

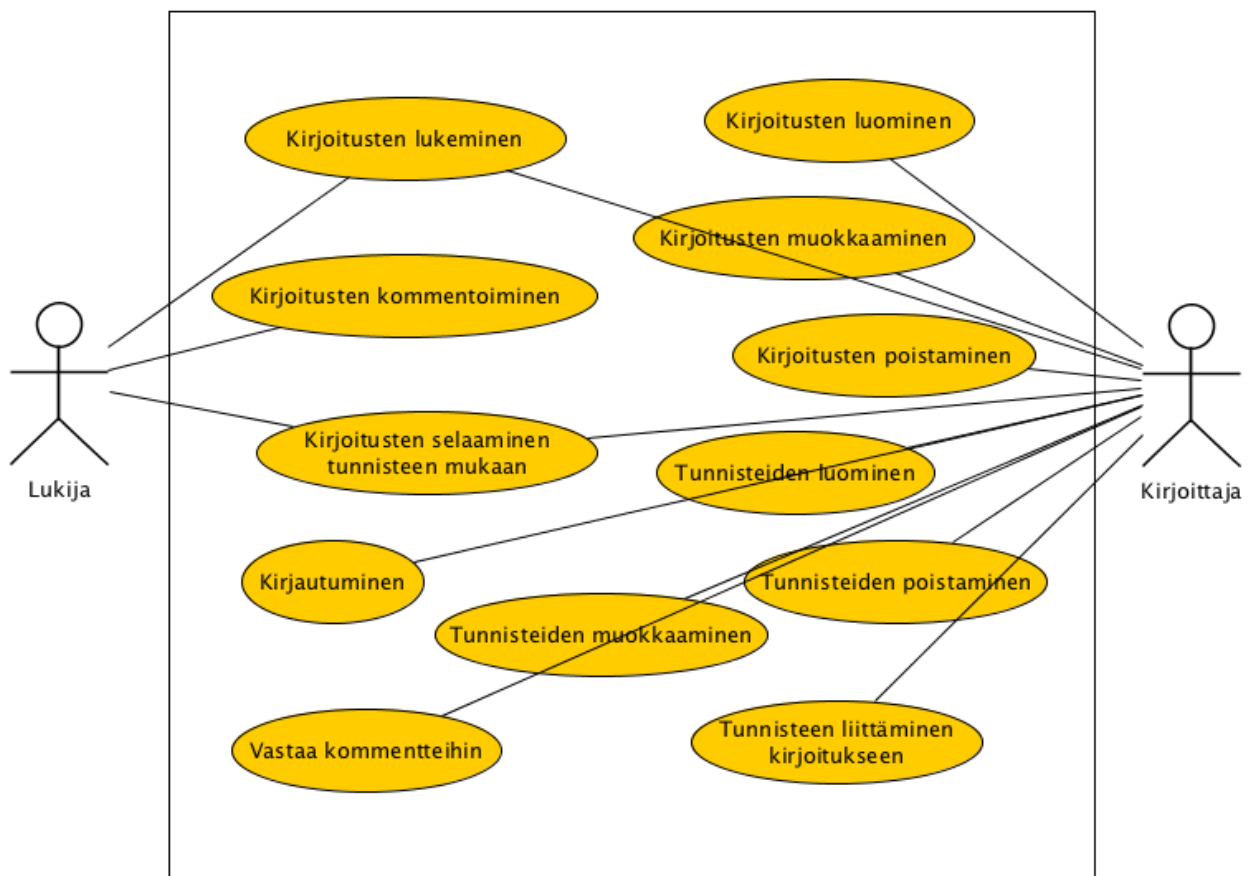
## 1. Johdanto

Tämä sovellus on blogi, eli julkinen päiväkirja. Blogin avulla kirjoittajat voi julkaista kirjoituksiaan vaivattomasti verkossa ja tavoittaa maailmanlaajuisen lukijakunnan. Lukijat voivat seurata blogin päivittymistä sekä selata vanhoja kirjoituksia aihetunnisteiden avulla. Jokaisen kirjoituksen kommentointiosiossa on mahdollista käydä keskustelua muiden lukijoiden sekä blogin kirjoittajien kanssa.

Blogisovellus toteutetaan Helsingin yliopiston Tietojenkäsittelytieteen laitoksen users-palvelimella käyttäen PHP-ohjelmointikieltä ja PostgreSQL-tietokantaa.

## 2. Yleiskuvaus järjestelmästä

### Käyttötapauskaavio



## **Käyttäjärühmät**

### **Lukija**

Lukija on kuka tahansa, joka vierailee blogin internetosoitteessa.

