

# Dokumentaatio

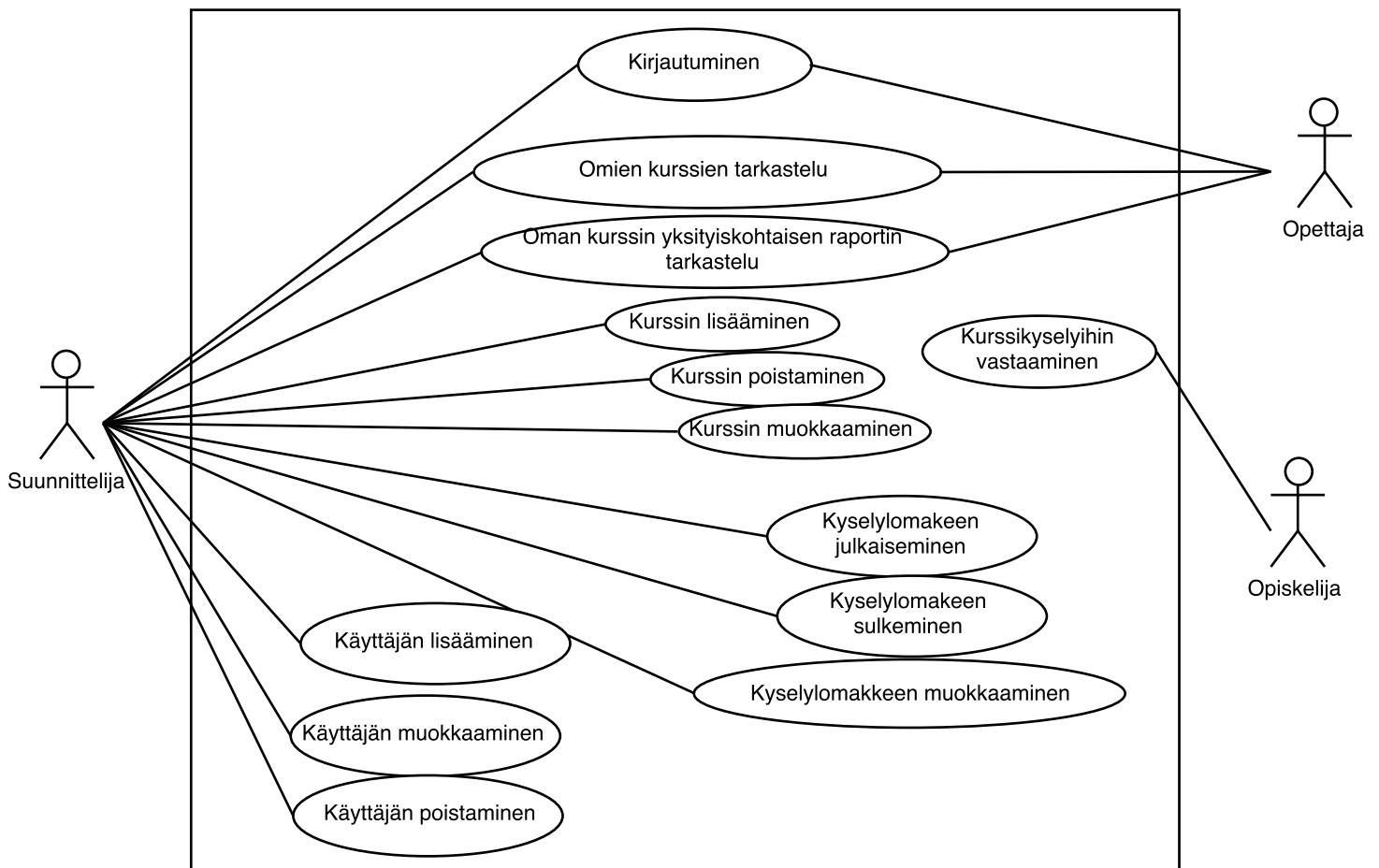
## Johdanto

Ideana on tuottaa järjestelmä, jonka avulla voidaan toteuttaa kurssikyselyjä opiskelijoille. Useammasta kurssista pitää voida tehdä kurssikysely. Vastaukset voivat olla tekstiä tai Likert- asteikollisia numeroarvoja. Kyselyihin pääsee vastaamaan ilman kirjautumista, mutta niiden tuloksia pääsee katsomaan vain jos on kirjautunut ja suunnittelija on lisännyt käyttäjän kurssin opettajaksi.

