

# Lukion kurssityöhallinta- järjestelmä

## Summary

1 Johdanto.....	2
2 Yleiskuva järjestelmästä.....	2
Käyttäjäryhmat:.....	4
Käyttötapauskuvaukset.....	4
3 Järjestelmän tietosisältö.....	5
4 Relaatiotietokantakaavio.....	6
5 Järjestelmän yleisrakenne.....	6
6 Käyttöliittymä ja järjestelmän komponentit.....	7
7 Asennustiedot.....	7
8 Käynnistys- / käyttöohje.....	7
9 Testaus, tunnetut bugit ja puutteet & jatkokehitysideat .....	8
10 Omat kokemukset.....	8

[\[git@github.com:paapu68/TKsovellus.git\]](https://github.com:paapu68/TKsovellus.git)

[\[markus.kaukonen@iki.fi\]](mailto:markus.kaukonen@iki.fi)

Markus Kaukonen

Opiskelijanumero 010974524

# 1 Johdanto

Työssä luodaan tietokanta lukiokurssin opettaja-oppilaat nettikommunikaatiota varten. Se mahdollistaa esimerkiksi tiedostojen latauksen ja talletuksen siten että opettaja voi tarkastaa harjoitustyön, tehdä siihen mahdollisia korjauksia ja tallettaa sen korjattuna takaisin tietokantaan josta oppilas voi ladata tarkastetun version itselleen. Opettaja määrittelee listan kurssiin kuuluvista harjoitustöistä jotka opiskelijan tulisi tehdä, lisäksi opiskelijan on mahdollista tehdä ylimääräisiä vapaavalintaisia harjoitustöitä. Opettajan tulee voida antaa kustakin työstä arvosana ja palaute jotka oppilas voi halutessaan katsoa. Harjoitustöillä voi olla määräaika. Opettaja voi saada tilastotietoa kunkin kurssin tilasta.

Ylläpitäjä luo opettajat. Muut käyttäjät rekisteröityvät itse ja hheidän oletusasemansa on 'opiskelija'.

Tavoitteena on tehdä tämä kommunikaatio mahdollisimman käyttäjäystävällisesti koulun jokaiselle kurssille jonka opettaja haluaa käyttää tätä palvelua.

Ohjelma toimii yleisimmissä selaimissa, ainakin Firefoxissa ja Chromessa. Käyttöjärjestelmistä Windows, Linux ja Mac ovat tuettuja. Toivottavaa olisi myös toiminta Android laitteissa. Ohjelma pyörii tulevaisuudessa Heroku palvelimella.

Web sovelluksen alustan tulee tukea web2py Pythonia. Selaimelta ei edellytetä mitään erikoista. Ohjelmisto toimii PostgreSQL tietokannalla, mutta sen voi muuntaa tarvittaessa toiminaan SQLite:n tai MySQL:n kanssa.

## 2 Yleiskuva järjestelmästä

Käyttötapauskaavio on esitetty alla (se on tehty Umbrello-ohjelmalla, Ubuntu12).



## Käyttäjärühmat:

**Ylläpitäjä:** koulun valitsema henkilö jonka turvaluokitus on paras mahdollinen.

**Opettaja:** koulun opettaja

**Opiskelija:** koulun oppilas

## Käyttötapauskuvaukset

### Ylläpitäjän käyttötapauskset:

Opettajan luonti: liittää käyttäjän ryhmään 'opettaja'. 'Opettaja' ryhmään kuuluva voi esim antaa arvosanoja.

Muita käyttötappauksia: kirjautuminen, varmuuskopion otto (automaattinen?), käyttäjän blokkaminen (eli käyttäjä ei pääse kirjautumaan järjestelmään).

### Opettajan käyttötapauskset:

Kurssin valinta: opettaja valitsee kurseistaan yhden tai useamman jonka parissa alkaa työskennellä.

