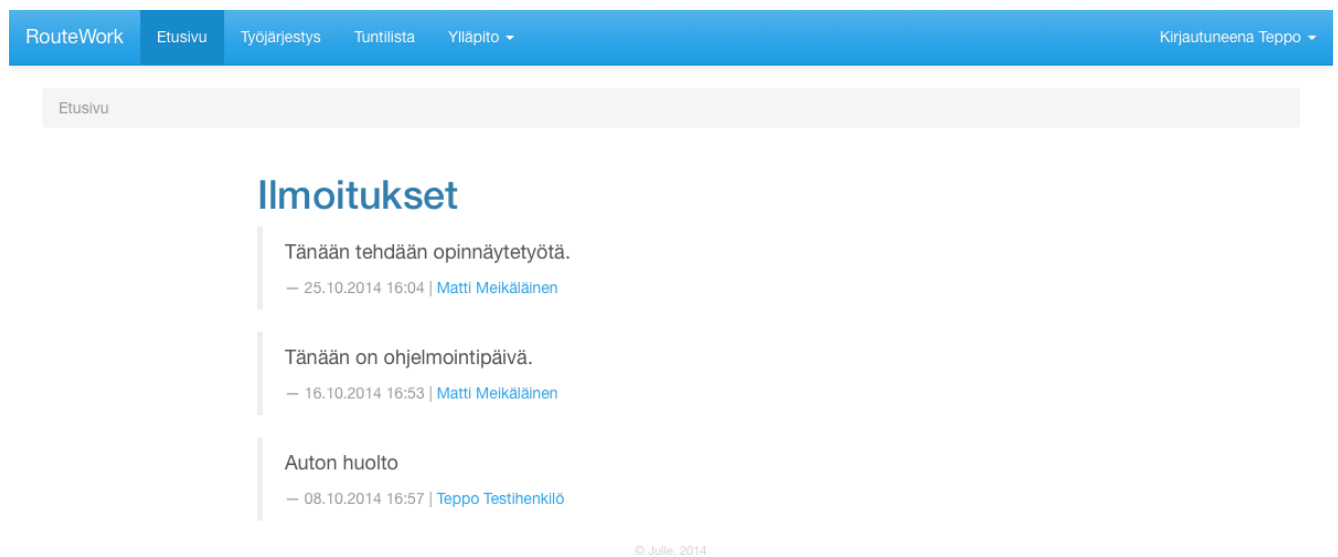
9.1 Tietokone

Selaimessa, jonka leveys on yli 1200 pikseliä, on käytössä kaikki järjestelmän ominaisuudet. Ylläpito-moduuli tarvitsee enemmän tilaa toimiakseen, joten se näkyy vain tietokoneen näytöllä. Riittävän isolla taulutietokoneen näytöllä se saattaa myös näkyä.

Seuraavaksi on listattuna käyttöliittymäkuvat, jotka näkyvät tietokoneella, kun selaimen leveys on 1215 pikseliä. Kuvat etenevät kuvion 2 mukaisesti ylhäältä alas ja vasemmalta oikealle. Tuntilistojen ylläpidosta eikä ilmoitukset-moduulista ole käyttöliittymäkuvia.



KUVA 4. Kirjautumissivu



KUVA 5. Etusivu

Järjestelmän ensimmäinen sivu on kirjautumissivu, joka on esitetty kuvassa 4. Muista minut -nappi tallentaa kirjautumisen selaimen kekseihin, jolloin seuraavalla kerralla kirjautuakseen järjestelmään ei tarvitse syöttää tunnuksia. Tämä edellyttää sitä, että järjestelmästä ei kirjauduta ulos, vaan suljetaan selainikkuna suoraan.

Kuva 5 näyttää järjestelmän etusivun, jonne siirrytään onnistuneen kirjautumisen jälkeen. Etusivulla näytetään järjestelmään luodut ilmoitukset. Etusivun ylälaudassa näkyy navigointipalkki, josta saa avattua järjestelmän muut sivut sekä tarvittaessa kirjautumaan ulos.

