

# **Vanhojen tietokoneiden virtuaalimuseo**

---

(c) 2017, Tom Olander ( 014686795 )

# 1. Johdanto

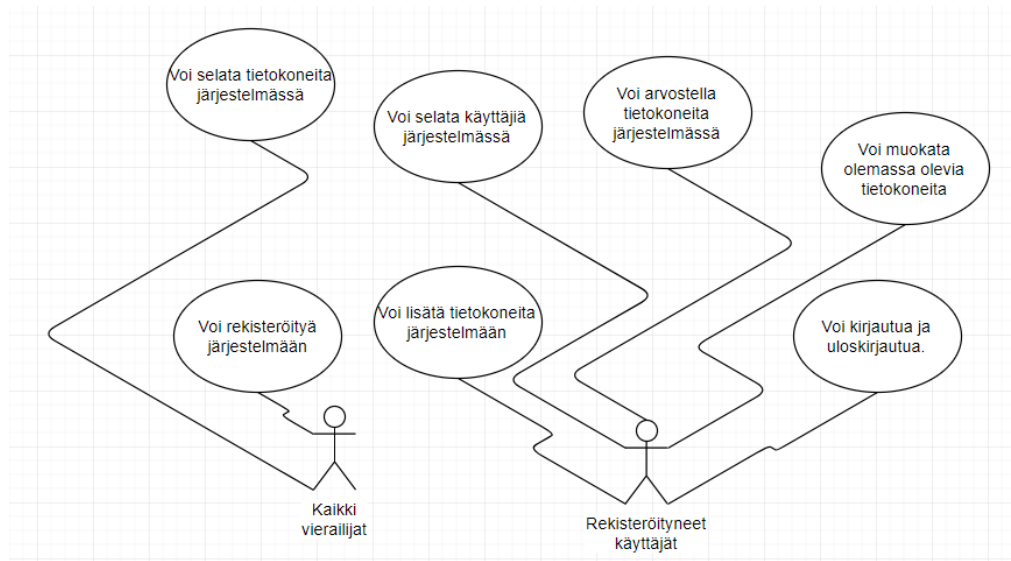
Tarkoituksena on rakentaa eräänlainen virtuaalimuseo vanhemmista tietokoneista (esim 8- ja 16-bittisiä tietokoneita). Idea olisi luoda tietopankki vanhoille tietokoneille missä tulisi esille niiden erikoisuudet sekä softan että kovon osalta.

Tavoitteena on luoda mahdollisimman helppokäyttöinen työkalu yllämainituille tarkoitukseen. Tarkoitus on myös luoda järjestelmä johon olisi suhteellisen helppo tulevaisuudessa lisätä ominaisuuksia.

Järjestelmän toteutus on PHP-kielellä ja Postgres-tietokannalla. Tässä projektissa ei syvennyttä kovinkaan paljon HTML:ään, CSS:ään eikä JavaScriptiin vaikka nämä ovat tekniikoita joita käytetään ja lisäämällä niiden käyttöä voitaisiin luoda hienommat ja interaktiivisemmat sivut. Järjestelmä ylläpidetään fyysisesti Helsingin Yliopiston users-palvelimella. Järjestelmä ei vaadi muuta kuin PHP- ja Postgresql-tuen, joten on helposti siirrettävissä muillekin palvelimille tarpeen vaatiessa.

## 2. Yleiskuva järjestelmästä

### Käyttötapauskaavio



### Käyttäjärühmät

Kaikki vierailijat

Tähän ryhmään kuuluvat kaikki jotka tulevat sivustolle, mukaanlukien kaikki jotka kuuluvat toisiinkin ryhmiin.

Rekisteröityneet käyttäjät Tähän ryhmään kuuluvat ne vierailijat jotka ovat rekisteröityneet ja kirjautuneet järjestelmään.

### Käyttötapauskuvaukset

Rekisteröityminen

Kaikki voivat rekisteröityä järjestelmään.

Käyttäjätiedot

Rekisteröidyt käyttäjät voivat muokata ja poistaa tietojansa.

”Tietokoneen” näyttäminen

Järjestelmän ”tietokoneet” ovat nähtävillä kaikille kävijöille.

