

LapSnapper -internet-tulospalvelu

LapSnapper

Ville Niskavaara

Opinnäytetyö
Tekniikan ja liikenteen ala
Sähkötekniikan koulutus
Insinööri (AMK)

2014

TIIVISTELMÄ

LAPIN AMMATTIKORKEAKOULU, Tekniikka

Koulutusohjelma:	Sähkötekniikan koulutusohjelma
Opinnäytetyön tekijä:	Ville Niskavaara
Opinnäytetyön nimi:	LapSnapper -internet-tulospalvelu
Sivuja:	24
Päiväys:	30.10.2014
Opinnäytetyön ohjaaja:	Tapani Ruokanen
Opinnäytetyön tarkoitus oli suunnitella ja toteuttaa LapSnapper-ajanottojärjestelmää tukeva tietokanta, web-käyttöliittymä, ja palvelinohjelmisto, joiden tarkoitus on ottaa vastaan ja säilöä LapSnapper-järjestelmän lähettämää ajanottotietoa sekä toimia käyttöliittymänä tuotteen loppukäyttäjälle ja ylläpitäjälle ajanottotietoja tarkasteltaessa ja muokatessa. Toteutettavan web-palvelun päätarkoitus on siis mahdollistaa LapSnapperin käyttäjien omien suoritusten tarkasteleminen internetissä. Toteutettava palvelinohjelmisto vastaanottaa LapSnapperin keräämää aikatietoa ja tallentaa sitä tietokantaan. Tietokantaan tallennettuja aikatietoja voidaan tarkastella asiaa varten toteutettavan käyttöliittymän kautta. Käyttöliittymä mahdollistaa myös tietokantaan kerätyn tiedon muokkaamisen. Tietokanta on suunniteltu PostgreSQL-tietokantajärjestelmää käyttäen, ja sinne tietoa syöttävä palvelinohjelma on toteutettu PHP-ohjelmointikielellä, kuten myös tietokantakäyttöliittymä. Tietokanta, palvelinohjelma ja käyttöliittymä toimivat Ubuntu-palvelimella. Lopputuloksena syntyi kolmesta osasta muodostuva joustava palvelukokonaisuus, joka toteuttaa tulospalvelulle asetetut vaatimukset. Palvelun osat on sijoitettavissa yhteen paikkaan, tai hajautettavissa kahdelle tai kolmellekin palvelimelle tarpeesta riippuen.	
Asiasanat: PHP, SQL, WWW, Ubuntu.	

ABSTRACT

LAPLAND UNIVERSITY OF APPLIED SCIENCES, Technology

Degree programme:	Degree of Electrical Engineering
Author	Ville Niskavaara
Thesis title:	LapSnapper internet result service
Pages:	24
Date:	30 October 2014
Thesis instructor:	Tapani Ruokanen
The goal of the thesis was to design and implement a database with a web interface and a server application capable of receiving timing information from LapSnapper sports timing system. The web interface is to be used by an end user and administrator to browse and modify timing data stored in the database. So the purpose of this web service is to provide the means for the users of LapSnapper to monitor their performance through the internet. The server application receives and stores timing data gathered and sent by LapSnapper into the database. The data stored in the database can be accessed through the web interface which provides the means to browse and modify the data stored in the database. The database designed for the thesis was created using PostgreSQL. The server application used to receive timing data as well as the web interface were created using PHP language. The database, the server application and the web interface are run on Ubuntu server. The service produced as a result of thesis met the requirements set for a result service. In addition it is also flexible as it is possible to be installed decentralized on up to three different servers if necessary.	
Keywords: PHP, SQL, WWW, Ubuntu.	

SISÄLLYS

TIIVISTELMÄ	3
ABSTRACT	4
SISÄLLYS	5
1. JOHDANTO	6
2. LAPSNAPPER	7
2.1. Järjestelmän käyttöönotto.....	8
2.2. Ajanotto kilpailutilanteessa.....	8
2.3. Opinnäytetyön rooli LapSnapperissa	8
3. KÄYTETYT JÄRJESTELMÄT JA OHJELMOINTIKIELET	10
4. TOTEUTETTAVAT OSA-ALUEET.....	13
4.1. Tietokanta.....	13
4.2. Palvelinsovellus	15
4.3. Web-käyttöliittymä	17
5. TESTAUS	22
6. POHDINTA	23
6.1. Jatkokehitys.....	23
LÄHTEET	24

1. JOHDANTO

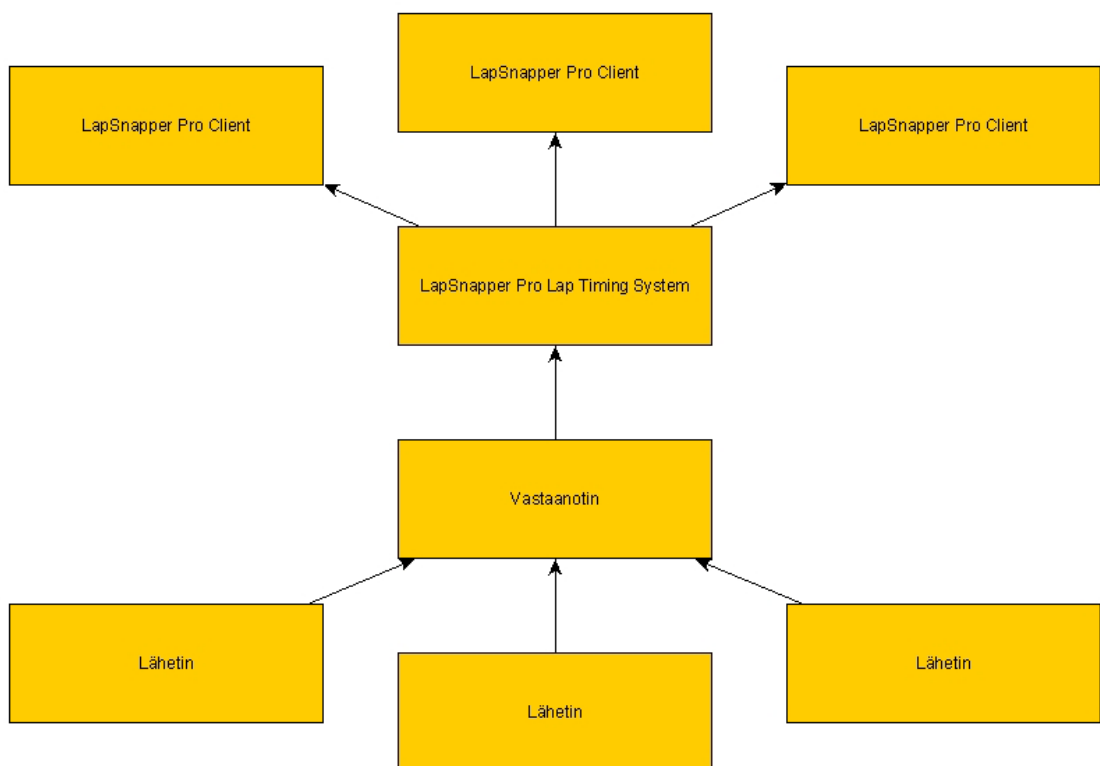
LapSnapper on langattomaan ajanottoon suunniteltu järjestelmä, jonka kohdeyleisö ovat rataurheilun harrastajat. Järjestelmä soveltuu käytettäväksi niin pienikokoisista autoradoista aina karting-radoille asti. Sen avulla voidaan järjestää kilpailun seuranta paikallisesti, eli kilpailua seuraavat katsojat voivat seurata tuloksia myös omilta älylaitteiltaan järjestelmään kuuluvan asiakassovelluksen avulla langattoman lähiverkon kantaman sisällä.