Kurssin työaiheiden antaminen: Opettaja kirjaa mahdollisten kurssitöiden otsikot järjestelmään. Kuhunkin työhön voi liittää vihjeen, esim. mistä tietoa voi hakea.

Tiedoston lataus: Vastanottaa opiskelijoiden tiedostoja (niiden korjausta varten opettajan omalla koneella).

Tiedoston talletus: Tallettaa korjatun harjoitustyön tiedoston systeemiin.

Yhden työn arvosanan anto: Antaa arvosanan oppilaalle kurssiin liittyvästä harjoitustyöstä.

Palautteen anto yhdestä työstä: Antaa kommentit opiskelijan tietystä harjoitustyöstä

Palautteen katsominen: Jos palautetta on sen pitäisi näkyä automaattisesti nettisivulla.

Kurssin tilaston katsominen: Opettaja voi tehdä tilastoja omista kurseistaan ja saada kurssin tilanteesta excel taulukko.

Muita käyttötappauksia: Oppilaan yhden kurssin tilaston tulostus, kurssin arvosanan anto, palautteen anto yhdestä työstä, palautteen anto oppilaalle koko kurssista, kurssin työaiheiden katselu.

### Opiskelijan käyttötapauskset:

Oppilaan luonti: Opiskelija luo itsensä järjestelmään.

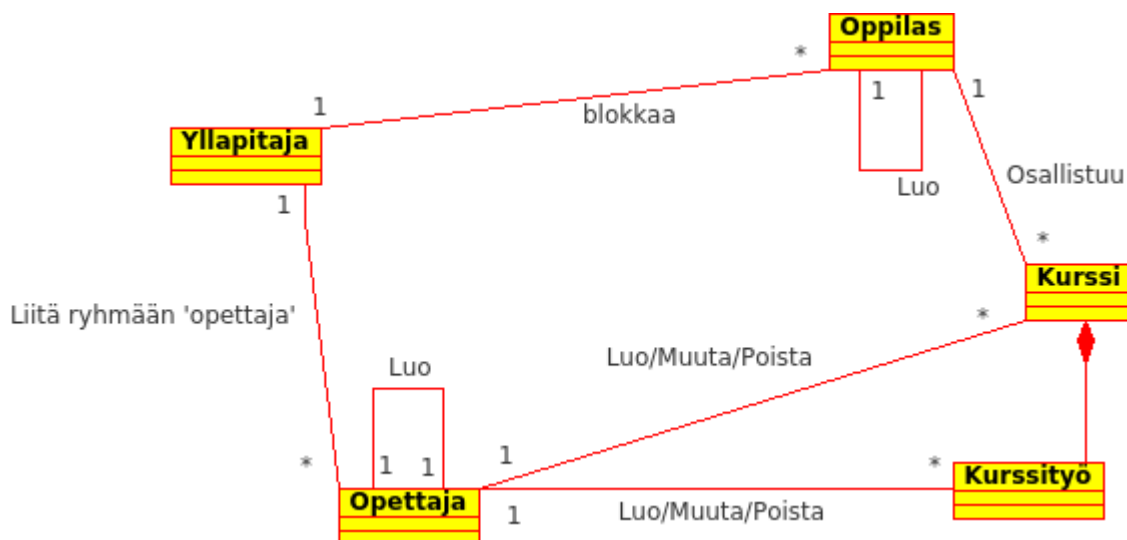
Oppilaan yhden kurssin tilaston katsominen: Opiskelija saa tilastotiedot kurssista montako harjoitustyötä kurssista on palauttamatta, kurssin harjoitustöiden arvosanojen keskiarvo ja mahdollinen kurssiarvosana.

Kurssin valinta: oppilas valitsee kurseistaan yhden tai useamman jonka parissa alkaa työskennellä.

Muita käyttötapauksia: kurssipalautteen katsominen, tiedoston lataus, tiedoston talletus, kirjautuminen, työaiheiden katselu.

