

# Aineopintojen harjoitustyö: Tietokantasovellus (periodi III)

## Johdanto

Työn aiheena on lautapeliien arkistointi, arvostelu ja informaatiojärjestelmä. Järjestelmään on tarkoitus tallentaa lautapelejä, niiden tietoja ja peli-informaatiota.

Myöhemmin järjestelmään on myös tarkoitus tallentaa itse pelistatistiikkaa, käyttäjien peli-arvosteluita ja muita asioita, jos aikaa riittää.

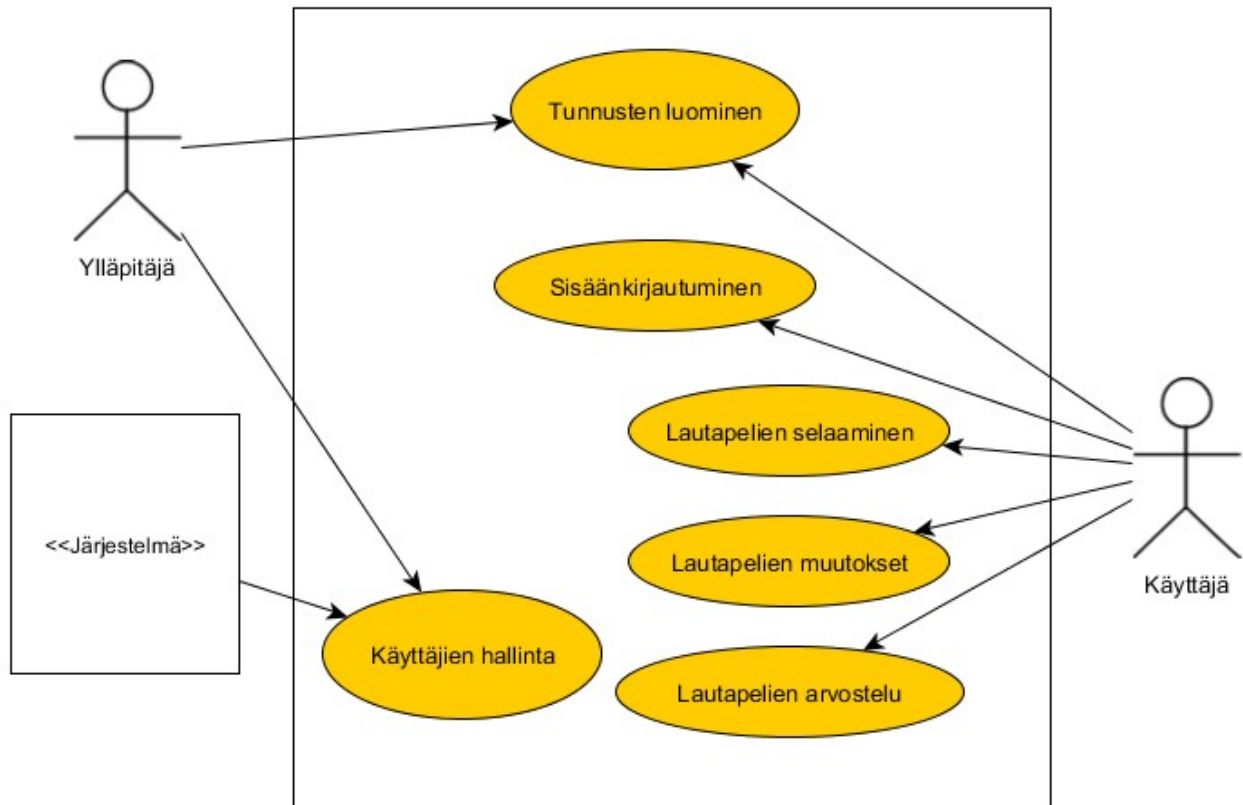
Käyttäjät tekevät järjestelmään käyttäjätunnuksen, jonka avulla he voivat oikeuksien mukaan syöttää järjestelmään tietoja, tai tarkastella sitä.