Jokaisella kurssilla voi olla kerrallaan yksi kysely käynnissä. Kyselylomaketta voi muokata vaikka siihen olisi jo ehditty vastaamaan. Kyselylomakkeen voi julkaista tai poistaa näkyvistä sivulta. Sen tiedot kuitenkin säilyvät ja sen voi palauttaa.

Järjestelmän toteutuskielenä käytetään PHP:tä ja se toteutetaan HY:n tietojenkäsittelytieteen laitoksen users-palvelimelle. Tietokantana käytetään PostgreSQL-tietokantapalvelinta. Käyttäjän selaimelle kohdistuvia vaatimuksia päivitetään kehityksen edetessä.

## Käyttötapaukset



# Käyttäjärühmät

## Opiskelija

Opiskelijalla tarkoitetaan henkilöä, joka on opiskellut jollakin laitoksen kursseilla ja saanut linkin järjestelmän sivustolle, jossa hän voi sitten vastata oman kurssinsa kurssikyselyyn.

## Opettaja

Opettaja luennoi yhtä tai useampaa kurssia laitoksella.

## Suunnittelija

Suunnittelija on jonkin laitoksen henkilökuntaa ja vastaa kurssien järjestämisestä.

# Käyttötapauskuvaukset

## Opiskelijan käyttötapauskuvat

### Kurssikyselyihin vastaaminen:

Opiskelija (käytännössä kuka tahansa) voi käydä järjestelmän sivustolla valitsemassa kurssin ja vastaamalla siihen liittyvään kurssikyselyyn.

## Opettajan käyttötapauskuvat

### Kirjautuminen:

Opettaja kirjautuu suunnittelijan luomilla tunnuksilla. Opettaja ei voi itse muuttaa omia tunnuksia.

### Omien kurssien tarkastelu:

Opettaja näkee kurssit listassa ja pääsee tarkastelemaan raportteja, jonkun suunnittelijan on ensin täytynyt lisätä opettaja kusiin yhteyteen.

### Oman kurssin yksityiskohtaisen raportin tarkastelu:

Opettaja voi valita yhden kurssistaan ja järjestelmä näyttää raportin vastauksista kurssia koskeneeseen kyselyyn. Tässä raportissa on mukana muunmuassa kaikki sanalliset vastaukset.

## Suunnittelijan käyttötapauskuvat

### Kirjautuminen

### Käyttäjän lisääminen:

Uudet käyttäjät lisätään järjestelmään, heille annetaan salasana, nimi ja käyttöoikeus, joka voi olla joko opettajan käyttöoikeus, tai suunnittelijan käyttöoikeus.

### Käyttäjän muokkaaminen:

Käyttäjälle voidaan antaa uusi käyttäjänimi tai salasana, tai hänen käyttöoikeutensa voidaan muuttaa.

### Käyttäjän poistaminen:

Käyttäjän tiedot poistetaan tietokannasta.

### Kurssin lisääminen, kurssin poistaminen:

Suunnittelija näkee kaikki kurssit listana ja voi lisätä ja poistaa kursseja.

### Kurssin muokkaaminen:

Suunnittelija voi muokata esimerkiksi kurssin nimeä.