### 3 Järjestelmän tietosisältö

Alla on esitelty järjestelmän tietösältö käsitekaavion avulla.

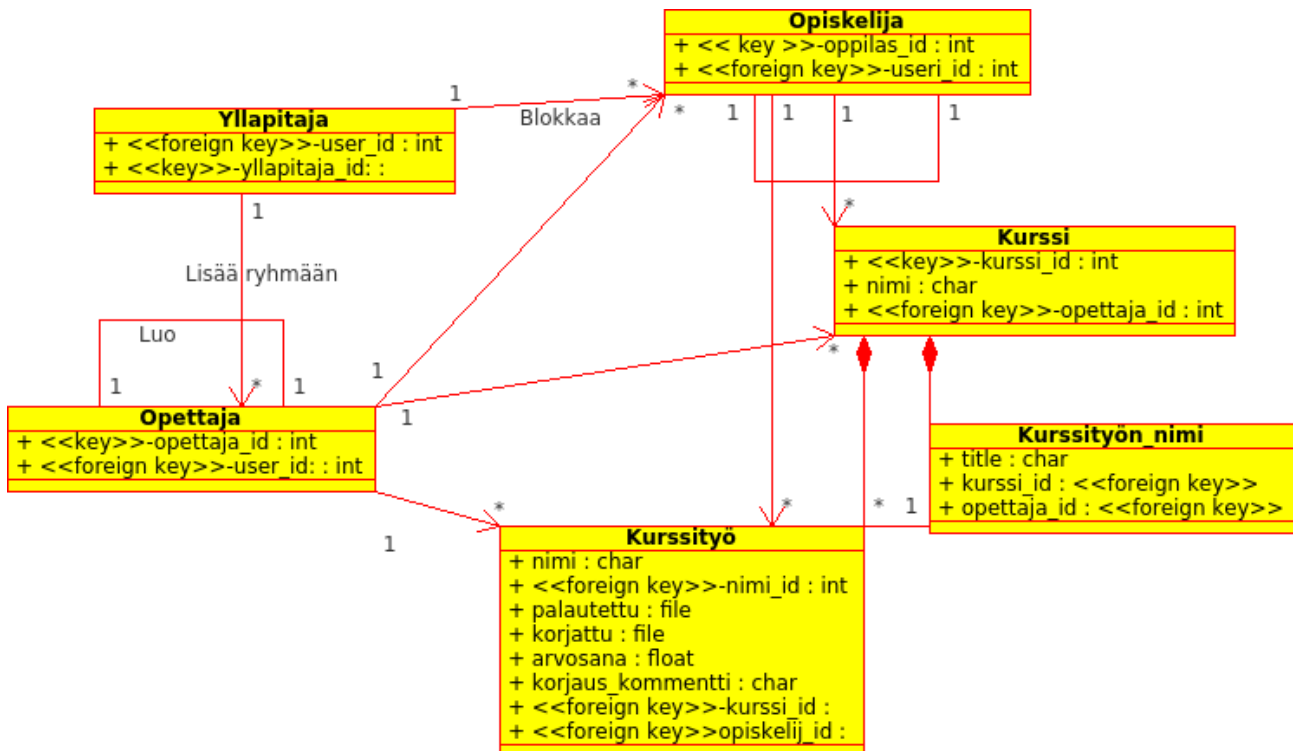


Avataan luokkakaavion käsitettä 'kurssityö' hieman tarkemmin oheisessa taulukossa.

Attribuutti	Arvojoukko	Kuvailu
kurssityon_nimi ref	Viite kurssityon_nimi	Kurssityön otsikko
Palautettu	tiedosto	Opiskelijan palauttama tiedosto
Korjattu	Tiedosto	Opettajan palauttama tiedosta
kurssi_id ref	Viite kurssiin	Kurssi johon kurssityö kuuluu
oppilas_id ref	Viite oppilaaseen	Oppilas

Luokkien välisiä suhteita on selitetty tarkemmin seuraavassa luvussa.