Järjestelmä toteutetaan PHP:lla ja PostgreSQL:llä, ja toimii loppukäyttäjälle selaimen välityksellä.

Tällä hetkellä suunnitelmissa ei ole erityistä tarvetta tuelle käyttäjän selaimessa. Normaali html-tuki riittänee.

Tarkoitus on tehdä työ vain yhdellä tietokannalla (PostgreSQL), sitä ei ole tarkoitus vaihtaa toiseen, mutta pienellä työllä se on mahdollista siirtää toiseen ympäristöön.

# Käyttötapaukset



## Käyttäjärühmät

Erilaisten toimijoiden (actor) määräksi mietin kolmea: itse järjestelmä, jonka kautta voidaan esim. poistaa inaktiivisia käyttäjiä, normaalit käyttäjät normaalein oikeuksin, sekä ylläpitäjän oikeuksin varustetut käyttäjät.

Normaali käyttäjä voi luoda itselleen tunnukset sivustolle, jotka ylläpitäjä vahvistaa ja aktivoi. Tämän jälkeen käyttäjä voi selata arkistoa, tehdä muutoksia (lisätä, poistaa, muokata) tai kirjoittaa arvosteluja arkiston sisällöstä.

Ylläpitäjä on sivuston pääkäyttäjä, hänen tärkein erillinen tehtävä on normaaliin käyttäjien tunnusten aktivointi ja yhdessä järjestelmän kanssa käyttäjien hallinta.

Järjestelmä on käyttötapauskaaviossa piirretty erikseen. Sen tarkoituksena on toimia systeemin taustalla mahdollistaen toiminnan. Ajan puitteissa on tarkoituksena myös lisätä perustoiminallisuutta järjestelmään, kuten inaktiivisten käyttäjien poistaminen (aktivoimattomat, bottien luomat käyttäjät).

## Käyttötapauksen kuvaukset

### Ylläpitäjän käyttötapaukset

Käyttäjien hallinta:

Ylläpitäjä voi muokata ja poistaa käyttäjiä, lukea käyttäjien tietoja (esim. Järjestelmään jätetty sähköpostiosoite).

## Tunnusten luominen

Kun käyttäjä on ensimmäisen kerran kirjautunut sivuille ja luonut tunnuksensa, pääkäyttäjä eli ylläpitäjä käy aktivoimassa hänen tunnuksensa.

## Järjestelmän käyttötapaukset

### Käyttäjien hallinta

Järjestelmä pystyy tarpeen tullen poistamaan käyttäjien joukosta ongelmatapauksia, kuten muiden järjestelmien luomia käyttäjiä.

## Käyttäjän käyttötapaukset

### Tunnusten luominen

Ennen kuin käyttäjä pääsee käyttämään järjestelmää, hänen tulee luoda tunnukset. Rekisteröityessä palveluun täytetään lomake, jossa pyydetään muutamia tärkeimpiä tietoja käyttäjästä. Ylläpitäjä käy aktivoimassa tunnukset.

### Sisäänkirjautuminen

Rekisteröitynyt käyttäjä voi kirjautua sisään käyttäjätunnuksellaan ja salasanallaan.

### Lautapeliien selaaminen

Rekisteröitynyt käyttäjä voi selata lautapeliarkistoa erilaisin hakuehdoin.

### Lautapeliien muutokset

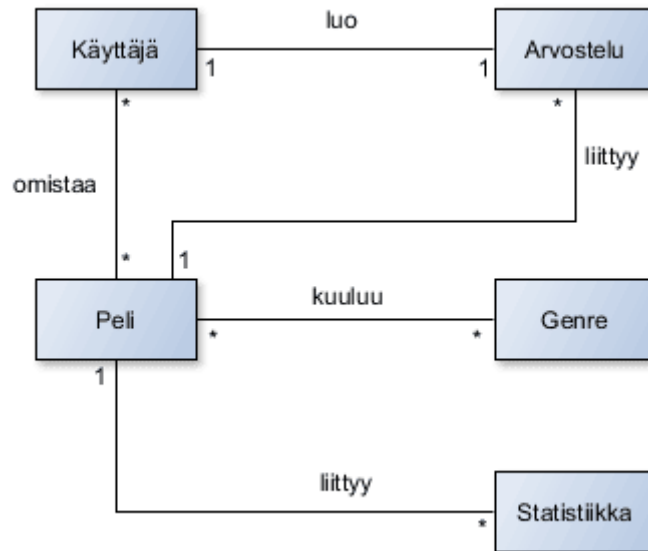
Muutoksilla järjestelmään tarkoitetaan tässä myös lisäyksiä, poistoja ja muokkauksia. Tähänkin vaaditaan rekisteröityminen.

