

# Web-sivuston kävijäseurantajärjestelmä

## 1. Johdanto

Järjestelmän tarkoitus on kerätä jonkin tietyn web-sivuston käyttäjien tekemiä toimintoja ylös ja tarjota työkaluja erilaisten analyysien/raporttien tekemiseen tästä kävijädatasta.

Järjestelmä kerää halutulta sivustolta sen käyttäjien tekemiä liikkeitä sivustolla. Järjestelmän käyttäjät voivat ajaa tilastollisia raportteja datasta esimerkiksi suosituimmat sivut tietyllä aikavälillä, sivujen latausmäärät joiden urlissa tiettyjä merkkejä, tai vaikka suosituimmat sivut evästeiltä, jotka tehneet jonkun tietyn toiminnon.

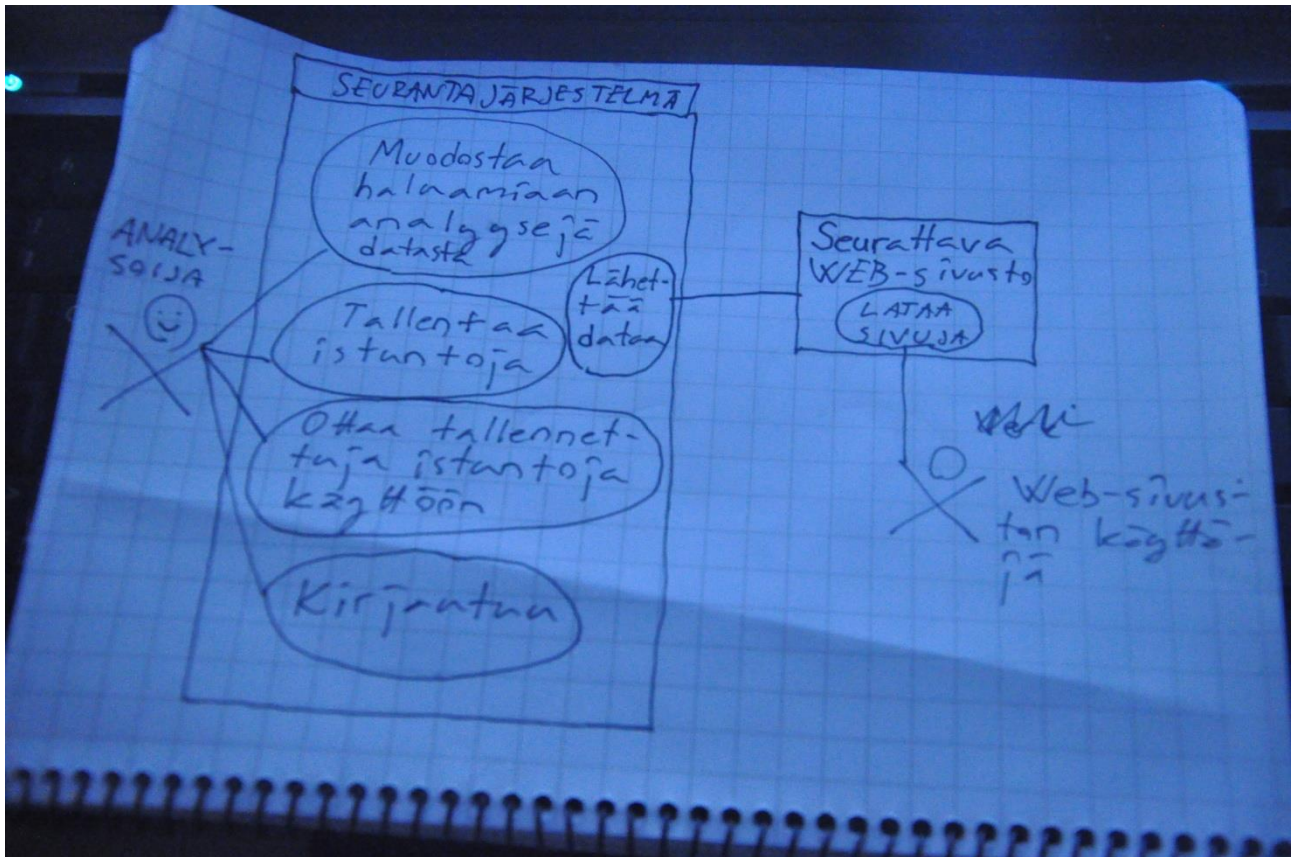
## 2. Yleiskuva järjestelmästä

Järjestelmän käyttäjät käyttävät palvelua omilla tunnuksilla ja he voivat tallettaa omia sessioita, jolloin on mahdollista saada joku tietty aiemmin tehty analyysi nopeasti käyttöön.