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Työjärjestys

Omat työt

Kaikki työt

Luo työjärjestys

Omat työt 43 / 2014 - 46 / 2014

ma	ti	ke	to	pe	la	su
20.10 Etelän auto	21.10 Etelän auto	22.10 Etelän auto	23.10	24.10	25.10 Kuopio KP2 Kuopio KP3 Muutto Kuopio - Rovaniemi	26.10 Kuopio KP2 Kuopio KP3
27.10 Etelän auto	28.10 Etelän auto	29.10 Etelän auto	30.10 Etelän auto	31.10 Etelän auto	01.11	02.11
03.11 Kuopio - Iisalmi	04.11 Kuopio - Iisalmi	05.11 Kuopio - Iisalmi	06.11 Kuopio - Iisalmi	07.11 Kuopio - Iisalmi	08.11	09.11 Kuopio KP2 Kuopio KP3
10.11 Etelän auto Kuopio KP2 Kuopio KP3 Lehtiajo Ouluun	11.11 Etelän auto Kuopio KP2 Kuopio KP3	12.11 Etelän auto Kuopio KP2 Kuopio KP3	13.11 Etelän auto Kuopio KP2 Kuopio KP3	14.11 Etelän auto Kuopio KP2 Kuopio KP3	15.11	16.11

Edellinen Valitse ajanjakso Tänään Seuraava

© Julie, 2014

Kuva 6. Omien työjärjestysten katselu

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Työjärjestys

Omat työt

Kaikki työt

Luo työjärjestys

Kaikki työt 43 / 2014

Reitti	ma 20.10	ti 21.10	ke 22.10	to 23.10	pe 24.10	la 25.10	su 26.10
Kuopio - Iisalmi	Antti	Antti	Antti	Antti	Antti		
Iltasanomat	Matti	Matti	Matti	Matti	Matti	Matti	
Etelän auto	Teppo	Teppo	Teppo	Mikko	Mikko	Mikko	Mikko
Kuopio KP2	Anna	Anna	Anna	Anna	Anna	Teppo	Teppo
Kuopio KP3	Anna	Anna	Anna	Anna	Anna	Teppo	Teppo
Muutto Kuopio - Rovaniemi						Teppo	

Edellinen Tänään Seuraava

© Julie, 2014

Kuva 7. Kaikkien työntekijöiden työt

Kuvassa 6 on näkymä, josta näkee omat työjärjestyksensä. Kalenterissa mustalla tekstillä näkyvät nimet tarkoittavat säännöllisiä töitä eli reittejä, kun taas sinisellä tekstillä kirjoitetut nimet ovat keikkoja. Kalenterin alareunassa on painikkeita, joilla voi navigoida kalenterissa. Lisäksi näkymässä voi valita ajanjakson yhden ja neljän viikon väliltä, joilta näytetään työvuorot.

Järjestelmässä voi myös selata kaikkien työntekijöiden työjärjestyksiä, kuten kuvassa 7 näkyy. Tässä näkymässä on reittien nimet myös kirjoitettu mustalla tekstillä sekä keikkojen nimet sinisellä tekstillä. Navigointi tapahtuu samalla tavalla kuin edellisessä kuvassa lukuunottamatta ajanjakson valintamahdollisuutta.

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Työjärjestys

Omat työt
Kaikki työt
Luo työjärjestys
Tallenna
Julkaistu!

Kaikki työt 43 / 2014

Reitti	ma 20.10	ti 21.10	ke 22.10	to 23.10	pe 24.10	la 25.10	su 26.10
Kuopio - Iisalmi	Antti	Antti	Antti	Antti	Antti	⊘	⊘
Iltasanomat	Matti	Matti	Matti	Matti	Matti	Matti	⊘
Etelän auto	Teppo	Teppo	Teppo	Mikko	Mikko	Mikko	Mikko
Kuopio KP2						Teppo	Teppo
Kuopio KP3						Teppo	Teppo
Muutto Kuopio - Rovaniemi	⊘	⊘	⊘	⊘			⊘

Edellinen Tanaan Seuraava

© Julle, 2014

- Anna
- Antti
- Matti
- Mikko
- Teppo
- Tyhjennä

KUVA 8. Työjärjestysten luonti

Kuvassa 8 näkyy työjärjestysten luontinäkömä. Näkymässä näytetään työjärjestys yhdeltä viikolta kerrallaan. Taulukosta voidaan valita haluttujen reittien päivät ja tämän jälkeen valita työntekijä avautuvasta valikosta päiville. Reiteiltä ja keikoilta on oletuksena estetty työntekijöiden laittaminen päiville, joihin ei ole töitä. Kuitenkin reiteiltä voi halutessaan kyseisen kiellon ohittaa. Kun työjärjestys on luotu, se pitää tallentaa sekä halutessaan julkaista, jotta se näkyisi kaikille työntekijöille.