### Lautapeliien arvostelu

Rekisteröityneet käyttäjät voivat myös sisään kirjaututtuan kirjoittaa lyhyen kuvauksen pelistä ja antaa sille arvosanan. Arvosteluasteikkoa tai eri arvosanoja ja niiden sisältöä ei vielä tässä vaiheessa ole tarkkaan päätetty (kouluarvosana vs. raskaus/aloittelijaystävällisyys/tuuripeli-suunnittelupeli/genre)

# Tietokanta

## Järjestelmän tietosisältö



### Tietokohde: Käyttäjä

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku	juokseva numerointi käyttäjien identifiointiin
nimi	merkkijono, max 255 merkkiä	käyttäjän nimi
salasana	merkkijono, suojattu	käyttäjän salasana

### Tietokohde: Peli

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku	juokseva numerointi pelien identifiointiin
nimi	merkkijono, max 255 merkkiä	Pelin nimi
julkaisuvuosi	kokonaisluku	Julkaisuvuosi
julkaisija	merkkijono, max 255 merkkiä	Julkaisijan nimi
genre	kokonaisluku	Foreign_key Genre-tauluun
pelaajat_min	kokonaisluku	Pelaajien minimimäärä
pelaajat_max	kokonaisluku	Pelaajien maksimimäärä
lisäyspäivä	päivämäärä	Päivämäärä, jolloin peli lisätty tietokantaan

## Tietokohde: Arvostelu

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku	juokseva numerointi arvosteluiden identifiointiin
arvostelija	kokonaisluku	Foreign_key käyttäjän id
arvostelu	Merkkijono, max 5000 merkkiä	Vapaamuotoinen arvio pelistä
arvio	desimaaliluku	Arvio asteikolla 1...5