Toteutus tehdään palvelimen päässä php:lla ja tiedot tallennetaan Postgres -tietokantaan. Seurattava sivusto tagitetaan sivujen html-koodiin sijoitettavalla tagilla, joka lähettää järjestelmälle tietoa jokaisesta ladatusta sivusta ja mm. kävijän selaimen evästeen, jolla yksittäinen selain voidaan tunnistaa. PHP parsii sisääntulevasta datasta oleelliset tiedot, jäsentää ne ja tallettaa tietokantatauluihin. Käyttöliittymä toteutetaan PHP:lla ja joka muodostaa HTML-sivuja järjestelmän käyttäjien valinnoista mm. muodostaa analyyskejä kannoista haettavasta datasta ja esittää ne luettavassa muodossa.

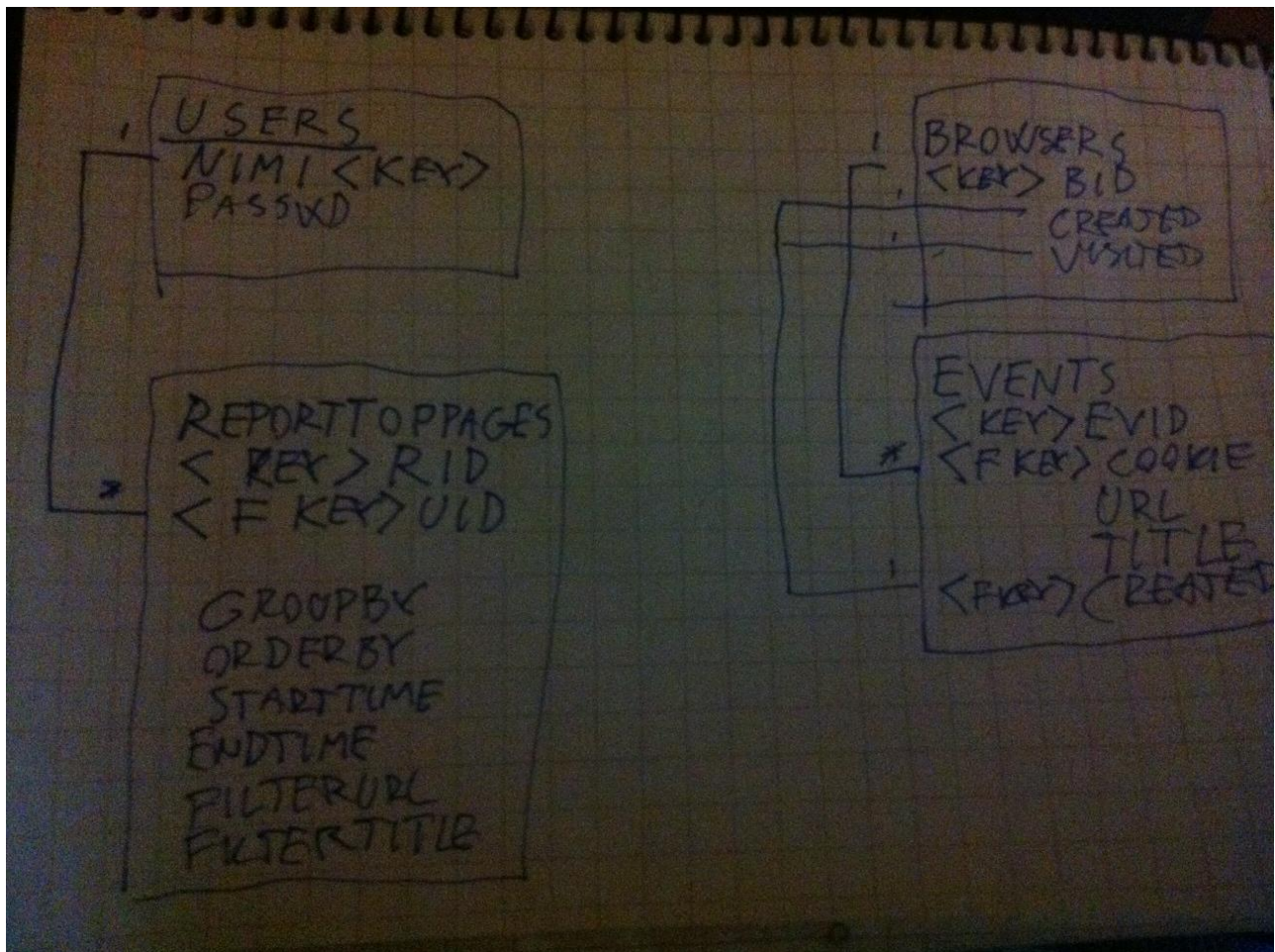
Käyttöliittymässä käyttäjä voi lisätä, poistaa ja tallettaa raporttinäkymiä haluamillaan asetuksilla.