## 4 Relaatietietokantakaavio



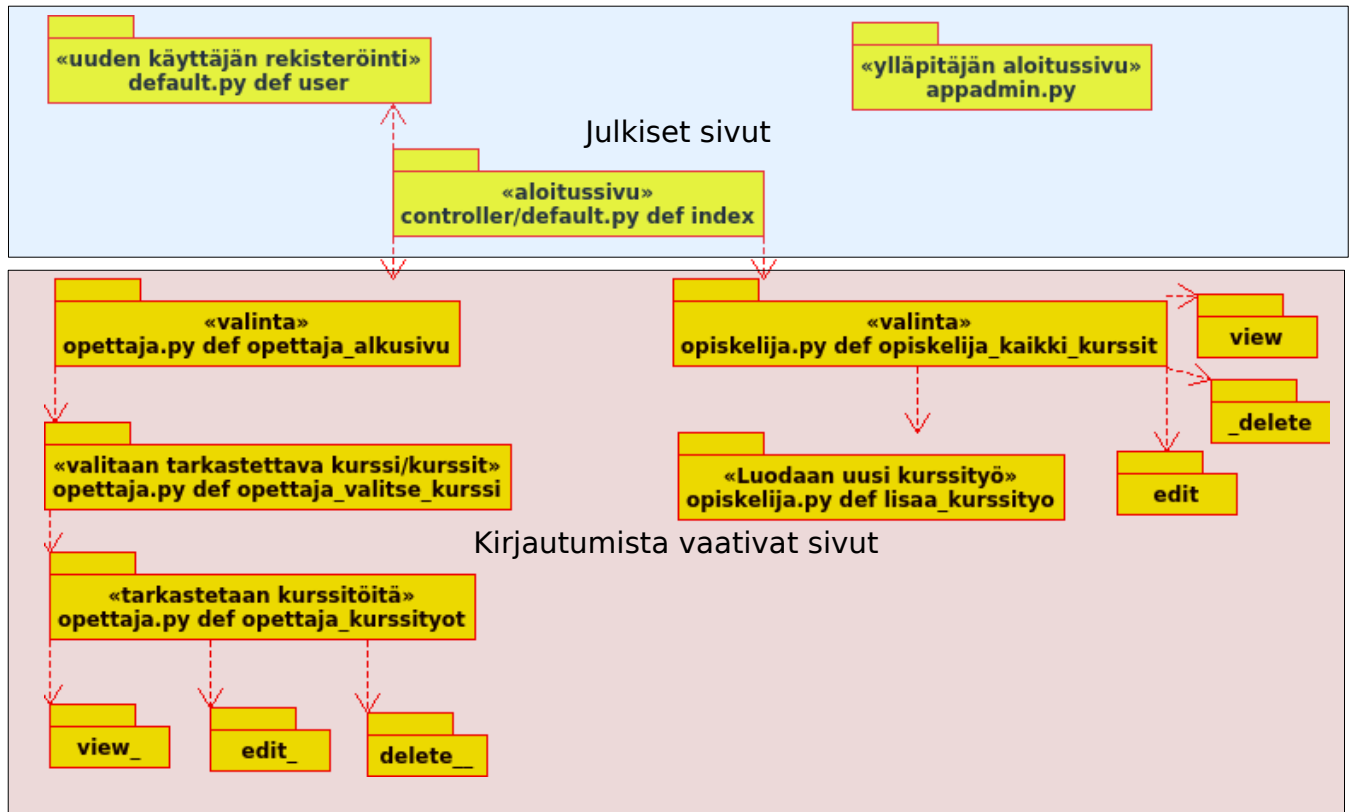
Yllä on esitetty kurssityöjärjestelmän relaatiotietokantakaavio. Ylläpitäjän rooli on siirtää tai luoda jäseniä ryhmään 'opettaja' ja blokata huonosti käyttäytyviä opiskelijoita tai opettajia. Järjestelmään kirjautuva uusi käyttäjä on oletusarvoisesti jäsen ryhmässä 'opiskelija'. Opettaja voi luoda kursseja ja kurssitoita. Oppilas voi palauttaa kurssitoita (palautettu) ja katsella tietoja omista kurssitoistaan ja kursseistaan. Opettaja antaa korjatut kurssityöt takaisin (korjattu) ja antaa niistä kommentit (korjaus\_kommentti) ja arvosanat (arvosana).