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Tuntilista

Oma tuntilista
Kesken
Tulosta

Tuntilista lokakuu 2014

Päivä	Työ	Tunnit
1	ke	☑
2	to	☑
3	pe	☑
4	la	☑
5	su	☑
6	ma	☑
7	ti	☑
8	ke	☑
9	to	☑
10	pe	☑

- Anna
- Antti
- Matti
- Mikko
- Teppo
- Tyhjennä

KUVA 9. Tuntilistasivun yläosa

23	to			
24	pe			
25	la	Kuopio KP2, Kuopio KP3, Muutto Kuopio - Rovaniemi	14	
26	su	Kuopio KP2, Kuopio KP3	6	
27	ma	Etelän auto	4	
28	ti	Etelän auto	4	
29	ke	Etelän auto	4	
30	to	Etelän auto	4	
31	pe	Etelän auto	4	
			Tunnit	53
			Sunnuntai	6
			Yhteensä	59

Edellinen **Tänään** Seuraava

© Julie, 2014

KUVA 10. Tuntilistasivun alaosa

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Tuntilista

Oma tuntilista
Kesken
Tulosta

Tuntilista

Päivä	Työ
1	ke
2	to
3	pe
4	la
5	su
6	ma
7	ti

Merkitse tuntilistaan

Reitti	Lisätieto	Tunnit
<input checked="" type="checkbox"/> Kuopio KP2	<input type="text"/>	3 <input type="text"/>
<input checked="" type="checkbox"/> Kuopio KP3	<input type="text"/>	3 <input type="text"/>
<input checked="" type="checkbox"/> Muutto Kuopio - Rovaniemi	<input type="text"/>	8 <input type="text"/>

Tallenna Peruuta

lokakuu 2014

Tunnit

KUVA 11. Tuntilistaan merkinnän tekeminen

Kuvat 9 ja 10 esittävät tuntilistan ulkoasun. Siinä on taulukoitu valitun kuukauden päivät, niillä tehdyt työt ja tuntimäärät. Sivun vasemmassa laidassa näkyy punainen painike, josta voi valita tuntilistan tilan, kuten kappaleessa 6.2 kerrotaan. Painike vaihtaa väriä ja tekstiä tuntilistan tilan mukaan.

Tuntilistataulukon oikeassa laidassa näkyy painike, josta avautuu kuvan 11 mukainen dialogi. Tässä dialogissa voi tehdä valitulle päivälle tuntilistamerkinnän. Merkintään voi halutessaan liittää myös reittikohtaisen lisätiedon, joka myöhemmin näkyy tuntilistalla kyseisen päivän ja reitin kohdalla sulkujen sisällä.

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Profiili

Profiili

ID 3

Käyttäjänimi teppo

Etunimi Teppo ✓

Sukunimi Testihenkilö ✓

Puhelin 05001234123 ✓

Sähköposti teppo.testihenkilo@gmail.com ✓

Vaihdataanko salasana? Anna uusi salasana kaksi kertaa.

Tallenna muutokset

© Jullie, 2014

KUVA 12. Profiilisivu

Profiilisivu näyttää järjestelmään tallennetut käyttäjän tiedot. Käyttäjänimeä ja ID-numeroa ei saa vaihtaa tietojen eheyden säilymisen takia.

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Ylläpito / Reitit

Reitit

Luo reitti Poista reitti Järjestä reitit Näytä poistetut

ID	Nimi	Kuvaus	Tila
1	Kuopio - Iisalmi	Viedään Keskon kuljetus Iisalmeen	Käytössä
2	Iltasanomat	Iltasanomien jakelua Kuopiossa	Käytössä
3	Etelän auto	Kuopiosta Helsinkiin lehtien vieni lentokentälle	Käytössä
4	Kuopio KP2	Kantopiiri 2 jakelu Kuopiossa	Käytössä
5	Kuopio KP3	Kantopiiri 3 jakelu Kuopiossa	Käytössä

© Jullie, 2014

KUVA 13. Reittien ylläpito

Reittien ylläpitosivu sisältää reitit listattuna sekä ylläpitoon liittyviä toimintoja. Listan yläpuolella näkyvillä toimintoja voi tehdä yleisiä reitteihin liittyviä toimenpiteitä. Järjestä reitit -painike avaa dialogi-ikkunan, jolla voi muuttaa reittien järjestyksen, jonka mukaan ne listautuvat työjärjestysmoduulissa (kuvat 7 ja 8). Listan oikealla reunalla olevista painikkeista avautuu reittikohtainen dialogi-ikkuna, josta voi muokata reitin tietoja.