### **Kirjoittaja**

Kirjoittaja on kirjautunut sisään blogisovellukseen.

## **Käyttötapauskuvaukset**

### **Lukijan käyttötapauskset**

Kirjoitusten lukeminen:

Kirjoitukset näkyvät blogin etusivulla laskevassa aikajärjestyksessä.

Kirjoitusten kommentoiminen:

Kirjoituksen lopussa on mahdollisuus jättää nimimerkillä varustettu kommentti.

Kirjoitusten selaaminen tunnisteiden mukaan:

Valitsemalla tunnisteiden, blogisovellus näyttää vain kirjoitukset, joihin blogin kirjoittajat ovat kyseisen tunnisteiden lisänneet.

### **Kirjoittajan käyttötapauskset**

Kirjoituksen luominen:

Kirjoittaja kirjoittaa ja lähettää tekstin.

Kirjoituksen muokkaaminen, kirjoituksen poistaminen

Kommenttiin vastaaminen:

Kirjoittaja voi valita kommentin ja vastata siihen.

Tunnisteiden luominen:

Kirjoittaja luo uusia tunnisteita.

Tunnisteiden liittäminen kirjoitukseen:

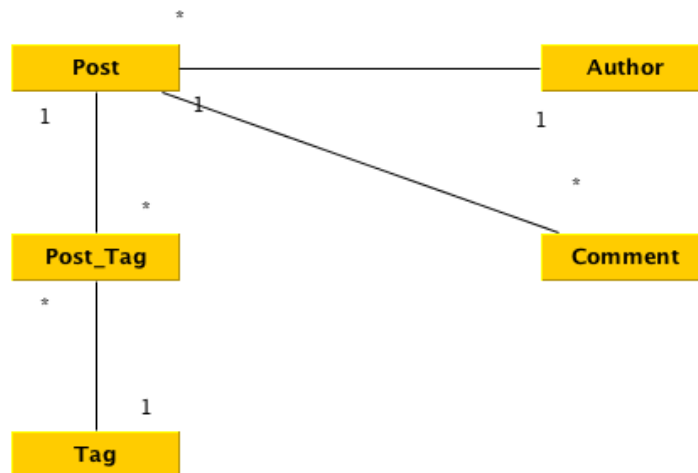
Kirjoitukseen voi liittää yhden tai useamman tunnisteiden.

Tunnisteiden muokkaaminen, tunnisteiden poistaminen

Kirjautuminen

### 3. Järjestelmän tietosisältö

#### Käsitekaavio



#### Tietokohteet

Tietokohde: Author

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max 50 merkkiä	Kirjoittajan nimi
password	Merkkijono, max 50 merkkiä	Salasana
image_url	Merkkijono, max 100 merkkiä	Profiilikuvan osoite
description	Merkkijono, max 100 merkkiä	Kuvaus kirjoittajasta

Author (*kirjoittaja*) on blogin kirjoittaja ja ylläpitäjä. Hän kirjautuu sisään palveluun.

Tietokohde: Post

Attribuutti	Arvojoukko	Kuvailu
header	Merkkijono, max 50 merkkiä	Otsikko
content	Merkkijono, max 1000 merkkiä	Sisältöteksti
created	aikaleima	Luotu
edited	aikaleima	Muokattu viimeksi

Post (*blogikirjoitus*) on blogin kirjoittajan tuottama sisältö. Kirjoitukseen liitetään tunniste.

Tietokohde: Comment

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max 50 merkkiä	Kommentoijan nimi
header	Merkkijono, max 50 merkkiä	Otsikko
content	Merkkijono, max 400 merkkiä	Sisältöteksti
created	aikaleima	Luotu
edited	aikaleima	Muokattu viimeksi

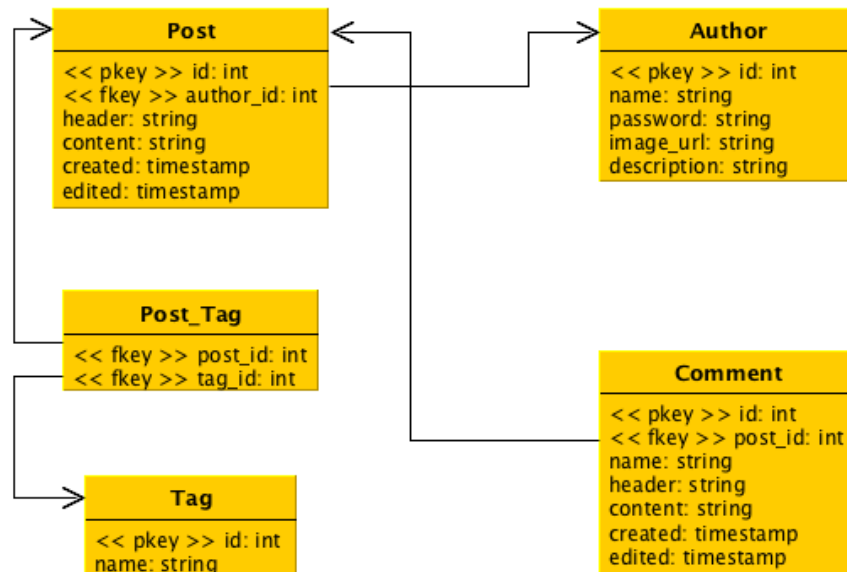
Comment (*kommentti*) on lukijan tietystä blogikirjoituksesta jättämä kommentti.