Internetiä hyödyntävä tulospalvelu on järjestelmässä looginen askel tallennetun aikatiedon tuomiseksi tuotteen käyttäjien saataville myös kilpailutapahtumien ulkopuolella. Opinnäytetyön tarkoitus on luoda tällainen palvelu. Työn tuloksena syntyvää yksinkertaista palvelua voidaan jatkokehittää tarpeellisten ominaisuuksien lisäämiseksi ja laajentamiseksi myöhemmin.

Käytännössä toteutettava opinnäytetyö sisältää internet-palvelimelle sijoitetun tietokantapalvelun, joka ottaa tietoa vastaan LapSnapper-järjestelmästä. Lisäksi palvelu sisältää julkisesti internetissä näkyvän web-sivuston tietokantaan tallennetun tiedon tarkastelemista varten. Kyseessä on siis yksinkertainen internet-tulospalvelu, jonne LapSnapperin keräämää tietoa lähetetään ja josta se on sitten tarkasteltavissa.

2. LAPSNAPPER

LapSnapper on langaton tuotekokonaisuus, joka koostuu kuvassa 1 esitetyllä tavalla esimerkiksi karting-autoon liitettävästä yksilöllisen tunnisteen omaavasta lähetinlaitteesta, äylaitteeseen usb-liittimellä kytkettävästä vastaanottimesta, äylaitteeseen asennettavasta LapSnapper Pro Lap Timing -sovelluksesta ja mahdollisista LapSnapper Pro Client -sovelluksista, joiden avulla kilpailuja seuraavat voivat seurata tuloksia omilta äylaitteiltaan langattoman lähiverkon kantaman alueella. Näiden lisäksi tarvitaan magneettinen maaliviivakomponentti.



Kuva 1. LapSnapper tuotekokonaisuuden osat.

Kokonaisuuteen voidaan lisätä lähettimiä käyttäjien määrän mukaan. Myös maalilinjoja voidaan lisätä aina kolmeen kappaleeseen asti, jolloin voidaan seurata kierrosaikojen lisäksi myös sektoriaikoja.

2.1. Järjestelmän käyttöönotto

LapSnapper-ajanottojärjestelmä sisältää vain pienen määrän fyysisiä komponentteja ja ohjelmistosovelluksia, joten sen käyttöönotto on yksinkertainen ja ajallisesti nopea prosessi. Lisäksi järjestelmän asennusta ja käyttöä helpottaa sen langattomuus.

Ajanottojärjestelmän käyttöönottamiseksi, asennetaan käytettävään rataan yksi tai useampi magneettinen maaliviiva (yksi kierrosaikojen mittaamiseksi, kaksi tai kolme sektoriaikojä varten). Lähetinlaitteet kiinnitetään ajokkeihin. Vastaanotin kytketään älylaitteeseen, johon myös asennetaan LapSnapper Pro Lap Timing -sovellus, jolla ajanottotapahtumaa hallinnoidaan. Lisäksi tapahtuman seuraajien on mahdollista asentaa älylaitteisiinsa LapSnapper Pro Client -sovellus, jonka kautta tuloksia pystytään seuraamaan langattoman verkon välityksellä. (LapSnapperin [www-sivut](#) 2014, hakupäivä 19.8.2014)

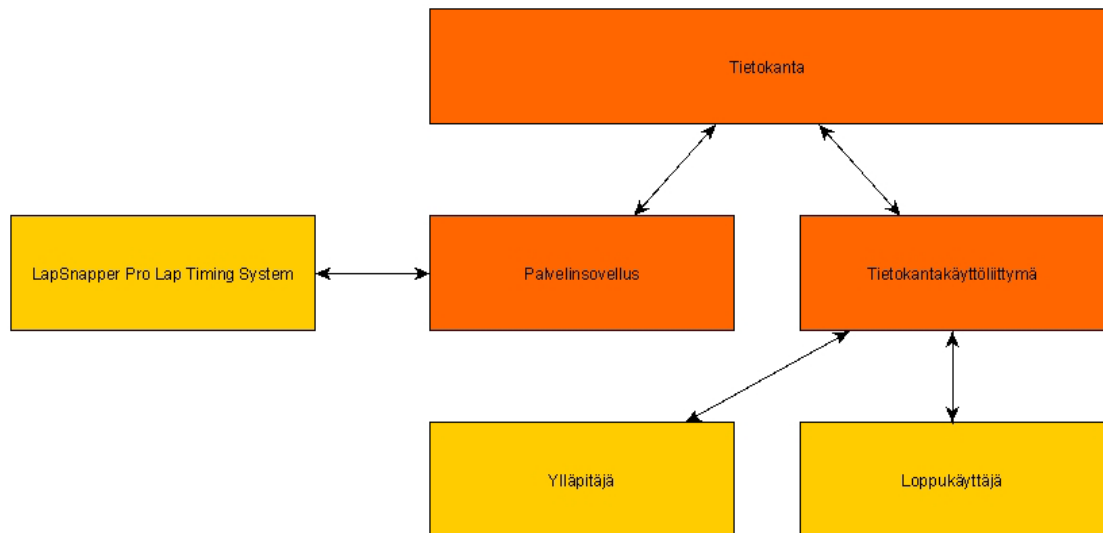
2.2. Ajanotto kilpailutilanteessa

Kun järjestelmän sisältämät komponentit on asennettu paikoilleen ja sovellukset asennettu älylaitteisiin voidaan kilpailutapahtuma luoda LapSnapper Pro Timing -sovelluksella, joka toimii kilpailutapahtuman hallintakäyttöliittymänä.