### 3. Käyttötapauskaavio



### 4. Reliaatiokaavio

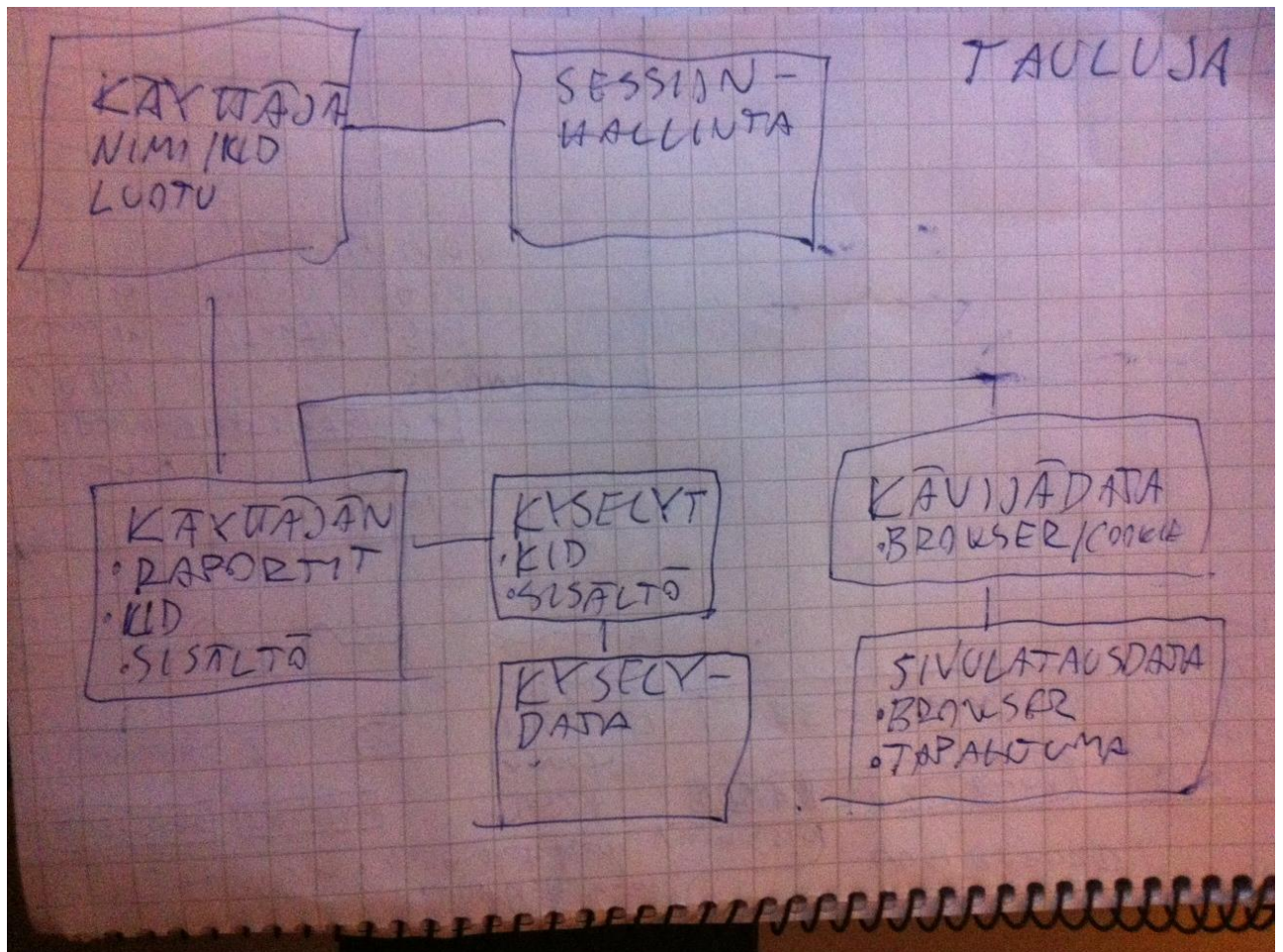
Tässä lopullinen relaatiokaavio:



EVENTS ja REPORTTOPPAGES on välillinen yhteys, jota ei merkata relaatiokaavioon. Esim. kood käyttää arvoja GROUPBY ORDERBY, STARTTIME, ENDDTIME, FILTERURL/TITLE muodostaakseen haun tauluun EVENTS viitaten sen attributteihin EVID, COOKIE, URL, TITLE ja CREATED

Aiempi luonnostelu relaatiokaaviosta:





## 5. näkymien varhaista luonnostelua

Kirjautuminen

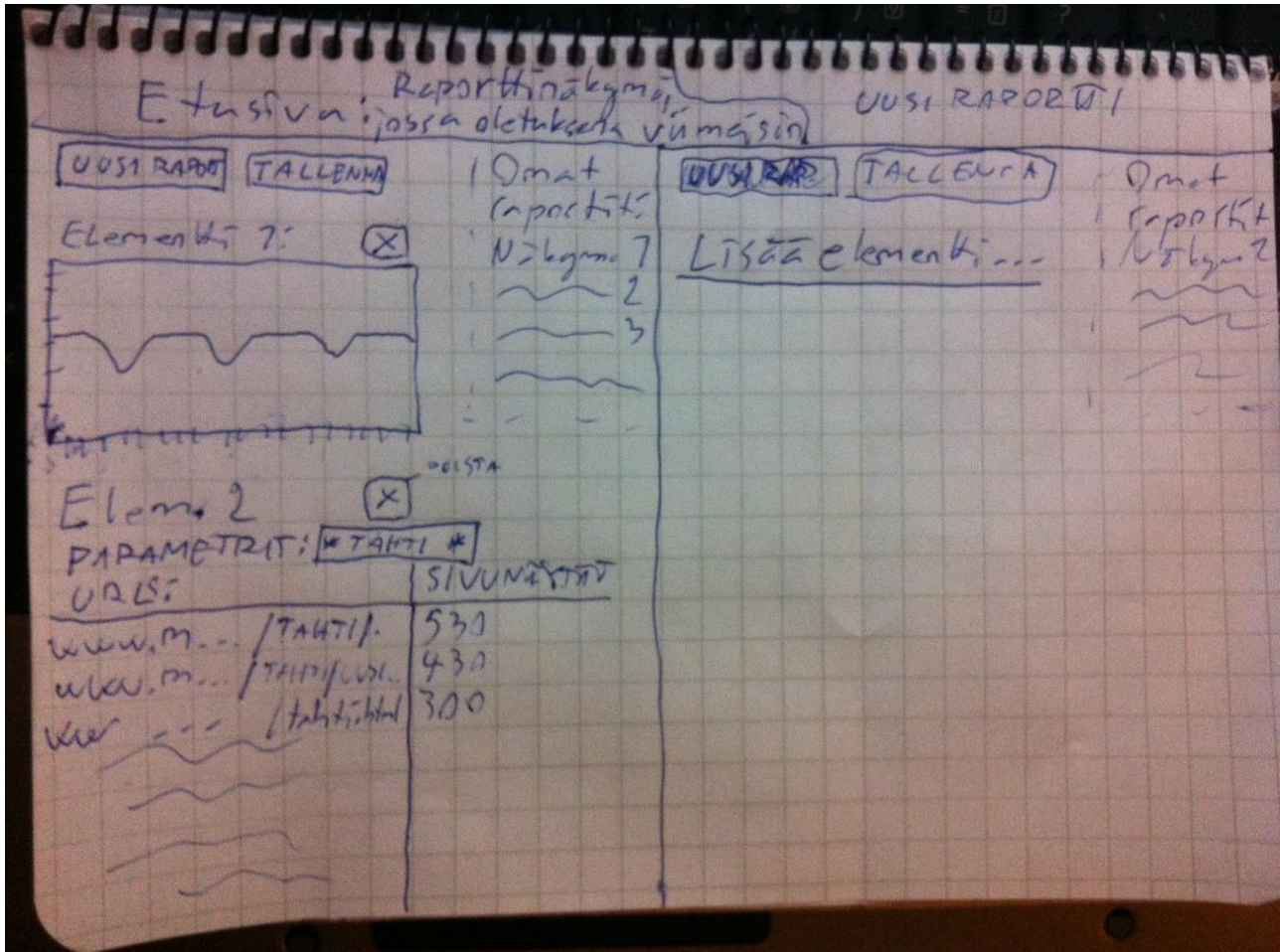
Tunnus:

Salasana:

☐ LÄHETÄ

"Kirjautuminen Vastaa!"





## 6. Järjestelmän yleisrakenne

Koostuu kahdesta rakenteellisesta osasta:

**Palvelun käyttäjien toimintoihin reagoivat**

**Tiedonkeruun toiminnot**

Näille yhteistä ovat ainoastaan tietokantataulut ja niiden hallintatiedosto. Ensimmäisessä toiminallisuudet on jaettu kontrolleri, malli ja näyttöön. Jälkimmäisessä on kontrolleri ja malli eriytetty, mutta ei näyttö-osaa ollenkaan, koska tiedonkeruu on seurattavan sivuston kävijöille näkymätöntä.

### Kontrolleri - Palvelun käyttöön liittyvä

*Index.php*

Hoitaa istunnonhallinnan ja ohjaa kirjautumiseen tarvittaessa

→kirjaudu.php (view)

Hoitaa käyttäjän syötteiden validoinnin ja koostamisen esim. raporttien komennot

→validoi.php (kontrolleri)

Pyytää tietokantojen hallinnalta syötteiden mukaisia dataa

→tietokantatoiminnot.php (model)

Ohjaa koostetut datat näytettäväksi

→ etusivu.php (view)

*libs/validoi.php (kts. yllä)*

### **Malli - Palvelun käyttöön liittyvä**

*libs/model/tietokantatoiminnot.php*

sisältää funktioita tietokantahakuihin, joilla kootaan raporteissa tarvittava data.