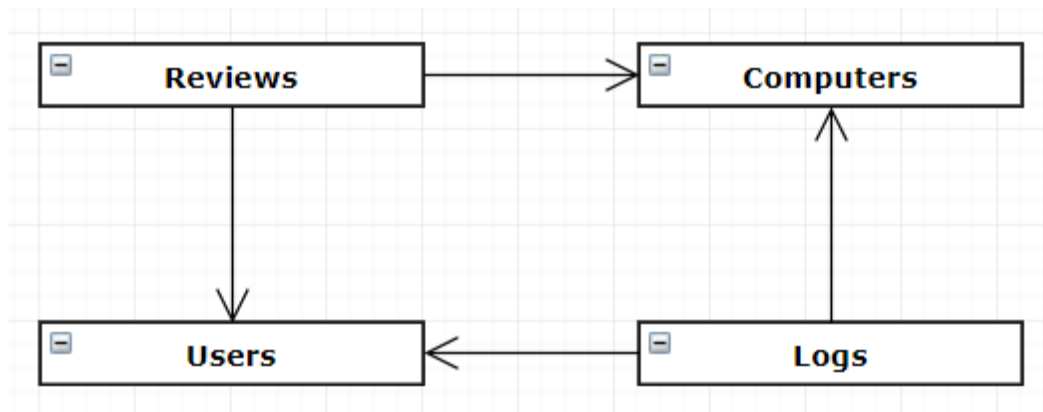
”Tietokoneen” muuttaminen ja lisääminen

Rekisteröidyt käyttäjät voivat luoda ja muokata ”tietokoneiden” tietoja.

Arvostelu

Rekisteröityneet käyttäjät voivat kirjoittaa arvosteluja ”tietokoneista” ja antaa niille arvosanan.

### 3. Järjestelmän tietosisältö



Tietokohde: Users

Name	Merkkijono max 50	Käyttäjän nimi järjestelmässä
Password	Merkkijono max 32	Salasana hash-muodossa, esim MD5
Motto	Merkkijono max 1024	Käyttäjän motto tai slogan, voi olla tyhjä
Datum	Timestamp aikaleima	Milloin käyttäjä on liittynyt järjestelmään

Taulu järjestelmän rekisteröityneistä käyttäjistä. Salasana tulee taulussa olla muussa muodossa kuin puhtaana tekstinä, eli käytetään jotain hajautusfunktiota hyväksi.