Käytännön kilpailutilanteessa esimerkiksi radio-ohjattavaan autoon liitetty lähetin lähettää magneettisen maaliviivan ylitettyään tiedon vastaanottimelle, joka puolestaan siirtää tietoa LapSnapper Pro Lap Timing -sovellukselle älylaitteeseen, johon kytkettynä se on. LapSnapper Pro Lap Timing -ohjelmisto pystyy näyttämään ajanmittaustulokset reaaliajassa. Sovelluksen keräämät tulokset välittyvät langattoman verkon välityksellä mahdollisiin LapSnapper Pro Client -sovelluksiin. (LapSnapperin [www-sivut](#) 2014, hakupäivä 19.8.2014)

2.3. Opinnäytetyön rooli LapSnapperissa

Opinnäytetyön tarkoitus oli suunnitella ja toteuttaa internet-palvelu LapSnapper Pro Lap Timing -sovelluksen keräämää ajanmittaustietoa varten, jotta lähetinkohtaiset ajanmittaustiedot olisivat kilpailuihin osallistuneiden tarkasteltavissa myös tapahtuman jälkeen internetin välityksellä.



Kuva 2. Opinnäytetyön osa-alueet oranssilla.

Opinnäytetyö koostuu kolmesta komponentista: tietokanta, palvelinsovellus ja web-käyttöliittymä. Kuvan 2 mukaisesti LapSnapper Pro Lap Timing -sovellus on vuorovaikutuksessa opinnäytetyöhön kuuluvan palvelinsovelluksen kanssa ja näiden kahden välinen yhteys toimii rajapintana kahden kokonaisuuden välisessä kommunikaatiossa.

Loppukäyttäjä pystyy selaamaan tietokantaan tallennettuja lähetinkohtaisia aikatietoja ja muokkaamaan käyttäjätietojaan web-käyttöliittymän kautta. Web-käyttöliittymä sisältää myös ominaisuuksia, joilla ylläpitäjät voivat muokata tietokantaan tallennettuja aikatietoja, käyttäjätietoja sekä palvelun asetuksia.

3. KÄYTETTYT JÄRJESTELMÄT JA OHJELMOINTIKIELET

Työn koostuessa kahdesta ohjelmointia vaativasta osa-alueesta ja yhdestä tietokannasta, sekä palvelimesta, jolle valmis työ sijoitettiin, oli olennaista valita käytettävät ohjelmointikielet, järjestelmät ja työkalut tarkoituksiin sopiviksi.

Ohjelmointikielten lisäksi valittiin varsinaisen ohjelmoinnin tekemistä varten tekstinkäsittelyohjelma, tietokantajärjestelmä, sekä suunnittelukaavioiden piirtämiseen tarkoitettu työkalu. Edellä mainittujen lisäksi tarvittiin myös palvelinkäyttöjärjestelmä, joka soveltuisi työn tuloksena syntyvien sovellusten ajamiseen.

Palvelinsovellus

Palvelinsovelluksen osalta mietittiin aluksi ohjelmointikieleksi C, tai C++ -kieltä, koska palvelinsovellusten osalta ne tiedettiin etukäteen luotettaviksi vaihtoehtoina. Palvelinsovelluksen yksinkertaisuus kuitenkin mahdollisti myös PHP-kielen harkitsemisen vaihtoehtoina.

C ja C++ -kielten tapauksessa toteutettava sovellus olisi kääntämisen jälkeen käyttöön soveltuva vain tietynlaisessa palvelinkäyttöjärjestelmässä. Tämä tarkoittaa sitä, että eri palvelinkäyttöjärjestelmille (Windows, Linux-pohjaset jne.) sovellus olisi käännettävä erikseen ja niihin tulisi todennäköisesti tehdä käyttöjärjestelmästä riippuen muutoksia. Tämä teettäisi lisää työtä, jos palvelimen käyttöjärjestelmää syystä tai toisesta jouduttaisiin vaihtamaan.

PHP:n eduksi palvelinsovellusta suunniteltaessa muodostui sen toimivuus käyttöjärjestelmästä riippumattomammin, kuin C ja C++. PHP on oikeastaan komentosarjakieli, jota ei varsinaisesti tarvitse kääntää etukäteen, vaan sen kääntäminen tapahtuu palvelimella käyttäjän avatessa PHP-sivun internet-selaimella. Lisäksi PHP:ssä on olemassa tuki palvelinohjelmiston vaatimaa tietoliikenneportin kuuntelua varten, joka tosin löytyy myös C ja C++ -kielistä. PHP:n tietokantatuki on myös vaihtoehtoja parempi ja helpommin käyttöönotettavissa.

PHP:n yksinkertaisempi siirrettävyys käyttöjärjestelmien kesken vaikutti ratkaisevasti sen valitsemiseen palvelinohjelmiston toteutuskieleksi.

Web-käyttöliittymä

Web-käyttöliittymän toteutuskieleksi valittiin PHP. Perusteena valinnalle olivat muun muassa sen tietokantatuki ja PHP:n valitseminen palvelinsovelluksen ohjelmointikieleksi. Molempien osa-alueiden toteuttaminen samalla kielellä mahdollistaa yhteisten funktiokirjastojen käyttämisen. Tämä säästi työhön kulunutta aikaa.

Tietokantajärjestelmä

Tietokantajärjestelmistä mietittiin MySQL ja PostgreSQL -järjestelmiä. Molemmista järjestelmistä oli käytännön kokemusta, mutta niiden eroavaisuuksista toteutettavan järjestelmän suhteen ei oikeastaan ollut tietoa. PostgreSQL:n valitseminen tietokantajärjestelmään pohjautuikin tuoreemmassa muistissa olevaan käyttökokemukseen.

Palvelinkäyttöjärjestelmä

Työn toteutuksessa yksi tärkeimmistä perusasioista on palvelin ja ennen kaikkea palvelinkäyttöjärjestelmä. Vaikka jokainen kolmesta osa-alueesta onkin suunniteltu toimivaksi sekä samalla palvelimella että jokainen käyttöjärjestelmästä riippumatta erikseen, päätettiin työ toteuttaa käyttäen yhtä palvelinta, jolla kaikki kolme osa-aluetta toimivat.

Palvelinkäyttöjärjestelmäksi valittiin Ubuntu Linux 12.04 käyttöjärjestelmä. Perusteluina Linux-pohjaisten järjestelmien soveltuvuus palvelinkäyttöön ja Ubuntu-yhteisön tarjoama laaja tuki. Windows-käyttöjärjestelmät jätettiin pois harkinnasta niiden heikon muokattavuuden ja käyttökustannusten takia. Vaikka Ubuntu Linux onkin ilmainen käyttöjärjestelmä, on siitä saatavilla varta vasten palvelinkäyttöön suunniteltuja versioita. Lisäksi ylimääräisten ominaisuuksien (kuten esimerkiksi graafisen käyttöliittymän) karsiminen pois on helpompaa Linux-käyttöjärjestelmissä.