### **Näyttö - Palvelun käyttöön liittyvä**

*view/pohjat/etusivu.php*

Sisältää pohjan käyttäjien päänäkyville, näkee kontrollerissa koostetun raportteihin liittyvän datan

Pohjasta kustutaan raporttien näyttämiseen tarvittavia funktioita, jotka käyttävät tuotua dataa

→ apufunktiot.php

*view/apufunktiot.php*

muodostaa ja näyttää pohjassa raportit kontrollerissa kootun datan mukaisesti

*kirjaudu.php*

Näyttää kirjautumissivun, käyttää kontrollerilta saatu dataa tarvittaessa

### **Kontrolleri - tiedonkeruu**

*track.php*

Käsittelee seurattavalta sivustoilta tulevan seurattavan datan mm. http-headereista, evästeistä ja GET-parametreihin lisättyä. Hallinnoi evästeiden käsittelyä saman käyttäjän tunnistamiseksi. Kommunikoi ja ohjaa kävijätiedon tietokantatauluihin tarkistuksia, lisäyksiä ja päivityksiä sisään tulevasta datasta.

→ tietokantatoiminnot.php

### **Malli - tiedonkeruu**

*tietokantatoiminnot.php*

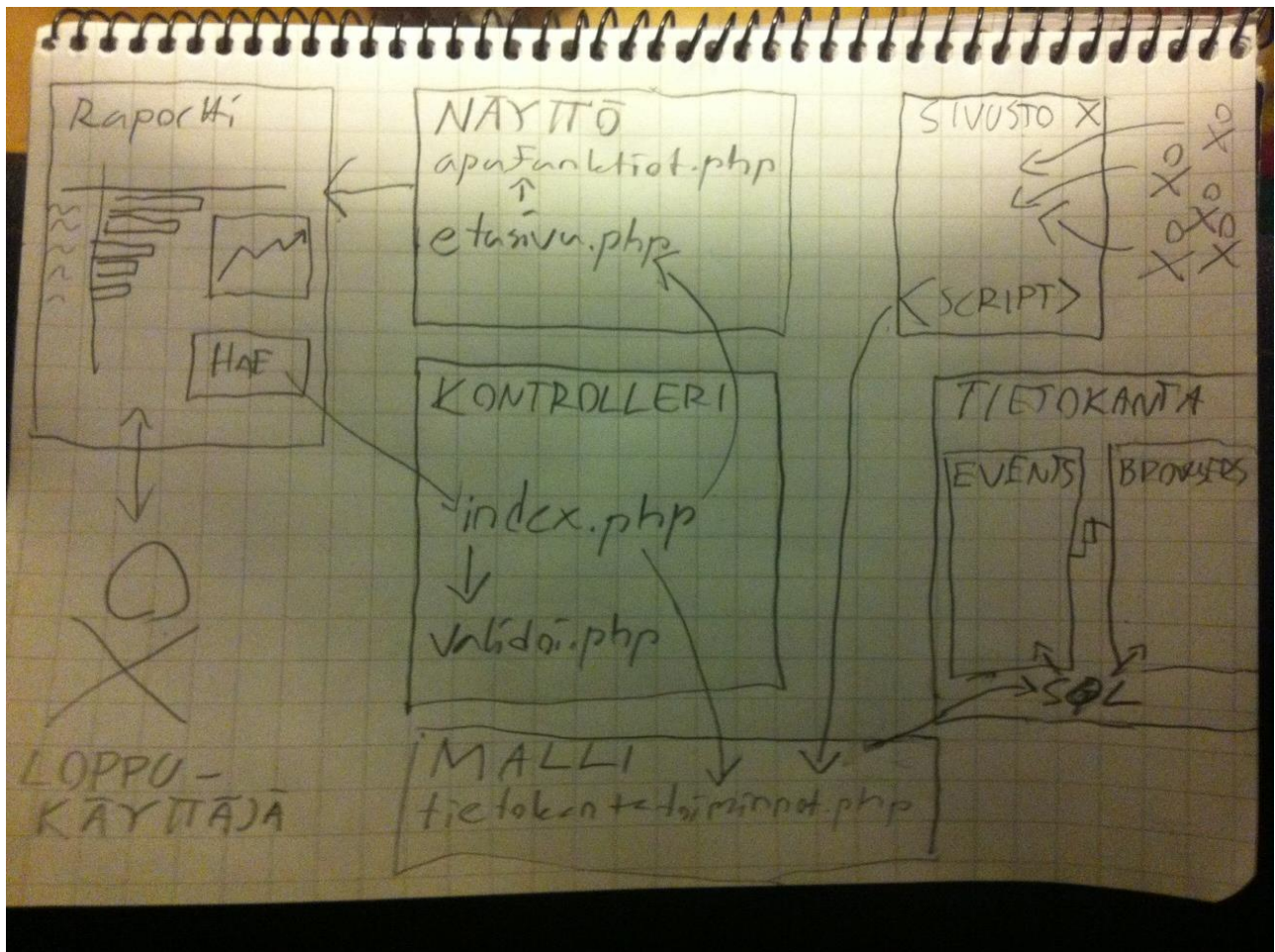
Sisältää tiedonkeruun toiminnot päivityksille, lisäyksille ja tarkastuksille