Tietokohde: Computers

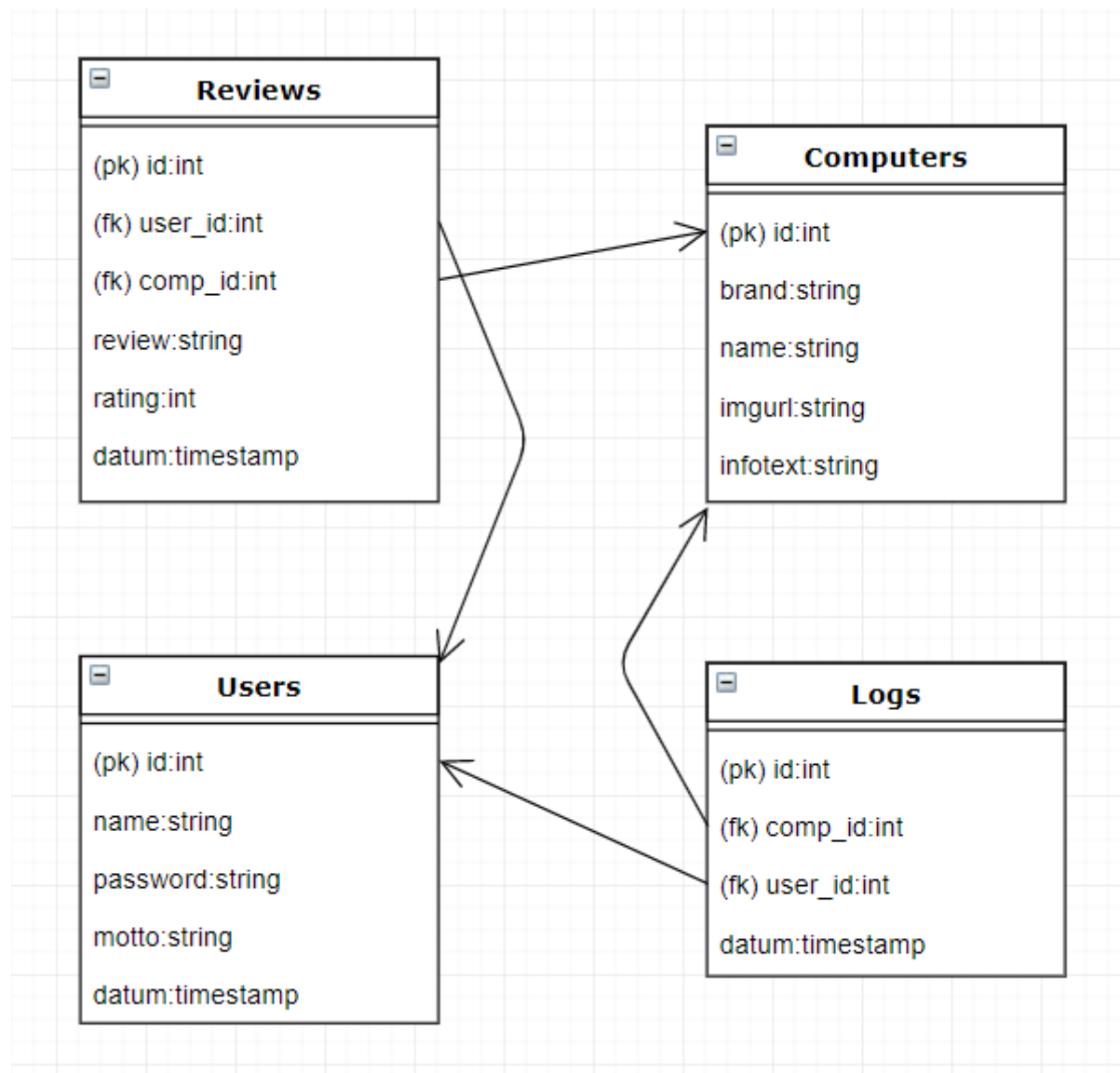
Brand	Merkkijono max 50	Tietokoneen merkki
Name	Merkkijono max 50	Tietokoneen nimi
Imgurl	Merkkijono max 255	Linkki kuvaan
Infotext	Merkkijono max 8192	Yleistä tekstiä tietokoneesta

Taulu järjestelmään liitetystä tietokoneesta, tähän tulee vain tietoja itse tietokoneesta ei muuta. Sekä Logs- että Reviews- taulut käyttävät tämän taulun pääavainta hyväksi viittauksissa.

Tietokohde: Logs		
Comp_id	Kokonaisluku	Viiteavain Computers-tauluun
User_id	Kokonaisluku	Viiteavain Users-tauluun
Datum	Timestamp aikaleima	Linkki kuvaan
<p>Liitostaulu Computers- ja Users-taulujen välille välttääksemme monesta moneen taulurelaatioita.</p>		

Tietokohde: Reviews		
User_id	Kokonaisluku	Viiteavain Users-tauluun
Comp_id	Kokonaisluku	Viiteavain Computer-tauluun
Review	Merkkijono max 2048	Arvostelun tekstiosuus
Rating	Kokonaisluku	Arvostelu (1-5)
Datum	Timestamp aikaleima	Milloin arvostelu tehty
<p>Tähän tauluun tulee käyttäjien tekemät arvostelut eri tietokoneista. Teksti ei saa sisältää HTML-tageja.</p>		

## 4. Relaatiotietokantakaavio



## 5. Järjestelmän yleisrakenne

Tietokantasovellusta tehdessä on noudatettu MVC-mallia. Kontrollerit, näkymät ja mallit sijaitsevat hakemistoissa controllers, views- ja models. Käytetyt apukirjastot on sijoitettu hakemistoon lib ja asetukset ovat tiedostossa settings.php.