Käytetyt ohjelmistot

Kaikki työhön liittyvä PHP-koodi kirjoitettiin textpad-tekstinkäsittelyohjelman ilmaisversiolla (TextPadin [www-sivut 2014](http://www.sivut2014.com), hakupäivä 26.9.2014). Tietokannat luotiin ja muokattiin palvelimelle asennetun PhpPgAdmin-tietokantahallintaohjelman avulla (PhpPgAdminin [www-sivut 2014](http://www.sivut2014.com), hakupäivä 26.9.2014). Kaaviot piirrettiin ilmaisella

yED Graph Editor -ohjelmalla (YED Graph Editorin [www-sivut](http://www.yworks.com) 2014, hakupäivä 26.9.2014).

4. TOTEUTETTAVAT OSA-ALUEET

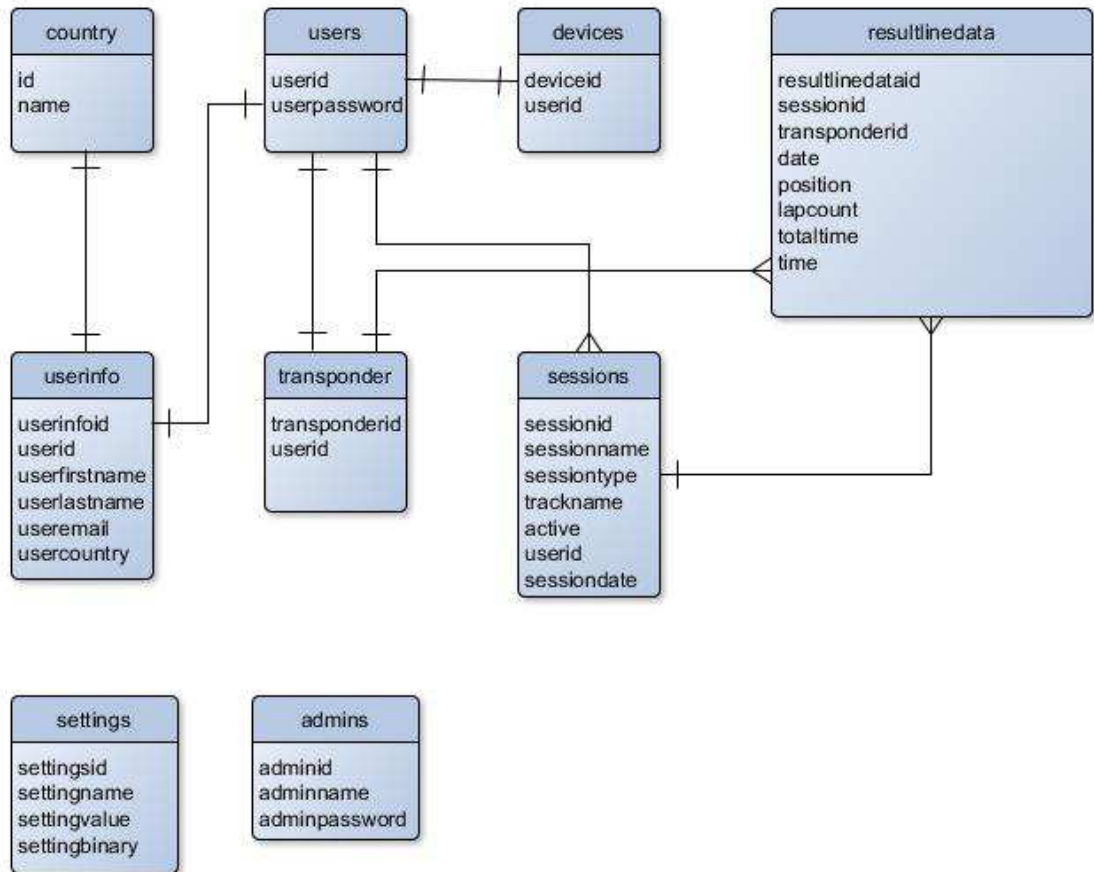
Opinnäytetyö voidaan jaotella kuvan 2 mukaisesti kolmeen erilliseen kokonaisuuteen. Nämä kokonaisuudet ovat tietokanta, palvelinsovellus ja web-käyttöliittymä. Mainituista osa-alueista keskeisimmässä osassa on tietokanta, johon palvelinsovellus ja web-käyttöliittymä toimivat rajapintoina.

Palvelinsovellus ottaa vastaan LapSnapper Pro Lap Timing -sovelluksen lähettämiä viestejä, ja tekee tietokantaa koskevia hakuja ja muutoksia vastaanotettujen viestien perusteella. Tämän jälkeen se lähettää vastausviestin, joka sisältää mahdollisen virheilmoituksen ja mahdollisesti haetun tiedon tietokannasta.

Web-käyttöliittymä on nimensä mukaisesti loppukäyttäjälle suunniteltu rajapinta, jonka kautta pystytään muokkaamaan omaa tunnusta ja seuraamaan rekisteröidyn lähettimen ajanmittaustuloksia. Web-käyttöliittymä sisältää myös ylläpito-osion, jonka kautta ylläpitäjäkäyttäjät voivat muokata tietokannan sisältöä.

4.1. Tietokanta

Työn keskeisenä osana on PostgreSQL tietokanta. Tässä tapauksessa tietokanta on asennettu Ubuntu Linux -palvelimelle, mutta PostgreSQL tietokanta voidaan asentaa Linux-käyttöjärjestelmien lisäksi myös BSD, Macintosh, Solaris ja Windows -käyttöjärjestelmiin (PostgreSQL:n [www-sivut](http://www.postgresql.org/) 2014, hakupäivä 6.8.2014). Vaikka tietokanta opinnäytetyön tapauksessa sijaitseekin samalla palvelimella, kuin muut osa-alueet (palvelinsovellus ja web-käyttöliittymä), voidaan se asentaa myös erilliselle palvelimelle.



Kuva 3. Tietokannan luokkakaavio.

Käytettävä tietokanta sisältää kuvan 3 mukaisesti yhdeksän taulua, joihin kerättävää tietoa jaotellaan lukuun ottamatta valtiot ja niiden lyhenteet sisältävää country-taulua ja palvelun asetukset sisältävää settings-taulua, joiden sisältämä rivimäärä pysyy mahdollisia manuaalisesti tehtäviä sisältöpäivityksiä lukuun ottamatta vakiona.