### Kyselylomakkeen muokkaaminen:

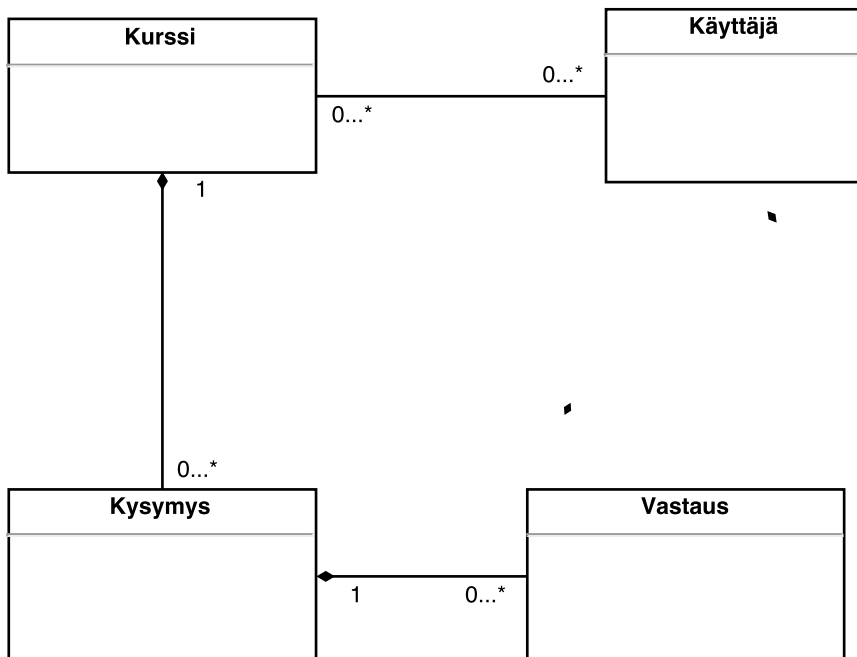
Suunnittelija voi lisätä lomakkeelle uusia kysymyksiä ja muokata ja poistaa vanhoja.

### Kyselylomakkeen julkaiseminen:

Kurssin kyselylomake valitaan julkaistavaksi ja opiskelijat voivat vastata siihen.

### Kyselylomakkeen sulkeminen:

## Järjestelmän tietosisältö



# Tietokohteet

## Kurssi

Attribuutti	Arvojoukko	Kuvailu
Nimi	Merkkijono max 100 merkkiä	Kurssin nimi
Laitos	Merkkijono max 100 merkkiä	kurssia pitävän laitoksen nimi
Julkaistu	Boolean	Onko kurssin lomake julkaistu
Suljettu	Boolean	ei tällähetkellä käytössä

Kurssilla voi olla nolla tai useita käyttäjiä, joilla on opettajan tai suunnittelijan käyttöoikeudet. Opettajilla voi olla pidettävänä useita kursseja tai ei yhtään. Kurssiin voi liittyä ei yhtään tai useampi kysymys, jotka sitten näytetään sivustolla kurssikohtaisella kyselylomakkeella.

## Kysymys

Attribuutti	Arvojoukko	Kuvailu
Nimi	Merkkijono max 100 merkkiä	käytetään kysymyksiä hallinnoitaessa ja muuttujanimenä
Kysymysteksti	Merkkijono max 500 merkkiä	Vastaajalle näkyvä kysymysteksti
Vastaustyyppi	merkkijono 20 merkkiä	kertoo onko vastaus sanallinen vai numeerinen (likert-asteikolla)

Kysymys voi sisältyä vain yhteen kurssiin, eli sen kyselylomakkeeseen. Kysymyksellä voi olla yksi, useampi tai ei yhtään vastausta.

## Vastaus

Attribuutti	Arvojoukko	Kuvailu
Vastaaja id	kokonaisluku	saman vastaajan vastausten tunnistamiseksi id
Vastausteksti	Merkkijono max 500 merkkiä	Vastaajan sanallinen vastaus
NumVastaus	kokonaisluku	numeerinen vastaus annetulla välillä
LikertVastaus	kokonaisluku	numeerinen vastaus välillä 1-5

Vastaus liittyy aina kysymykseen, yhdellä kysymyksellä voi olla useita vastauksia, mutta vastauksella vain yksi kysymys.