## 5 Järjestelmän yleisrakenne

Tietokantasovellusta tehdessä on noudatettu MVC(model-view-controller)-mallia. Kontrollerit, näkymät ja mallit sijaitsevat hakemistoissa controllers, views- ja models. Tietokantahaut löytyvät hakemistosta modules. Opettajaan liittyvät kontrollerit ovat tiedostossa controllers/opettaja.py ja opiskelijaan liittyvät tiedostossa controllers/opiskelija. Vastaavasti opiskelijan ja opettajan näkymät ovat tiedoistoissa view/opiskelija.py ja views/opettaja.py. Työssä on vain yksi tietokantatiedosta models/db.py. Opiskelija ja opettajan tietokantahaut ovat tiedoistoissa modules/opiskelija\_haut.py ja modules/opettaja\_haut.py. Kurssihin liittyvät tietokantahaut sijaitsevat tiedostossa modules/kurssi\_haut.py.

Ylläpitoon kirjaudutaan web2py admin tunnukseella ja tietokantaa hallinnoidaan web2py:ssä olevan valmiin ylläpitäjän käyttöliittymän avulla.

## 6 Käyttöliittymä ja järjestelmän komponentit



## 7 Asennustiedot

Sovellus käynnistetään etänä Heroku-pilvessä esim. Firefox tai Chrome selaimella. Käyttäjän ei tarvitse tehdä muuta (ks. Seuraava kappale sovelluksen käynnistämisestä)

## 8 Käynnistys- / käyttöohje

Sovelluksen www osoite on <http://mokaukon.users.cs.helsinki.fi/esittelysivu.html>

Sovellus käynnistyy osoitteesta: [kurssihallinta](#)

kirjaudu username 'olli' password 'olli' -> oppilaan näkymät

kirjaudu username 'onerva' password 'onerva' -> opettajan näkymät

Pääkäyttäjän sivu:[Ylläpitäjä](#)

Ylläpitäjän salasana test

Oppilas voi lisätä kurssitöitä itselleen ja ladata niihin tiedostoja.

Opettaja voi lisätä itselleen kursseja tai kurssitöiden otsikoita. Toisaalta hän voi selata oppilaiden palauttamia töitä, ladata niitä itselleen, ladata korjatun työn järjestelmään ja antaa arvosanoja.

## 9 Testaus, tunnetut bugit ja puutteet & jatkokehitysideat

Tilastosivut jäivät tekemättä, esimerkiksi opettaja tai oppilas voisi tarkastella erilaisia keskiarvoja. Samoin töihin liittyvät deadlinet jäivät tekemättä. Lisäksi jotkut sql-tauluihin liittyvät lisäykset jäivät huonosti tarkastetuiksi. Nyt esimerkiksi opettaja voi lisätä kurssitöiden otsikoita muiden opettajien kursseille. Tämön tekeminen web2py ympäristössä ei ole mitenkään helppoa.

## 10

## Omat kokemukset

web2py oli aluksi helppokäyttöinen, mutta jotkut lomakkeisiin liittyvät rajoitukset ja tarkistukset olivat yllättävän työläitä. Myös se että web2py on vasta muutaman vuoden vanha aiheuttaa sen että dokumentaatio ei ole kaikilta osin niin kattavaa kuin olisin toivonut.

### Muu dokumentaatio

SQL taulut ovat osoitteessa

<https://github.com/paapu68/TKsovellus/blob/master/sql/sql.log>