Tarvittavat kuvatiedostot ja tyylitiedostot löytyvät assets-hakemiston alahakemistoissa img ja css.

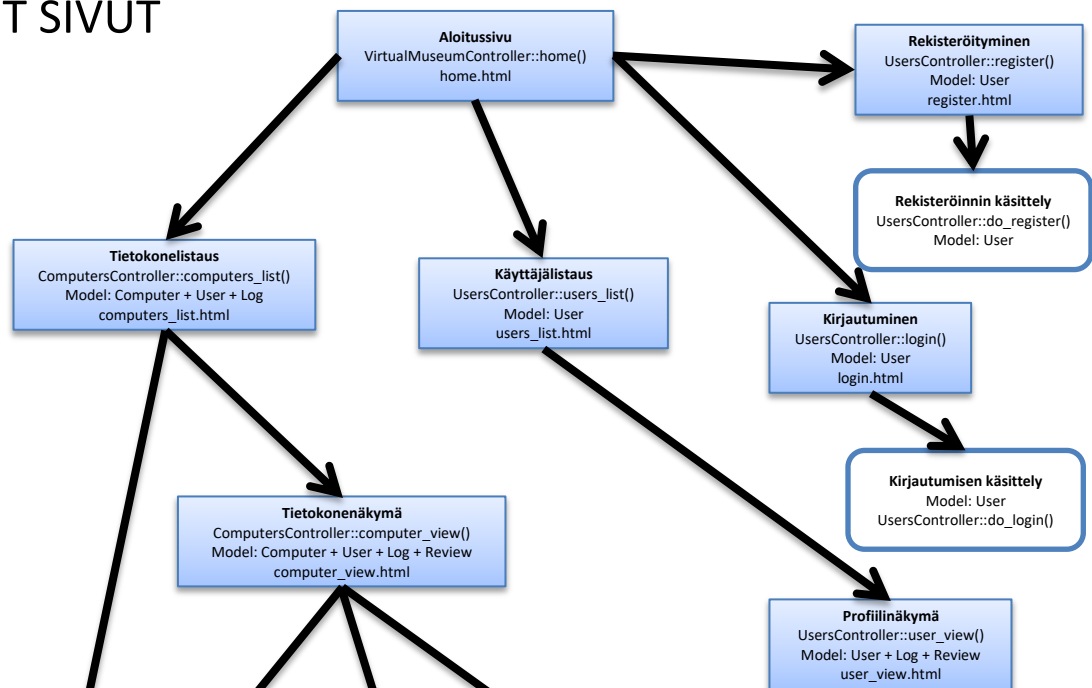
Tällä hetkellä ei käytetä muita JavaScript-toimintoja kuin ne mitkä mahdollisesti tulevat käytetyn Bootstrapin kautta.

Käyttäjän kirjautumisen yhteydessä käytetään PHP:n \$\_SESSION-supergloaalimuuttujaa. \$\_SESSION-muuttujaan ei tallenneta muutakuin käyttäjän tietokantatietueen id-arvo, jotta voidaan minimoida mahdollisia tietoturvariskejä.

Ohjelmakoodi on kommentoitu, josta selviää tarkemmin eri osien käyttö ja tarkoitus (hieman kesken vielä... lisää tulossa).

## 6. Käyttöliittymä ja järjestelmän komponentti

### JULKISET SIVUT



### KIRJAUTUMISTA VAATIVAT SIVUT

Sivustolla on myös naviigointipalkki jolla nopeasti ja helposti aina pääsee kirjautumis/uloskirjautumissivuille, rekisteröintisivulle, tietokoneistaukseen, käyttäjälistaukseen sekä takaisin aloitussivulle.



## 7. Asennustiedot

Järjestelmä on helppo ottaa käyttöön uudella palvelimella, varsinkin jos on Linux-palvelin, jolloin koko paketti siirretään uudelle palvelimelle. Config-hakemistossa olevat tiedot tietokannalle ja ympäristölle pitää muokata uudelle palvelimelle sopivaksi. Jos haluaa käyttää valmista `deploy.sh` skriptia, niin sen sisällä olevat `rsync`- ja `ssh`-käskyt pitää myös muokata hieman.