### **Tiedonkeruu**

*testisivu.php*

Seurattava sivuston jokaiselle sivun loppuosaan on lisätty Javascript -pohjainen tagi, joka lähettää

jokaisesta sivusta http-kutsun track.php tiedostolle. Kutsussa välitetään tietoa http-headereilla, evästeillä ja mm. sivun otsikko GET-parametrina.



## 7. Käyttöohjeet tarkastukseen

Kaikki tarvittavat linkit testaukseen löytyvät:

<http://porna.users.cs.helsinki.fi/tsoha/esittelysivu.html>

Muutamia huomiota testaukseen:

Kun lisää tai poistaa raportteja palvelussa, niin näkymä ei anna erityisen selkeää indikaatiota siitä, että lisäys tehtiin - ainoastaan nähdään, että TOP-sivut raporttien lukumäärä osoitin sivun alussa muuttuu. Muokkaus luo aina uuden sivulatauksen, jolloin saattaa hypätä sivun alkuun siitä, kohtaa missä teki muutoksen, jolloin voi tulla vaikutelma, että mitään ei tapahtunut. Näin ei kuitenkaan ole, vaan niiden pitäisi toimia oikein. Uusi raportti ilmestyy aina Top-sivut raporttien viimeiseksi.

Merkkaukset käyttöliittymään esim. värillisellä tekstillä siitä, mitä tehtiin edellisessä toiminnossa jäivät vielä uupumaan lopullisesta versiosta.

## 8. Mitä opin jne.?



Tietokantojen hallinnan perusproblematiikkaa tuli kohdattua hyvin mm. crudin haltuunotto vaati työstöä, jotta sen osasi tehdä järkevähköllä tavalla. SQL - injektio oli uutta hyödyllistä tietoa, joskin lopulta aika suoraviivainen asia.

Haastavinta oli syötteiden hallinta web-sivulta kontrollereihin ja siitä eteenpäin. Mitä enemmän toimintoja tuli, niin samalla monimutkaistui monet asiat. Ehkäpä se miten html:n syötekenttiä olisi voinut indeksoida fiksummin, jäi uupumaan kurssilla.

Bootstrap oli uusi tuttavuus ja vaikutti erittäin hyödylliseltä paketilta tehdä siistiä visuaalista jälkeä kätevästi. Valitettavasti en ehtinyt oikein perehtyä siihen kunnolla ja jäi edelleen paljon opittavaa siitä.