Users-taulu sisältää nimensä mukaisesti web-palveluun rekisteröityneiden käyttäjien tiedot. Koska rekisteröitynyt käyttäjä voi rekisteröityä vain lähettimen avulla (erikoistapauksia lukuun ottamatta), toimii käyttäjätunnuksena lähettimen yksilöllinen tunniste, joka tallennetaan transponder-tauluun.

Transponder-taulu sisältää web-palveluun rekisteröityneille käyttäjille merkittyjen lähettimien yksilölliset tunnistet, ja viitteet siihen users-tauluun merkittyyn käyttäjään, jolle lähetin on rekisteröity.

Devices-taulu sisältää palvelinsovellukseen yhteydessä olevan laitteen virtuaalisen tunnisteen, joka luodaan uudelleen jokaisella sisäänkirjautumiskerralla erikseen. Virtuaalis-

ta tunnistetta käytetään tapahtumien (käytännössä kilpailut, joiden yhteydessä aikoja mitataan) luomisessa ja niihin liittyvien ajanottotietojen lähettämisessä palvelimelle. Se on tarpeellinen siksi, että palvelinsovellukseen yhteydessä olevia laitteita voi samaan aikaan olla useita ja jokainen laite on osattava erottaa toisistaan. Virtuaalitunniste tuhoetaan uloskirjautumisen yhteydessä.

Userinfo-taulu sisältää Web-palveluun rekisteröidyn käyttäjän tarkemmat käyttäjätiedot. Henkilötietojen lisäksi se sisältää viitteen siihen users-aulun käyttäjään, jolle tiedot kuuluvat, sekä viitteen käyttäjän valitsemaan country-aulun maahan.

Country-taulu sisältää luettelon maailman valtioista, sekä niiden kaksikirjaimisen lyhenteen. Tätä taulua käytetään web-palvelun käyttäjätietojen päivittämisen yhteydessä. Taulun sisältö on haettu Countries List –sivustolta (Countries Listsin [www-sivut](http://www.countries-lists.com) 2014, hakupäivä 26.4.2014).

Resultlinedata-taulu sisältää kaiken web-palveluun tallennettavan ajanottotiedon. Tieto sisältää viitteen siihen lähettimeen, jonka aikatieto on kyseessä, sekä siihen tapahtumaan, johon aikatieto liittyy. Taulun tietojen perusteella luodaan web-palvelun kautta näytettävät tilannetiedot tapahtumien kulusta.

Sessions-taulu sisältää tapahtumien tietoja. Muun muassa tapahtuman nimen ja radan nimen. Tapahtuman tunnistetta käytetään yhdistämään resultlinedata-aulun aikatieidot niihin tapahtumiin, joihin aikatieidot liittyvät.

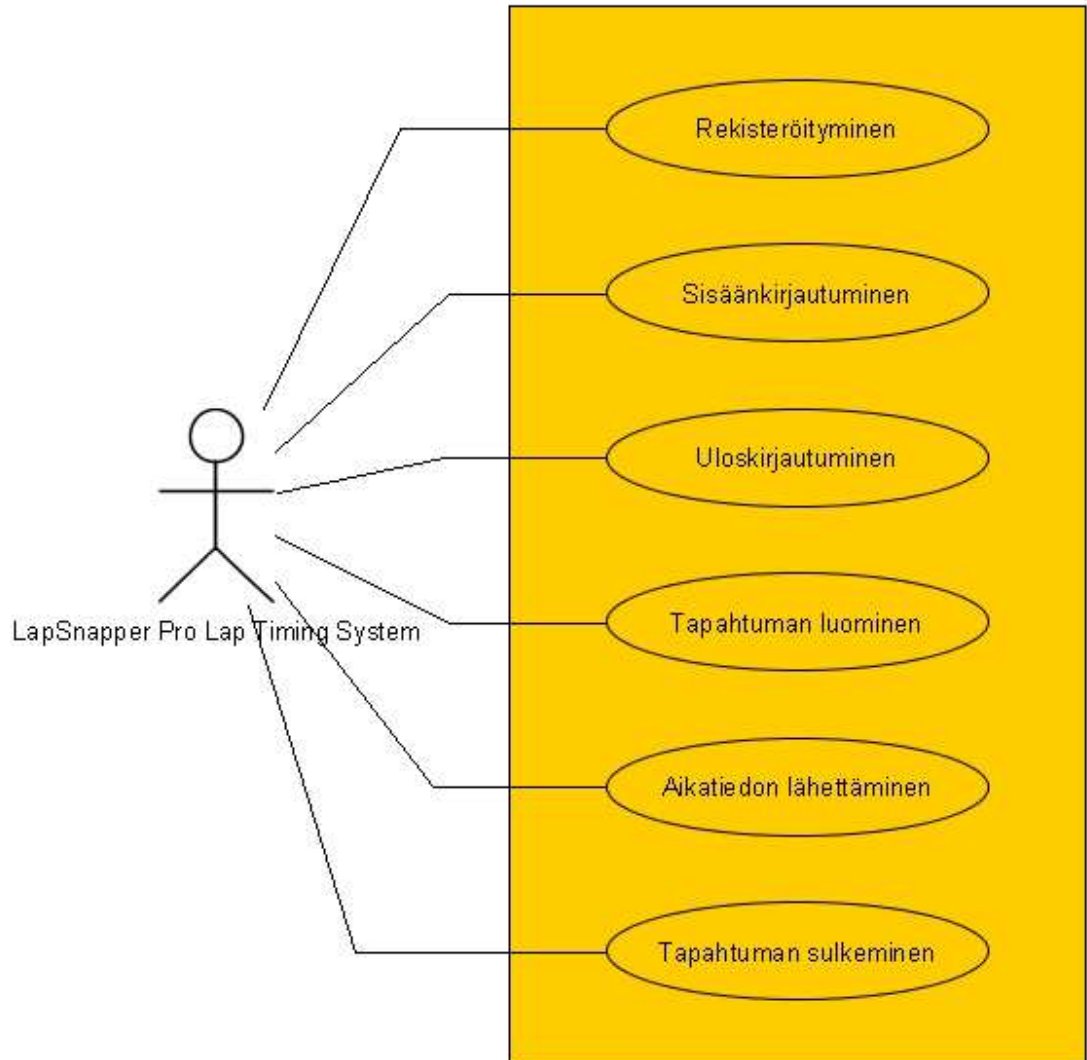
Admins-taulu sisältää web-palvelun ylläpitäjien kirjautumistiedot, sillä ei ole viitteitä muihin tauluihin.