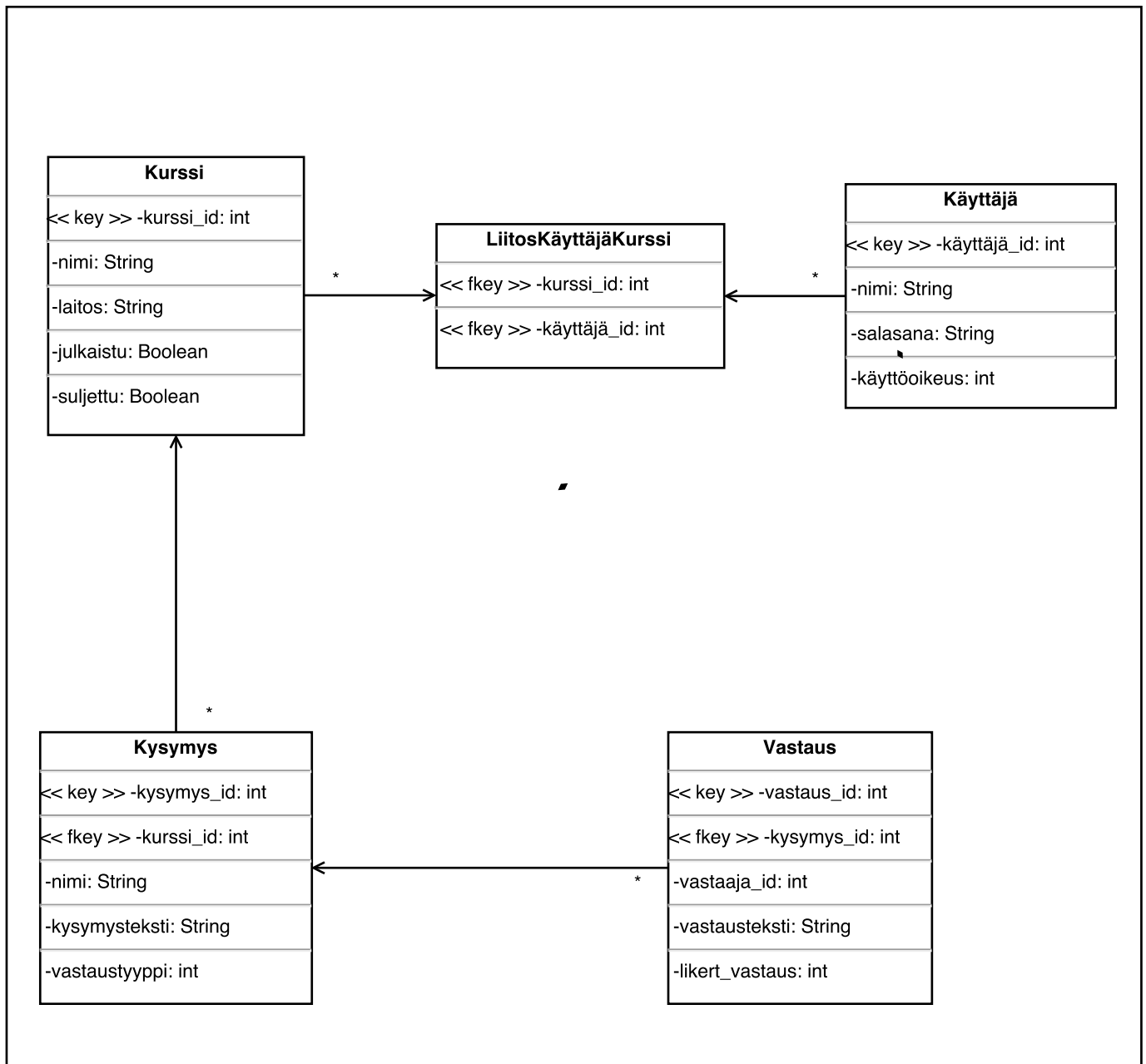
## Käyttäjä

Attribuutti	Arvojoukko	Kuvailu
Nimi	Merkkijono 100 merkkiä	Käyttäjän virallinen nimi
Salasana	merkkijono max 100 merkkiä	kirjautumiseen
Käyttöoikeus	kokonaisluku 0 tai 1	0=opettaja, 1 =suunnittelija

Käyttäjän ollessa opettaja tai suunnittelija voi hänellä olla pidettävänään useita kursseja, tai ei yhtäkään.