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Ylläpito / Keikat

Keikat

Luo keikka Poista keikka Edellinen lokakuu 2014 Seuraava Näytä kaikki Näytä poistetut

ID	Nimi	Kuvaus	Päiväys	Tunnit	Tila
3	Muutto KPO-VRK	Muutto Kuopiosta Varkauteen	29.10.2014 07:00	5.00	Käytössä
2	Yritysmuutto	Kuopiossa yritysmuutto.	28.10.2014 14:26	3.00	Käytössä
1	Muutto Kuopio - Rovaniemi	Tilattu muutto Rovaniemelle. Hinta 900 €.	25.10.2014 10:00	18.00	Käytössä

© Jullie, 2014

KUVA 14. Keikkojen ylläpito

Keikkojen muokkausnäkyminen mukaillee reitien muokkausnäkyminen. Näkyminen sisältää samanlaiset toiminnot pieniä poikkeuksia lukuunottamatta. Keikkoja selataan kuukausi kerrallaan listan yläpuolella olevilla painikkeilla. Keikkoja pääsee myös muokkaamaan listan oikealla puolella olevista painikkeista.

RouteWork Etusivu Työjärjestys Tuntilista Ylläpito Kirjautuneena Teppo

Etusivu / Ylläpito / Käyttäjät

Käyttäjät

Luo käyttäjä Poista käyttäjä

ID	Etunimi	Sukunimi	Puhelin	Email	Kirjautunut	Näkyvillä	Käytössä	Käyttäjäoikeudet	Salasana
5	Anna	Nieminen	0449991234	anna.nieminen@gmail.com		Kyllä	Kyllä	Näytä	Palauta
4	Antti	Virtanen	0401112222	antti.virtanen@me.com		Kyllä	Kyllä	Näytä	Palauta
2	Matti	Meikalainen	04001234123	matti.meikalainen@gmail.com		Kyllä	Kyllä	Näytä	Palauta
6	Mikko	Mallikas	0453334444	mikko.mallikas@mail.ru		Kyllä	Kyllä	Näytä	Palauta
3	Teppo	Testihenkilö	05001234123	teppo.testihenkilo@gmail.com	25.10.2014 15:52	Kyllä	Kyllä	Näytä	Palauta

© Jullie, 2014

KUVA 15. Käyttäjien ylläpito

Kuva 15 näyttää käyttäjien ylläpitosivun. Tällä sivulla voi lisätä tai poistaa käyttäjiä, sekä määrittää käyttäjäkohtaisia asetuksia kuten käyttäjäoikeuksia. Jos käyttäjä unohtaa salasanan, voi tältä sivulta sen tarvittaessa palauttaa. Käyttäjän voi myös väliaikaisesti poistaa käytöstä sekä piilottaa työjärjestysnäkymin.

9.2 Mobiililaite

Mobiililaitteelle käyttöliittymän tuottaminen asettaa koko rajoituksia. Tämän vuoksi, kuten aikaisemmassa kappaleessa todettiin, ylläpito-moduuli ei ole käytössä mobiililaitteilla. Jos taulutietokoneella näytön todellinen leveys on yli 1200 pikseliä, niin ylläpitotoiminnot ovat käytettävissä, jos käyttöoikeudet sen sallivat. Bootstrap-alustan ansiosta mobiililaitteen käyttöliittymä on lähes samanlainen tavallisen käyttöliittymän kanssa, mutta ne skaalautunut eri tavalla.

Kuten taulukosta 20 käy ilmi, erilaisia vaihtoehtoja näytön leveydelle mobiilialustalla on kolme kappaletta, mutta seuraavat kuvat on otettu, kun leveys on pienin mahdollinen. Muut leveydet sisältävät samat näkyvät komponentit, mutta eri tavalla skaalautuneina.

The image shows a mobile login interface for 'RouteWork'. It features a blue header with the 'RouteWork' logo. Below the header, there is a section titled 'Käyttäjätunnus' with a text input field. Underneath is a section titled 'Salasana' with another text input field. A checkbox labeled 'Muista minut' is located below the password field. At the bottom of the form is a blue button labeled 'Kirjaudu'.