Settings-taulu sisältää web-palveluun liittyviä asetuksia, kuten esimerkiksi automaattisen vanhojen aikatieidojen poistamisen.

4.2. Palvelinsovellus

Palvelinsovelluksen tarkoitus on toimia rajapintana tietokannan, ja LapSnapper Pro Lap Timing -sovelluksen välissä. Kommunikaatio kahden komponentin välillä tapahtuu TCP/IP protokollaa käyttäen. LapSnapper Pro Lap Timing -sovellus toimii aktiivisena

osapuolena, eli se lähettää palvelinsovellukselle viestin, johon palvelinsovellus lähettää vastausviestin, joka sisältää vähintään virhekoodin mahdollisia virheitä ja puutteita varten, sekä mahdollisesti lisätietoja, kuten esimerkiksi virtuaalisen laitetunnisteen.



Kuva 4. Käyttötapauskaavio palvelinsovelluksen toiminnallisuuksista.

Palvelinsovelluksen sisältämät toiminnallisuudet on esitelty kuvan 4 käyttötapauskaaviossa. Kuten kuvasta ilmenee, sovellus on tarkoitettu vain ja ainoastaan LapSnapper Pro Lap Timing -sovelluksen käyttöön, eli se on ainoastaan ohjelmistorajapinta toisen ohjelmakokonaisuuden ja tietokannan välillä.

Rekisteröitymisviestissä LapSnapper Pro Lap Timing -sovellus lähettää yhden lähettimen rekisteröintitiedot web-palveluun. Rekisteröityminen mahdollistaa tapahtumien luomisen ja niihin liittyvien aikatietojen lähettämisen lisäksi web-palvelun käytön.

Sisäänkirjautuessa palvelinsovellukselle lähetetään kirjautumisviesti, joka sisältää lähettimen tunnusteen ja lähettimelle rekisteröidyn salasanan. Onnistuneen sisäänkirjautumisen jälkeen palvelinsovellus lähettää vastausviestissä virtuaalisen laitettunnusteen, jota tarvitaan tapahtumia koskevien viestien lähettämiseen.

Uloskirjautumisviestin vastaanotettuaan palvelinsovellus kirjaa käyttäjän ulos, ja tuhoaa virtuaalisen laitettunnusteen.

Tapahtuman luominen tapahtuu niin, että LapSnapper Pro Lap Timing -sovellus lähettää palvelinsovellukselle luotavan tapahtuman tiedot (nimi, päivämäärä, jne). Tapahtuman luominen edellyttää sisäänkirjautumista ja voimassaolevaa virtuaalista laitettunnistetta.

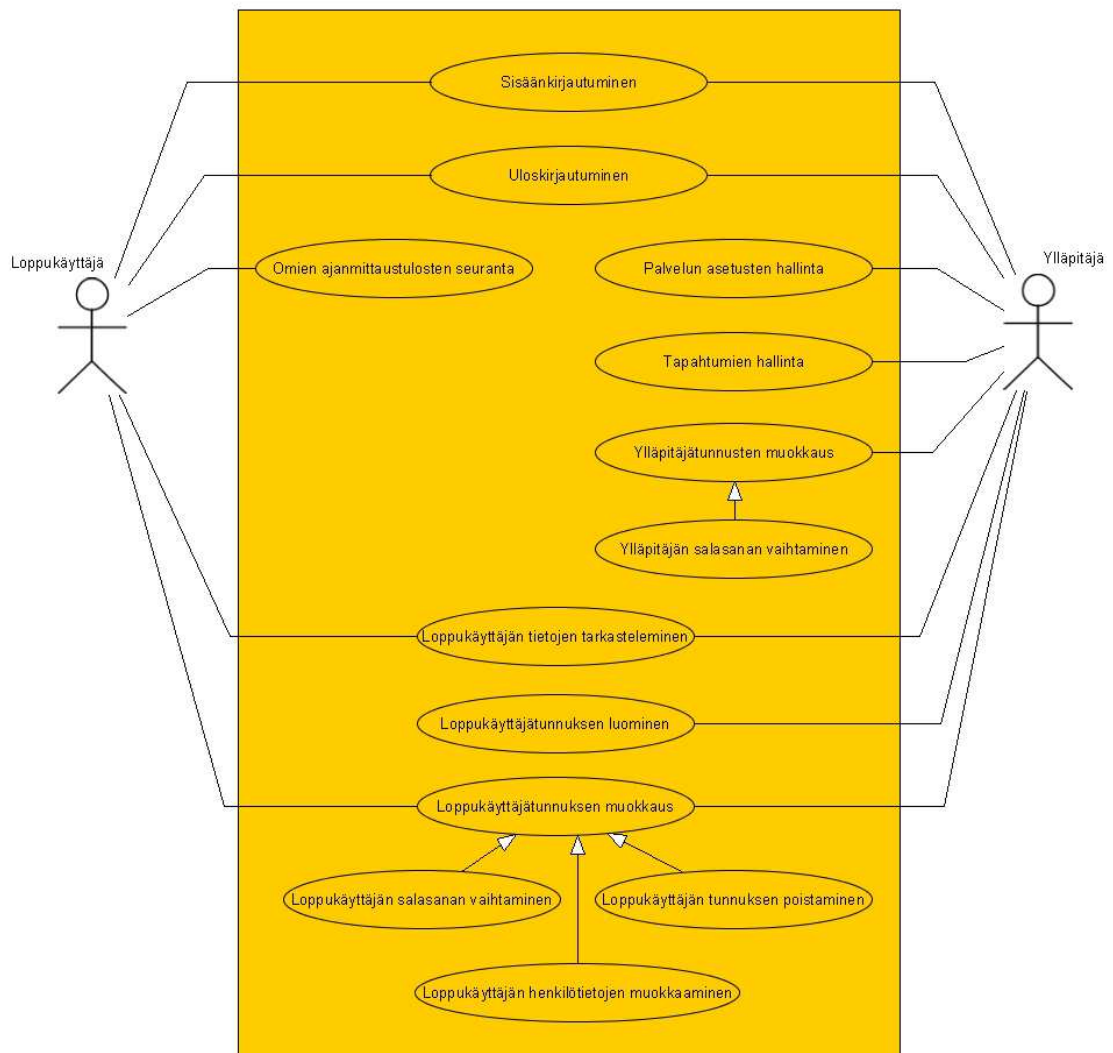
Aikatiedon lähettäminen tulospalveluun tapahtuu LapSnapper Pro Lap Timing -sovelluksen lähettäessä palvelinsovellukselle aikatiedot sisältävän viestin. Aikatiedon hyväksyminen palvelinsovelluksen toimesta edellyttää voimassaolevaa virtuaalista laitettunnistetta ja tapahtuman aktiivista tilaa.