Käyttäjän oikeuksilla voi tehdä kaiken minkä opettajankin oikeuksilla, mutta ei toisinpäin.

## Relaatiotietokantakaavio



# Järjestelmän yleisrakenne

Tietokantasovellusta tehdessä on noudatettu MVC-mallia. Kontrollerit, näkymät ja mallit on sijoitettu hakemistoihin controllers, views- ja models. Käytetyt apukirjastot on sijoitettu hakemistoon lib. Sql -taulujen hallintaan tarkoitetut tiedostot ovat omassa kansiossaan. CSS -ja Javascript -tiedostot sijaitsevat assets -hakemiston alahakemistoissa. HTML- tiedostot sijaitsevat views-kansiossa ja on jaoteltu suurinpiirtein käyttötarkoituksen mukaan alahakemistoihin. Kaikki tiedostonimet on kirjoitettu pienellä. Lib- kansiota löytyy tärkeät base\_controller ja base\_model -luokat, joiden metodeja sovelluksen muut luokat perivät. (esimerkiksi kirjautumiseen ja validointiin liittyviä metodeja.)

Ohjelmakoodi on jaoteltu äsken mainituihin kontrollereihin, html -tiedostoihin ja tietokantoja käsitteleviin luokkiin, kaikki eri kansioissa. Järjestelmässä on neljä tietokantaluokkaa, neljä controlleriluokkaa ja yksi apuluokka (VastausKysymysPari) ei pitkäaikaiseen säilömiseen.

Kayttaja- luokka mahdollistaa käyttäjien kirjautumisen, hakemisen ja lisäämisen. Käyttäjillää on kahdenlaisia käyttölupia, numeroiksi koodattun 1 tai 0, jotka määrittävät onko käyttäjä "suunnittelija" vai "opettaja". Käyttäjäkontrolleri hallinnoi käyttäjien poistojen ja lisäysten lisäksi käyttäjän lisäämisen kurssille ja pois sieltä. Tämä tarkoittaa sitä, että lisätyt käyttäjät voivat lukea kurssin kyselyn tuloksia, lisäämättömät eivät. Järjestelmä käyttää istuntoa niin, että kirjautumisen jälkeen controllerit käyvät ennen kirjautumista vaativia toimintoja tarkastamassa kirjautuneen käyttäjän ja varmistavat, että tällä on sopiva käyttölupa.