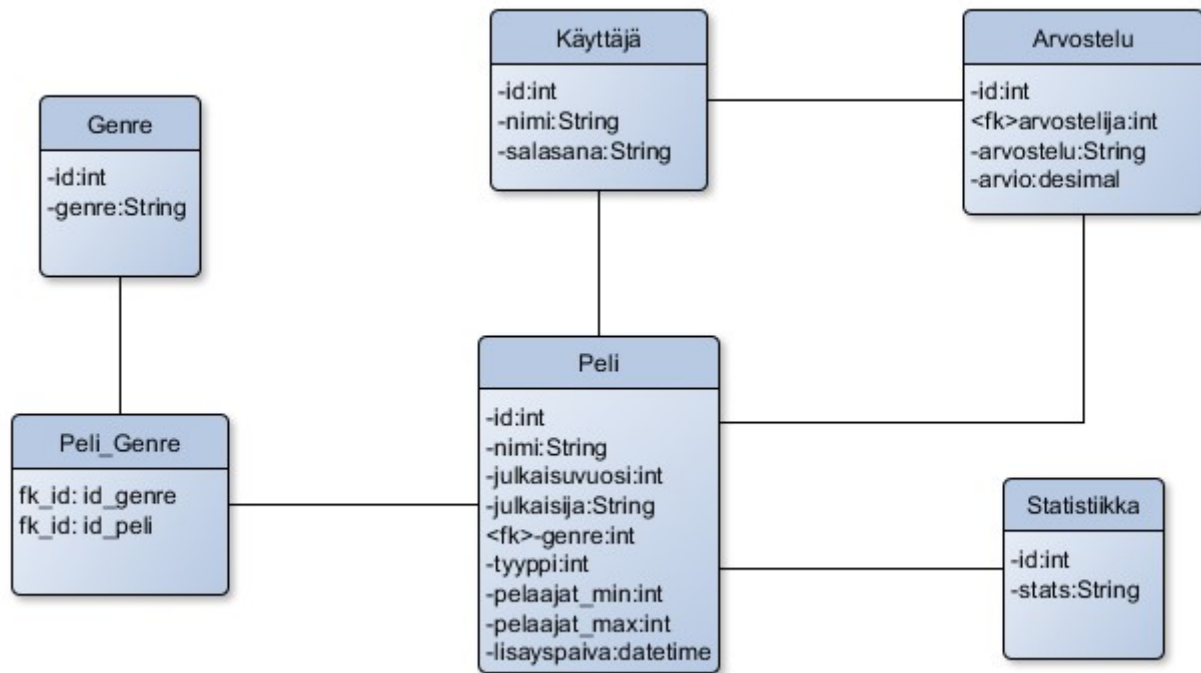
## Tietokohde: Genre

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku	juokseva numerointi genren identifiointiin
genre	Merkkijono, max 255 merkkiä	Genren tyyppi

## Tietokohde: Statistiikka

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku	juokseva numerointi statistiikan identifiointiin
stats	Merkkijono, max 5000 merkkiä	Teksti yhden pelin kulusta ja/tai pisteiden jaosta

## Relaatiotietokantakaavio



# Käynnistys- ja käyttöohje

Tietokantaa pääsee katselemaan ilman sisäänkirjautumista, mutta muokkauksia (uusien luonti, vanhojen muokkaaminen, olemassaolevien poistaminen) silloin ei pääse tekemään.

Sisään kirjaudutaan klikkaamalla oikeassa yläkulmassa olevaa Login-tekstiä. Tämän jälkeen käyttäjä saa eteensä lomakkeen, johon tulee syöttää käyttäjätunnus ja salasana. Näiden syöttämisen jälkeen painetaan “Kirjaudu sisään” -nappia. Oikein syötetyillä käyttäjätunnuksella ja salasanalla päästään pääsivulle, muuten saadaan virheilmoitus ja yritetään uudelleen.

Pääsivulla listataan kirjastosta löytyvät lautapelit. Taulukossa pelit on aakkosjärjestyksessä. Näkymässä näkee pelin nimen, julkaisuvuoden, julkaisijan, minimi- ja maksimimäärän pelaajia sekä pelin lisäyspäivämäärän. Pelin tietoja pääsee katsomaan pelin nimeä klikkaamalla tai painamalla Näytä-nappia. Yksittäisen pelin tietoja pääsee muuttamaan avaamalla pelikohtaisen esittelysivun ja painamalla Muokkaa -nappia.

Kun pelin nimeä klikkaa, saa eteensä pelin esittelysivun. Siinä kerrotaan tärkeimmät tiedot pelistä sekä vapaamuotoinen kuvaus. Peliä pääsee muokkaamaan “Muokkaa” napilla, tai pelin voi kokonaisuudessaan poistaa “Poista”-napilla.

## Järjestelmän yleisrakenne

Sovelluksessa on käytetty MVC (model, view, controller) -mallia. Järjestelmän mallit löytyvät kansiota */app/models*, näkymät kansiota */app/views* ja kontrollerit löytyvät kansiota */app/controllers*.

Järjestelmän asetukset ja reitit löytyvät kansiota */config*, jonka alla on ylläpitoa ja reitistystä varten tarvittavat tiedostot, mm. *routes.php*, jossa määritellään, mikä sivu esitetään käyttäjälle hänen osoittaessa selaimensa tiettyyn hakemistoon.