Tapahtuman sulkemisviestin vastaanotettuaan palvelinsovellus sulkee tapahtuman, joten siihen liittyvää aikatietoa ei enää oteta vastaan. Toiminto edellyttää voimassaolevaa virtuaalista laitettunnistetta.

4.3. Web-käyttöliittymä

Web-käyttöliittymä käsittää suurimman osan toteutetusta työstä. Se sisältää erilliset käyttöliittymät ominaisuuksineen rekisteröidyille loppukäyttäjille ja ylläpitokäyttäjille. Loppukäyttäjälle suunnatut toiminnallisuudet käsittävät omien aikatietojen seurannan niiden tapahtumien osalta, joihin rekisteröidyllä lähettimellä on osallistuttu. Loppukäyttäjä voi myös halutessaan lisätä palveluun tarkempia tietoja itsestään. Ylläpitokäyttäjän käyttöliittymä on suunniteltu tarjoamaan toimintoja rekisteröityneiden käyttäjien, tapah-

tumien, sekä web-palvelun asetusten hallintaan. Web-käyttöliittymä tarvitsee toimiakseen internet yhteyden ja selaimen.



Kuva 5. Web-käyttöliittymän käyttötapauskaavio.

Vaikka loppukäyttäjän ja ylläpitäjän käyttöliittymät ovat käytännössä kokonaan toisistaan erillään, niiden toiminnallisuuksilla on joissain tapauksissa täsmälleen samanlaiset vaikutukset. Kuten kuvasta 5 ilmenee, loppukäyttäjätunnuksen luominen on mahdollista ainoastaan ylläpitäjälle erikoistapauksessa, sillä normaalisti loppukäyttäjätunnuksen luominen tapahtuu LapSnapper Pro Lap Timing -sovelluksen kautta.

Istunnot

Web-käyttöliittymien toiminnallisuudet vaativat sisäänkirjautumisen. Sisäänkirjautumisen yhteydessä aloitetaan istunto, joka tarkoittaa käyttäjän tunnistetietojen tallentamista istuntomuuttujiin web-käyttöliittymän käytön ajaksi. Muuttujat tuhotaan uloskirjautumisen tai web-palvelimen asetuksista riippuvan aikarajan ylittymisen yhteydessä. Näitä muuttujia tarkastelemalla käyttöliittymän eri osa-alueet tunnistavat kirjautuneen käyttäjän. Mikäli web-käyttöliittymän toimintoja yritetään käyttää ilman kirjautumista, uudelleenohjataan selain sisäänkirjautumissivulle.

Sisäänkirjautumisessa web-käyttöliittymään syötetään käyttäjätunnus ja salasana. Loppukäyttäjän käyttäjätunnus on sama kuin hänen rekisteröimänsä lähettimen tunniste. Ylläpitäjän käyttäjätunnus ei ole yhteydessä lähettimiin, vaan se voidaan määrittää vapaammin ylläpitotunnusta luotaessa. Sisäänkirjautumisessa käyttäjäkohtaisia tietoja tallennetaan istuntomuuttujiin.

Uloskirjautuminen suoritetaan kirjautumissivulla olevan painikkeen kautta. Uloskirjautuminen tyhjentää istuntomuuttujat.

Omien ajanmittaustulosten seuranta on oleellisin loppukäyttäjälle suunnattu ominaisuus, joka mahdollistaa rekisteröidylle lähettimelle kirjattujen aikatietojen seurannan. Saatavilla on aikatiedot kaikista niistä tapahtumista, joihin lähettimellä on osallistuttu. Tämä sisältää myös muille kyseiseen tapahtumaan osallistuneille lähettimille kirjatut aikatiedot. Web-palvelu näyttää tapahtumakohtaisesti tiedot sen tyypistä riippuen:

Harjoitustulokset näytetään siinä järjestyksessä, että ensimmäiseksi merkitään eniten kierroksia ajanut. Saman kierrosmäärän ajaneista paremmaksi merkitään se, jonka kokonaisaika on pienempi. Tämä menetelmä on samanlainen kuin varsinaisessa kilpailussa. Lisäksi näytetään kaikki lähetinkohtaiset kierrosajat.

Aika-ajojen tuloksissa kilpailijat (eli lähettimet) listataan nopeimman kierrosajan mukaan. Lisäksi näytetään harjoituksen tapaan kaikki lähetinkohtaiset kierrosajat.

Varsinaisen kilpailun lopputulokset listataan kuten harjoituksissakin. Lopputulosten lisäksi listataan kilpailijat myös nopeimpien kierrosaikojen mukaan. Lopuksi näytetään myös lähetinkohtaiset kierrosajat.

Toiminnallisuudet **loppukäyttäjän tietojen tarkastelemiseen ja muokkaamiseen** ovat sekä loppukäyttäjällä itsellään, että palvelun ylläpitäjillä. Perustietojen, kuten nimen, sähköpostin ja kotimaan lisäksi voidaan myös salasana vaihtaa. Lisäksi koko rekisteröity käyttäjätunnus voidaan poistaa. Tämä ei kuitenkaan poista tietokantaan tallennettuja aikatietoja, jotka ovat merkitty rekisteröidylle lähettimelle.

Ylläpitokäyttäjällä on myös mahdollisuus **luoda loppukäyttäjätunnuksia**. Tämä on erikoistilanteita varten, sillä tavallisesti loppukäyttäjätunnus luodaan LapSnapper Pro Lap Timing -sovelluksen kautta.

Ylläpitäjän on mahdollista **muokata ylläpitäjätunnuksia** ja niiden salasanoja. Web-palvelussa ei varsinaisesti ole olemassa minkäänlaista yhtä pääkäyttäjää, vaan kaikille ylläpitäjille on mahdollista luoda lisää ylläpitäjätunnuksia, vaihtaa niiden salasanoja ja poistaa tunnuksia. Ainoa rajoitus on, että viimeistä olemassa olevaa ylläpitotunnusta ei voi poistaa, vaikka normaalitilanteessa ylläpitäjä voikin poistaa oman tunnuksensa. Ongelmatilanteessa tietokannan ylläpitäjä voi luoda ylläpitäjätunnuksen suoraan tietokantaan.

Tapahtumien hallinta -ominaisuus on ylläpitäjän työkalu, jolla voidaan muokata tietokantaan luotujen tapahtumien tietoja, tai poistaa tapahtumia. Lisäksi on mahdollista poistaa tietokannasta tapahtumia antamalla vuosina ja kuukausina aikamäärä, jota vanhemmat tapahtumat poistetaan.