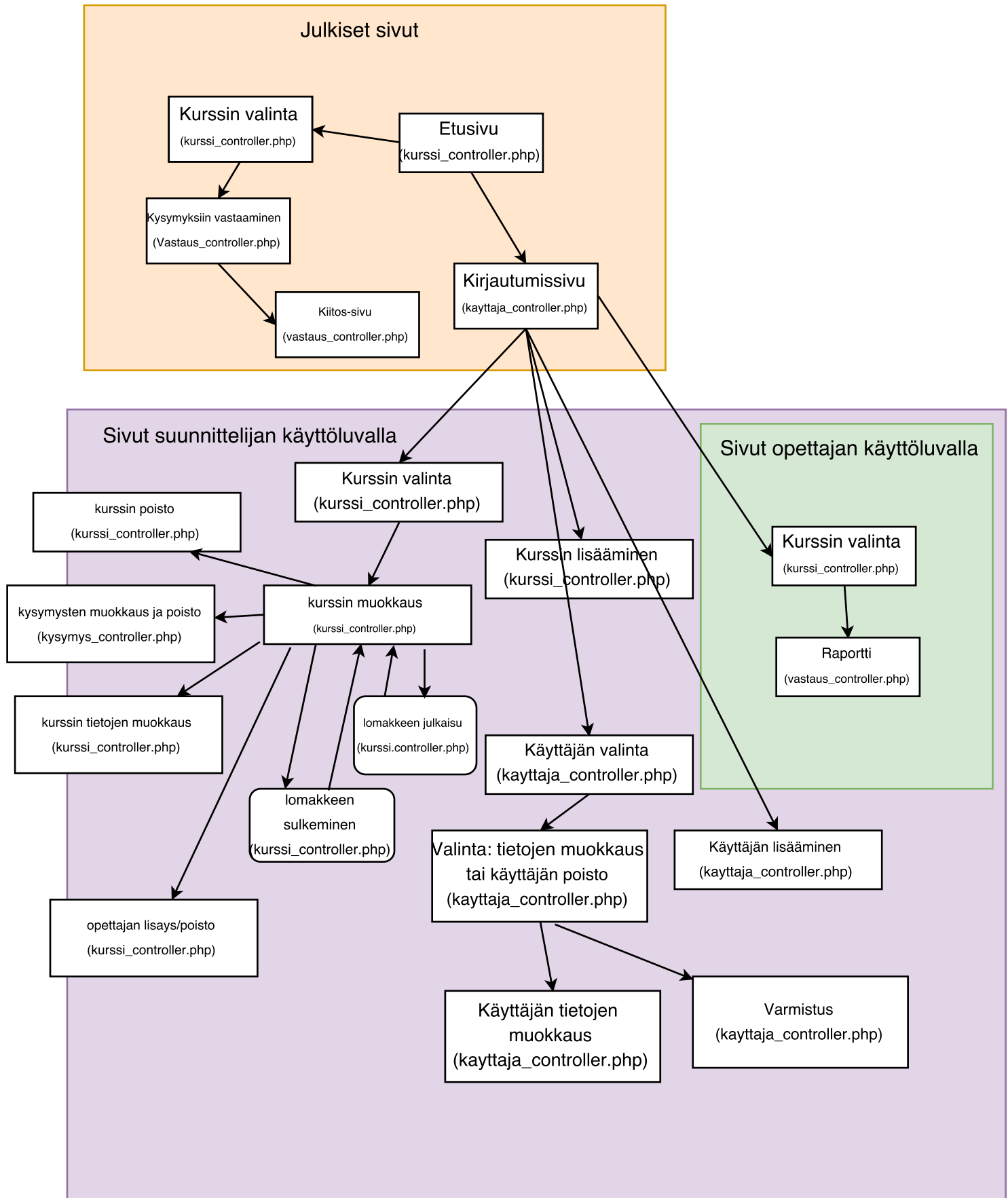
KurssiControllerin metodit keskittyvät kurssien hallinnointiin ja tässä tapauksessa myös kyselyjen sulkemiseen ja avaamiseen. Kysely voisi sinänsä olla synonyymi kurssille.

KysymysController hallinnoi lomakkeessa näytettäviä kysymyksiä. Silti luokassa on kohtuu paljon tavaraa johtuen lähinnä kysymysten muokkaus-näkymän toteuttamisesta (joka oli itselleni hiukan hankalaa). VastausController hallinnoi vastauksia ja tekee niistä raportin, kyselyn tulokset.

Controllerit ja mallit on tehty php-kielellä, näkynissä on käytetty html lisäksi esimerkiksi Twig- kirjastoa, joka mahdollistaa näkymän sisältöön vaikuttamisen kontrollirakenteilla. Html- näkymissä ei kuitenkaan ole käytetty php-koodia. Näkymien tekemisensi "kauniimmiksi" näppärästi on käytetty Bootstrap frameworkkia, valmiina käytetyssä tsoha-githubrepossa, jonka päälle sovellus on rakennettu.

# Käyttöliittymä ja järjestelmän komponentit

Sivuilla on navigaatiopaljikki, jonka avulla mistä vain pääsee etusivulle ja kirjautumaan. Etusivulla näytetään jokseenkin eri sisältöä käyttäjän luvasta riippuen.