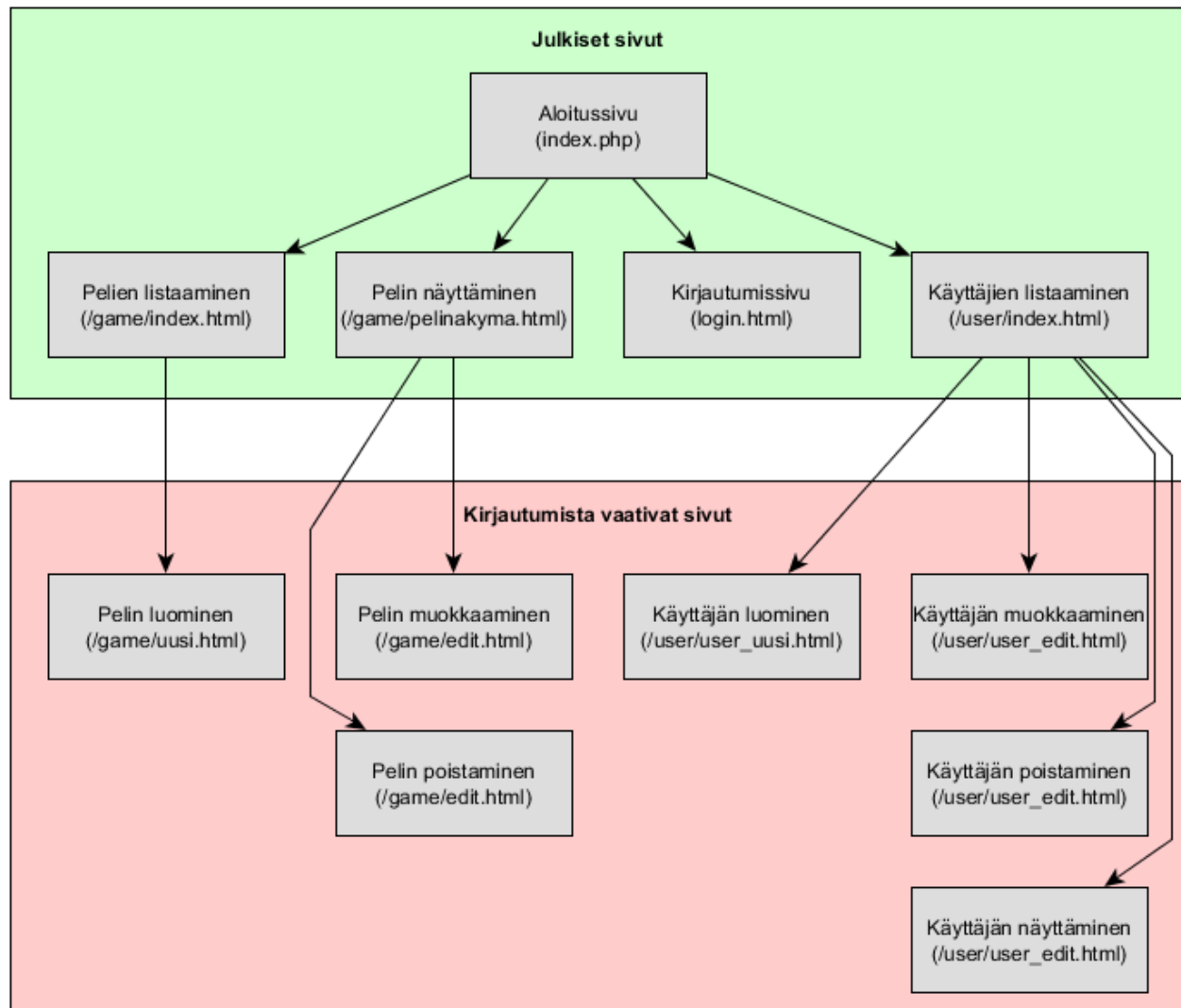
Järjestelmän sisällä näkymät on jaettu kahteen eri kansioon, joista toisessa sijaitsevat lautapeliin listaamiseen, näyttämiseen, editoimiseen ja poistamiseen liittyvät näkymät (*/app/views/game* -kansio) ja toisessa kansiossa (*/app/views/user*) on samat listaus-, näyttö-, editoimis- ja poistonäkymät käyttäjien hallintaan. Sivujen html-koodi on sijoitettu näihin kansioihin.