KUVA 16. Mobiilikirjautumissivu

The image shows a mobile home page for 'RouteWork'. It has a blue header with the 'RouteWork' logo and a hamburger menu icon. Below the header is a grey navigation bar with the text 'Etusivu'. The main content area is titled 'Ilmoitukset' (Announcements) in large blue text. There are three announcement items, each with a vertical line on the left side. The first item is 'Tänään tehdään opinnäytetyötä.' with a timestamp '– 25.10.2014 16:04' and the author 'Matti Meikäläinen'. The second item is 'Tänään on ohjelmointipäivä.' with a timestamp '– 16.10.2014 16:53' and the author 'Matti Meikäläinen'. The third item is 'Auton huolto' with a timestamp '– 08.10.2014 16:57' and the author 'Teppo Testihenkilö'. At the bottom right, there is a small copyright notice: '© Julle, 2014'.

KUVA 17. Mobiilietusivu

Kuvat 16 ja 17 ovat mobiiliversioita kirjautumis- ja etusivusta. Ne ovat lähes täysin identtisiä tietokoneversion kanssa. Mobiilialustalla yläpalkin valikko on piilotettu napin taakse.

RouteWork

Etusivu / Työjärjestys

Omat työt Kaikki työt

Omat työt 43 / 2014

maanantai 20.10
Etelän auto

tiistai 21.10
Etelän auto

keskiviikko 22.10
Etelän auto

torstai 23.10

perjantai 24.10

lauantai 25.10
Kuopio KP2, Kuopio KP3, [Muutto Kuopio - Rovaniemi](#)

KUVA 18. Omat työt mobiilissa

Työjärjestysten selaaminen onnistuu myös mobiilissa. Käyttöliittymä on skaalautunut näytön koon mukaan, kuten kuvista 18 ja 19 näkyy. Poikkeuksena kuvasta 19 voidaan huomata, että kaikkien henkilöiden töitä ei voida selata viikko kerrallaan, vaan niitä selataan päivä kerrallaan, koska mobiilissa näytön leveys ei ole riittävän suuri.

Kuvassa 20 näytetään tuntiastianäkymä mobiilialustalla. Se sisältää samat toiminnot kuin tietokoneen näkymä. Tuntilistaan voi tehdä merkintöjä ja sen voi merkitä valmiiksi. Tulostustoiminto jätettiin myös näkyville, sillä se todellisuudessa avaa PDF-tiedoston, jonka voi halutessaan tallentaa puhelimelle tai taulutietokoneelle.

RouteWork

Etusivu / Työjärjestys

Omat työt Kaikki työt

Kaikki työt 25.10.2014

Reitti	Henkilöt
Kuopio - Iisalmi	
Iltasanomat	Matti
Etelän auto	Mikko
Kuopio KP2	Teppo
Kuopio KP3	Teppo
Muutto Kuopio - Rovaniemi	Teppo

Edellinen **Tänään** Seuraava

KUVA 19. Kaikkien työt mobiilissa.

RouteWork

Etusivu / Tuntilista

Oma tuntilista Kesken

Tulosta

Tuntilista lokakuu 2014

Päivä	Työ	Tunnit
1	ke	
2	to	
3	pe	
4	la	
5	su	
6	ma	
7	ti	
8	ke	

KUVA 20. Tuntilista mobiilissa

Reitti	Lisätieto	Tunnit
<input checked="" type="checkbox"/> Kuopio KP2	<input type="text"/>	<input type="text" value="3"/>
<input checked="" type="checkbox"/> Kuopio KP3	<input type="text"/>	<input type="text" value="3"/>
<input checked="" type="checkbox"/> Muutto Kuopio - Rovaniemi	<input type="text"/>	<input type="text" value="8"/>

Tallenna Peruuta

31 pe Etelän auto	4
Tunnit	53
Sunnuntai	6
Yhteensä	59

KUVA 21. Tuntilistaan merkintä mobiilissa

RouteWork

Etusivu / Profiili

Profiili

ID

Käyttäjänimi

Etunimi

✓

Sukunimi

✓

Puhelin

✓

KUVA 22. Profiili mobiilissa

Kuva 21 sisältää kuvan dialogi-ikkunasta, jossa tehdään tuntilistaan merkintä. Se on muuten samanlainen kuin isommilla näytön leveyksillä, ainoastaan kapeampi. Kuvassa 22 käyttäjä voi muokata profiiliansa mobiilialustalla. Näkymä mukailee myös tietokoneella näkyvää käyttöliittymää.

10 YHTEENVETO

Tavoitteena oli luoda opinnäytetyön tilaajalle, Kuljetukset ja Vuokraus Timo Pölläselle, toimiva järjestelmä, jonka avulla se voisi hallita töitään, työntekijöitään ja työtuntikirjanpitoa. Kehitystyö on ollut pitkä prosessi ja jatkuu vielä hieman tämän opinnäytetyön jälkeenkin. Kuitenkin tähän asti saatu tulos vaikuttaa erittäin hyvältä ja on testauksissa osoittautunut toimivaksi.