## Asennustiedot

Asennuksessa kansion config tiedostossa database.php voidaan määrittää käytetäänkö PostgreSQL:ää (niinkuin nyt), vai MySQL:ää. Tiedostossa on myös ohjeet salasanan ja käyttäjätunnuksen asettamiseen ja ne asetetaan tähän tiedostoon.

## Käynnistys- / käyttöohje

Järjestelmän sivulle pääsee osoitteesta <http://jarmokal.users.cs.helsinki.fi/poll/>. Kirjautuakseen heikommilla "opettajan" tunnuksilla voi salasanaksi antaa "pizzaisyummy" ja käyttäjänimeksi "Dolan". Käyttäjänimellä "Gooby" ja samalla salasanalla "pizzaisyummy" pääsee käyttämään järjestelmää vahvemmillä "suunnittelijan" käyttöoikeuksilla.

## Testaus, tunnetut bugit ja puutteet & jatkokehitysideat

Järjestelmässä ei tällä hetkellä ole yksikkötestejä. Tämä hankaloittaa refaktorointia ja muutosten tekoa, eli on suuri puute. Testaus on tähän asti tehty ajamalla tiedostot palvelimelle ja kokeilemalla itse miten sivusto toimii ja katsomalla virheraportteja. Järjestelmä ehti kuitenkin jo tämän lyhyen projektin aikana hiukan kasvaa joten jos kehittäjä jatkettaisiin olisi jonkinlainen automaattinen testaus käytännössä pakollista.

HTML-näkymien tiedostot eivät ole "tyylikkäämmiin" mahdollisesti jaoteltu ja käytetty. Tiedostoissa on jonkin verran toistoa jota saisi varmasti pois esimerkiksi makrojen avulla.

Tällä hetkellä, kun käyttäjä lisää uusia kysymyksiä kysymyslomakkeelle, ei hän voi vaihtaa kysymysten paikkaa muuten kuin kikkailemalla päivittämällä kysymyksiä, mikä ei tietysti ole toivottavaa. Kunnollinen kysymysten järjestely tarvittaisiin.

Koskaan ei ole liikaa tilastoja. Suunnittelijalle voisi antaa työkalun, jolla hakea tietoa järjestelmästä, kuitenkin "suojatulla tavalla" eli niin, ettei järjestelmä kaadu. Vastaajien ei tarvitse kirjautua vastatakseen kysymyksiin, mutta saman henkilön saman kurssin vastaukset voidaan yksilöidä. Jos vastaajienkin olisi kirjauduttava, saataisiin vielä tarkempaa tietoa henkilöiden mielipiteistä (tai siitä miten vastasivat kyselyyn.)

## Omat kokemukset

Linkkien kanssa väkslääminen ja sen varmistaminen, että homma vielä toimii jos jotain muuttuu. Pitäisi tehdä yksikkötestejä tai testata jollain robotilla että homma aina toimii, nyt kävi vähän niin, että aluksi tuli tehtyä nopeasti vähän suunnittelematta ja huonosti, ja siitä on varmasti jäänyt vähän jälkiä. Sovelluksen siirto palvelimelle toimii näppärästi ja nopeasti. Users on toiminut varsin hyvin. Php tuntui aluksi vähän vieraalta, mutta toistolla se on alkanut tuntua kotoisemmalta. Valitsin vähän liian laajan aiheen, ja jouduin karsimaan toimintoja. Tälläkin hetkellä ja jälkikäteen on ehkä turhaa, että sovellus tukee kahta erilaista käyttöoikeutta, yhdenlainen kirjautunut käyttäjä olisi varmasti riittänyt. Mutta tulipahan säädettyä. Koodasin sublimella ja virheitä tuli tietysti enemmän kuin jos vääntäisi javaa netbeansilla yms jolloin ei tarvitsisi palata muuttamaan puuttuvia puolipisteitä myöhemmin. Kuitenkin tuolliset virheet löytyi hyvin nopeasti että ei sinänsä ollut mahdotonta.