Mallien ja kontrollerien pohjana käytetyt *base\_model.php* ja *base\_controller.php* löytyvät kansiota */lib*. Testidataa ja taulujen luomista/poistamista varten skeemat ja skriptit löytyvät kansiota */sql*, ja loput bash-skriptit projektin juurikansiota.

Kaikki projektin tiedostonimet on kirjoitettu pienillä kirjaimilla.



# Käyttöliittymä ja järjestelmän komponentit



Kaaviossa on ilmoitettu sivut (näkymät), joiden kautta toiminnot etenevät. Itse toiminnoista vastaa *pelikontrolleri.php* tai *kayttajakontrolleri.php*, riippuen onko kyseessä peli vai käyttäjä.

Itse mallit ovat *pel.php* ja *kayttaja.php*, joskaan niitä ei tähän kaavioon ole merkitty.

Sovelluksessa on käytössä navigaatiopalkki, jonka kautta pääsee siirtymään aloitussivulle, pelien listaussivulle, käyttäjien listaussivulle ja kirjautumissivulle. Sisään kirjaututtua palkin linkistä voi kirjautua suoraan ulos, sisään kirjaututtua palkissa näkyy “(käyttäjä) kirjautunut sisään”.

# Asennustiedot

Sovellus on asennettuna Tietojenkäsittelytieteen laitoksen unix-palvelimelle, jossa se voi hyödyntää verkkopalvelinta.

Jos sovellus halutaan asentaa toiseen paikkaan, tulee varmistaa, että järjestelmä tukee php:ta ja postgresql:ää. Githubista voi hakea sovelluksen lähdekoodin ja asentaa sen haluamaansa paikkaan, tehdä tarvittavat muutokset */config*-hakemiston tiedostoihin, *environment.sh* ja *database.php*. Näistä ensimmäiseen tulee kirjoittaa palvelimelle kirjautuessa tarvittava käyttäjänimi sekä projektin hakemisto. Database.php-tiedostoon täytetään tietokannan tiedot.

Tämän jälkeen käynnistetään taulukoita luovat skriptit päähakemistosta, *create\_tables.sh* ja *add\_test\_data.sh*.

# Käynnistys- ja käyttöohje

Sovellus on asennettu Helsingin Yliopiston Tietojenkäsittelytieteen laitokselle, opiskelijoiden palvelimelle. Sen URL-osoite on

<http://luan.users.cs.helsinki.fi/tks/>

Vierailemalla tuolla osoitteella selaimella käyttäjä saa eteensä pääsivun, joka opastaa käyttäjää alkuun, ja referoi sovelluksen käytön. Sisään voi kirjautua oikean ylänurkan Login -linkistä. Testikäyttäjä voi kirjautua sisään tunnuksella *Antti* ja salasanalla *porkkana3*.

Sovelluksen käyttö on yksinkertaista, ja sovelluksen sisällä liikkuminen suoraviivaista.

# Testaus, tunnetut bugit ja puutteet sekä jatkokehitysideat

Sovellusta on testattu omin ja lainavoimin käyttämällä sovellusta sen omilla toiminnoilla verkkoselaimessa. Tietokantaa koskeissa kyselyissä komennot on ensin hiottu täydellisiksi psql-tulkilla users.cs.helsinki.fi -palvelimella, sitten lisätty osaksi sovellusta, kun ollaan varmoja että ne toimivat oikein. Tämänkin jälkeen sovellusta on vielä testattu käsin verkossa.

Jokainen sivu, malli, kontrolleri ja taulu on tarkistettu ja testattu, jokainen komento ja lauseke suoritettu, ja tällä hetkellä en ole löytänyt sovelluksesta yhtään bugia.

Puutteita ei sovellukseen sellaisenaan jäänyt – sovellus toimii kuten pitää, joskaan käyttäjäryhmiä ei sovellukseen ole luotu alkuperäisten suunnitelmien vastaisesti. Sovelluksessa toimii kolmelle tietokohteelle CRUD-nelikko, ja monesta-moneen -suhteille löytyy genret ja pelit yhdistävä taulu.