Työn tekeminen oli mielenkiintoista ja haastavaa. Tilaajan antamien vaatimuksien lisäksi oma työkokemus kuljetusyriyksessä auttoi paljon hahmottamaan jo alussa, mitä järjestelmään tarvitaan ja mitä ei. Kattavan suunnittelun jälkeen oli helppo siirtyä ohjelmointivaiheeseen.

Projektissa opin eniten juuri suunnitteluvaiheen aikana. Tämä oli ensimmäinen iso ohjelmistoprojektini, joten halusin panostaa siihen nimenomaan suunnittelun osalta. Ohjelmointivaiheen aikana sain paljon kokemusta käyttämästäni tekniikoista ja ohjelmointikielistä. Tämä kokemus auttoi minua jopa saamaan nykyisen työpaikkani.

Ohjelmointivaiheen aikana jokaiselle luokalle tehtiin omat funktiot tietokantakyselyjä varten. Myöhemmin kuitenkin keksin, että olisi voitu luoda vain geneeriset tietokantauseet, jotka muodostetaan luokan nimen ja sen muuttujien perusteella. Tässä vaiheessa luokat olivat kuitenkin jo tehty, joten niitä ei kannattanut lähteä muuttamaan. Tämä olisi kuitenkin ohjelmoinnin alkuvaiheessa säästänyt paljon aikaa.

Projektin loppupuolella huomasin, että olisi voinut hyödyntää muutamia uusia web-sovellustekniikoita, kuten AngularJS-koodikirjastoa, mutta siinä vaiheessa oli liian työlästä alkaa muuttaa mitään. AngularJS-koodikirjasto olisi nopeuttanut lomakkeiden käsittelyä käyttäjien, reittien ja keikkojen luonnin yhteydessä.

Työn tekeminen antoi paljon uutta tietoa web-ohjelmoinnista ja siihen liittyvistä lainalaisuuksista. Erityisesti jouduin panostamaan jQuery-koodikirjaston opetteluun, sillä se ei ollut niin tuttu. Työ jäi sellaiseen vaiheeseen, että siihen pitää vielä tehdä muutamia lisäyksiä, ennen kuin sen voi asentaa tuotantoon. Järjestelmä on myös kehitetty niin, että sitä on helppo lähteä jatkokehittämään ja lisäämään uusia moduuleita.

LÄHTEET JA TUOTETUT AINEISTOT

The PHP Group. Ei päiväystä. History of PHP. [WWW-dokumentti]. [viitattu 2014-10-26].

Saatavissa: <http://php.net/manual/en/history.php.php>

Database Friends, helmikuu 2014. History of MySQL. Helmikuu 2014. [WWW-dokumentti].

[viitattu 2014-10-26]. Saatavissa: <http://www.databasefriends.co/2014/02/history-of-mysql.html>

W3C. 2012-06-27. A Short History of JavaScript. [WWW-dokumentti]. [viitattu 2014-10-26].

Saatavissa: https://www.w3.org/community/webed/wiki/A_Short_History_of_JavaScript

YORK, Richard 2011. Beginning JavaScript and CSS Development with jQuery. [E-kirja].

[viitattu 2014-10-26]. Saatavissa: [http://books.google.fi/books?](http://books.google.fi/books?id=L9otyT4crSOC&pg=PP1&hl=fi&pg=PP1)

[id=L9otyT4crSOC&pg=PP1&hl=fi&pg=PP1](http://books.google.fi/books?id=L9otyT4crSOC&pg=PP1&hl=fi&pg=PP1)

OTTO, Mark 2012. Building Twitter Bootstrap. A List Apart. [viitattu 2014-10-26].

Saatavissa: <http://alistapart.com/article/building-twitter-bootstrap>

RESIG, John 2007. jQuery UI: Interactions and Widgets. jQuery Foundation. [WWW-artikkeli].

[viitattu 2014-10-26]. Saatavissa: <http://blog.jquery.com/2007/09/17/jquery-ui-interactions-and-widgets/>

MCMILLIAN, Robert 2005. After controversy, Torvalds begins work on "git". PC World.

[WWW-artikkeli]. [viitattu 2014-10-26]. Saatavissa: http://www.pcworld.idg.com.au/article/129776/after_controversy_torvalds_begins_work_git/