Palvelun asetusten hallintaan kuuluu automaattinen vanhojen tapahtumien poistominaisuus. Automaattiselle poistolle määritellään kuukausimäärä, jota vanhemmaksi ilmoitetut tapahtumat poistetaan automaattisesti palvelimen ajastustoiminnon avulla. Ominaisuuden voi ottaa käyttöön tai poistaa käytöstä. Tämä on ainoa olemassa oleva asetus, mutta asetukset ovat laajennettavissa.

Web-palvelu käyttää automaattisten tapahtumien poistamisen toteuttamiseen Ubuntu-palvelimen ajastuspalvelua (cron). Ajastuspalvelu on asetettu ajamaan vuorokauden

vaihtuessa PHP-skripti, joka tarkastaa tietokannasta onko ajastettu vanhojen tapahtumien poisto otettu käyttöön. Jos ominaisuus on otettu käyttöön, ajetaan PHP-skriptissä SQL-kysely, joka poistaa tietokannasta kaikki ne tapahtumat, joihin merkitty päivämäärä on vanhempi kuin nykyinen päivämäärä, josta on poistettu automaattiseen poist ominaisuuteen merkitty aikamäärä. Ajastinpalvelun käyttäminen ei rajoita web-käyttöliittymän käyttöjärjestelmävaatimuksia, sillä ajastinpalveluita on olemassa myös muissa käyttöjärjestelmissä. Ajastinpalvelu tulee vain määrittää ajamaan automaattisen poistopalvelun sisältävä PHP-skripti.

5. TESTAUS

Opinnäytetyön lopputuloksena syntyneet sovellukset (palvelinsovellus, ja web-käyttöliittymä) testattiin ensin yksikkötestausmenetelmällä, jolloin keskityttiin testaamaan vain yhtä osa-aluetta kerrallaan. Yksikkötestaus oli rajoittunutta web-käyttöliittymän ja tietokannan osalta, sillä niiden toiminta on pitkälti riippuvaista toisistaan. Lopuksi suoritettiin järjestelmätestaus, jossa kaikki osa-alueet olivat toiminnassa muodostaen työn tavoitteen mukaisen tulospalvelujärjestelmän.

Yksikkötestaus palvelinsovellukselle suoritettiin käyttämällä opinnäytetyön ulkopuolista C++-ohjelmointikielellä kirjoitettua ohjelmaa, joka simuloi LapSnapper Pro Lap Timing -sovelluksen lähettämiä viestejä. Palvelinsovelluksen tekemät toimenpiteet tulostettiin ohjelmasta ulos palvelimen kovalevylle tekstitiedostoon. Tekstitiedostosta pystyttiin seuraamaan sovelluksen toimintaa.

Web-käyttöliittymän osalta yksikkötestaus jäi vähäiseksi, sillä ilman tietokantaa siihen ei voida kirjautua sisälle. Ainoastaan käyttöliittymän näkyminen web-palvelimella pystyttiin varmentamaan. Testauksessa käytettiin apuna tietokantaa. Testaamisprosessi suoritettiin toteutetun käyttöliittymän kautta ja seuraamalla sitten PhpPgAdmin-tietokantahallintaohjelman kautta web-käyttöliittymän tekemiä muutoksia.

Tietokannan testaaminen toteutui osittain sekä web-käyttöliittymän testauksen yhteydessä, että järjestelmätestauksessa. Järjestelmätestausta suoritettiin lähettämällä palvelinohjelmistoa varten suunnitellulla ohjelmalla simulaatioviestejä, jotka palvelinsovellus vastaanotti, ja joiden sisältämän tiedon se talletti tietokantaan. Tietokantaan tallennettu tieto oli taas tarkasteltavissa web-käyttöliittymän kautta.

6. POHDINTA

Työn tuloksena syntynyt palvelukokonaisuus vastasi hyvin kuvaa siitä kokonaisuudesta, joka palvelusta pystyttiin saamaan sen suunnitteluvaiheessa. Työn jakaminen kolmeen selkeästi toisistaan eroavaan osa-alueeseen helpotti työskentelyä muun muassa tilanteissa, joissa paikannettiin ohjelmointivirheitä.

Testausten perusteella valmis työ on kevyt ja nopeatoiminen palvelu, joka tekee juuri sen, mitä siltä vaaditaan, ilman ylimääräisiä ominaisuuksia. Ulkoasun yksinkertaisuus voidaan nähdä palvelussa heikkoutena verratessa nykyaikaisiin internet-sivuihin. Työssä kuitenkin tähdättiin eri toimintojen toimivuuteen, eikä niinkään ulkoasun näytettävyyteen. Ulkoasun parantaminen jätettiin jatkokehityksen aiheeksi.

Palvelinsovellusta testatessa havaittiin myös, että PHP-kielillä on mahdollista toteuttaa sovellus, joka kuuntelee palvelimelle tulevaa liikennettä. Tätä ominaisuutta lähdettiin aluksi lähinnä vain testaamaan, mutta sen osoittauduttua toimivaksi ei nähty enää syytä toteuttaa sovellusta perinteisemmin C tai C++ -kielillä.

6.1. Jatkokehitys

Tulospalvelun ensimmäinen jatkokehityskaskel tulee todennäköisimmin olemaan sektoriaikojen lisääminen tulosten seurantaan. Tämä tietysti edellyttää sektoriaikojen tunnistamisen tulosdatasta, mikäli sellaisia on. LapSnapper-ajanottojärjestelmä tukee tätä ominaisuutta jo, joten on luonnollista tuoda ominaisuus myös tulospalveluun.

Palvelun ulkoasun kehittäminen on myös yksi mahdollinen jatkokehityksen aihe. Paranneltu ulkoasu tekisi palvelun käyttämisestä miellyttävämmän kokemuksen. Lisäksi mahdollisia muita lisäominaisuuksia lisätessä harkitumpi ulkoasu voi muodostua sivuston helppokäyttöisyyden kannalta jopa välttämättömäksi.

LÄHTEET

Countries Listsin www-sivut 2014. Hakupäivä 26.4.2014. <<http://www.countries-list.info/>>

LapSnapperin www-sivut 2014. Hakupäivä 19.8.2014. <<http://www.lapsnapper.com>>

PhpPhAdminin www-sivut 2014. Hakupäivä 26.9.2014.

<<http://phppgadmin.sourceforge.net>>

PostgreSQL:n www-sivut 2014. Hakupäivä 6.8.2014. <<http://www.postgresql.org>>

TextPadin www-sivut 2014. Hakupäivä 26.9.2014. <<http://www.textpad.com>>

YED Graph Editorin www-sivut 2014. Hakupäivä 26.9.2014.

<http://www.yworks.com/en/products_yed_about.html>