Lisäominaisuuksia sovellukseen voi vielä tehdä paljonkin – yhden periodin aika ei työssäkäyvälle ihmiselle ole kauhean paljon, mutta tarpeeksi, että sovellukseen saa pääkohdat ja muutamia ylimääräisiä toimintoja.

Muutamia asioita, mitä sovellukseen voi vielä lisätä:

- Uuden käyttäjän omatoiminen rekisteröityminen
- Hakutoiminnot kaikille tietokohteille (jäi kesken)
- Sivutus listauksille
- Deletoimisen varmistaminen (jäi kesken, ei toiminut)
- Pääkäyttäjien ja normaalien käyttäjien käyttöoikeuksien erottaminen
- Pelien статистиikan seuraaminen (ehkä pelien esittelysivulle, toiseen välilehteen?)

# Omat kokemukset

Projekti oli hyvin opettavainen. En ollut aiemmin tehnyt juuri mitään PHP:lla, SQL oli tuttu vain Tietokantojen Perusteet -kurssilta, sitä ennen konseptina. Nyt kurssin jälkeen tuntuu, että työkalupakki on paljon laajempi, ja varsinkin verkkosovellusten MVC-mallin ymmärtäminen on avannut silmät muitakin verkkosivuja selatessa. Tuntuu, että (ainakin aluksi) yksinkertaisia sovelluksia voi luoda itsekin tulevaisuudessa.

Kurssin materiaalia on monet kehuneet, itse yhdyn samaan kuoroon. Se lähtee hyvin yksinkertaisesti liikkeelle, ja esimerkit selventävät tapauksia hyvin. Virheilmoitusten käsittely oli ainakin minulle vaikeaa aluksi, kun ei ollut varma, tuleeko virheet PHP:sta, SQL:stä vai sovellusarkkitehtuurista (Twig, Bootstrapper). Turhauttavaa oli myös se, kun koodia tehtyä saikin vain eteensä tyhjän valkoisen sivun.

PHP-logien etsiminen oli ensin vaikeaa, mutta kun niiden paikan löysi (irc-kanavan keskustelua seuraamalla) huomasi, että tuota kautta virhekoodit selvisivät nopeasti ja niiden korjaaminenkin oli helppoa. Tätä ehkä kannattaa painottaa tulevaisuudessa tuon materiaalin suhteen – kenties tehdä erillinen alasivu virheiden käsittelystä ja voittamisesta. Nyt php-logien lukeminen mainitaan tekstissä, mutta huomasin sen vasta viikolla viisi tekstiä uudelleen silmäillessä.

Kirjoitin sovellusta lähinnä kotona Windows 8.1 -käyttöjärjestelmällä varustetulla koneella. Ajoin sen alla Hyper-V -virtuaalikoneessa Ubuntu 14.04:ää, jonka alle kurssin materiaalin sai kivasti asennettua. En tiedä, onko kurssia mahdollista järjestää siten, että koodi tehdään Windowsin alla NetBeansissa ja ajetaan samalla tavalla palvelimelle, mutta vähemmän virtuaalikoneilla puljanneille tai linuxia käyttäneille tämä voisi olla harkinnanalainen ratkaisu. Kaikilta ei kotoa löydy linux-konetta, eikä nykyaikaisilla SafeBoot-koneilla dualboottia saa kovin helposti aikaiseksi.

Kiitokset vielä myös lukuisista ohjauskerroista, ja siitä, että niitäkin järjestettiin kello 16 jälkeen. Ihmiselle, joka on arkisin töissä tämä oli kallisarvoista ja todella hyödyllistä.

Kaiken kaikkiaan, kiitos hyvästä ja opettavaisesta kurssista!

## Muu dokumentaatio

Git-repositorio projektille löytyy osoitteesta

<http://github.com/anti-l/tns>

SQL-skriptit taulujen luomiseen ja tuhoamiseen sekä testidatan luomiseen löytyvät kyseisen osoitteen alta, hakemistosta *sql*.