Järjestelmä voidaan myös ottaa käyttöön ilman `deploy.sh`:ta, mutta silloin pitää itse huolehtia kansioden luku- ja kirjoitusoikeuksista.

## 8. Käynnistys- / käyttöohje

Sovellus toimii verkkoselaimessa joten muuta ei tarvita kuin linkki aloitussivulle:

<https://tolander.users.cs.helsinki.fi/tsoha/>

Jotta järjestelmää voidaan käyttää kaikilla toiminnoilla järjestelmään on luotu myös testikäyttäjää.

### Testikäyttäjä

Käyttäjätunnus:	Dani
Salasana:	1234

HUOM! Käyttäjätunnus ei ole merkkikokoriippuvainen.

Testikäyttäjä on jätetty järjestelmään, vaikka rekisteröityminen toimii ja sitä pitäisi käyttää.

## 9. Testaus, tunnetut bugit ja puutteet & jatkokehitysideat

Järjestelmän kaikki toiminnot ovat testattu laajasti, eikä testikäytössä ole ilmaantunut bugeja. Testaus on tapahtunut manuaalisesti ilman yksikkötestejä. Vaikka järjestelmän vikasietoa on pyritty tekemään hyväksi, niin järjestelmää ei olla testattu jos itse tietokantajärjestelmä takkuilee. Toisaalta niissäkin tapauksissa järjestelmä pitäisi toimia kunnolla ja ilmoittaa ongelmista käyttäjälle, sovellus ei pitäisi kaatua missään tapauksessa.

Järjestelmä on melko askeettinen ja hyvin yksinkertainen. Todellisuudessa tällaisessa järjestelmässä pitäisi mahdollistaa kuvien liittämistä sivuille ja myös jonkinlaista merkkauksjärjestelmää tulisi käyttää tekstiosuuksille, esimerkiksi Markdown tai BB-codea. Haku toimintoja olisi helppo lisätä järjestelmään, ja olisivatkin hyvä olla kun järjestelmään lisätään enemmän tietokoneita, sama koskee myös listauksien sivutusta mitä ei tässä versiossa vielä tehdä.

Optimaalisessa järjestelmässä tietyt asiat olisi toteutettu hieman eri tavalla, mutta se olisi tehnyt työstä aika paljon laajemmin. Esimerkiksi tietokonevalmistajat voisivat olla eri tietokantataulussa mihin niitä lisätään ja voisi lisätä myös yleistä tietoa yrityksestä, esimerkiksi sen historiasta, ja sen jälkeen itse tietokoneet eivät enää sisältäisi valmistajan tiedot tekstijonona vaan linkattuina tähän edellämainittuun tauluun.

Järjestelmään voisi myls lisätä muita tietoja tietokoneista joita sitten voisi käyttää hakuehdoissa, kuten ensiesittely vuosi ja prosessorin bittisyys, mahdollisesti voisi jopa luoda oman taulun prosessoreille koska samoja prosessoreita käytettiin monessa eri tietokoneessa, esimerkiksi MOS6502, Z80 ja MC68000 olivat erittäin suosittuja vanhoissa tietokoneissa.

## 10. Omat kokemukset

Sovelluksen tekeminen oli suhteellisen helppoa, kun näitä on ennenkin tullut tehtyä aika paljon. Vaikka olenkin käyttänyt PHP ohjelmointikielenä monessakin edellisissä sovelluksissani, niin en ole kovinkaan paljon käyttänyt PHP:n objekteja ohjelmissa, joten oli mielenkiintoista tutustua tarkemmin niiden toimintaan ja miten PDO-tietokantakutsut toimivat.

Myös bootstrapissa käytetty template-järjestelmä, Twig, oli uusi tuttava, joten piti hieman selvittää miten tietyt asiat siinä toimivat.

Kaikenkaikkiaan oli hauska päästä tekemään tietokantasovellus pitkästä ajasta, kun edellisestä kerrasta on jo päässyt vierähtämään monta vuotta (en laske tähän mukaan TiKaPe-kurssissa tehtyä keskustelupalstaa).