## Tietokohde: Tag

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max 50 merkkiä	Tunnisteen nimi

Tag (*tunniste*) helpottaa blogikirjoitusten selaamista, sillä lukija voi listata kaikki kirjoitukset, jotka on merkitty tietyllä tunnisteella.

## 4. Relaatiotietokantakaavio



## 5. Järjestelmän yleisrakenne

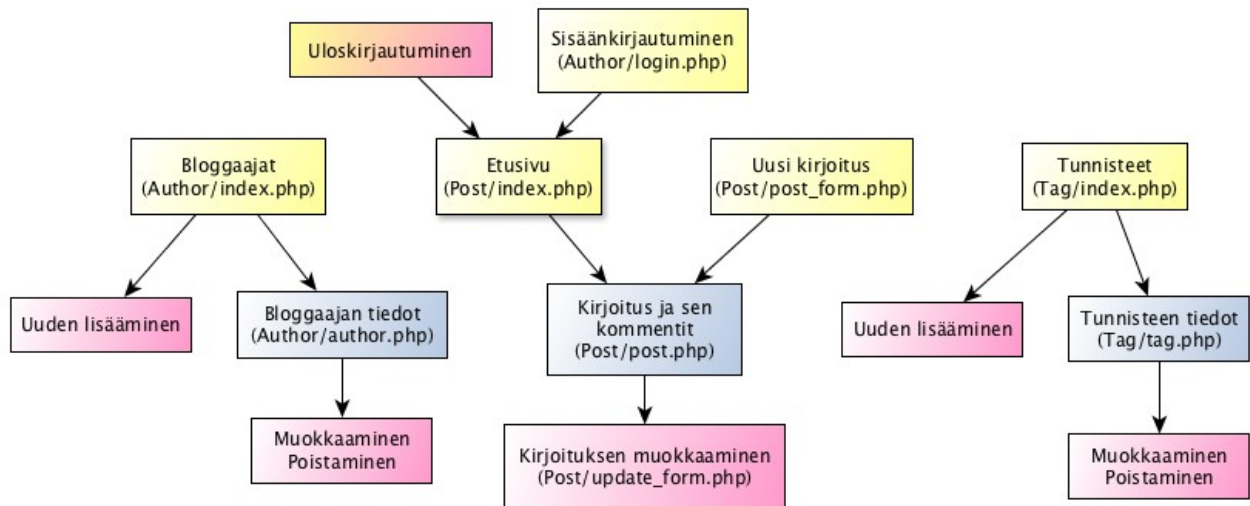
Järjestelmä noudattaa MVC-mallia, eli mallit (models), näkymät (views) ja kontrollerit (controllers) on eritetty omiin kansioihinsa. Asetukset löytyvät settings.php -tiedostosta. Kansiot on nimetty niin, että ne kuvaavat sisältöään, esimerkiksi apukirjastot löytyvät kansiota lib.

## 6. Käyttöliittymä ja järjestelmän komponentit

Kaikki näkymien dynaaminen sisältö renderöidään staattiseen base.html-sivupohjaan.

Käyttöliittymäkaaviossa keltaisiin näkymiin pääsee suoraan navigaatiopalkista. Siniset näkymät ovat julkisia ja vaaleanpunaisiin näkymiin pääsee vain kirjautunut käyttäjä.

Kaavioon on kirjoitettu jokaisen näkymän sijainti views-kansiossa, esim. Bloggaajat (Author/index.html). Jokaisen views-kansion alikansiolle on oma kontrolleri, joka huolehtii kyseisen kansion näkymistä. Author-kansion näkymistä huolehtii authors\_controller.php, Comment-kansiosta huolehtii comments\_controller.php ja niin edelleen.



## 7. Asennustiedot

Blogisovellus asennetaan kopioimalla sovelluksen tiedostot palvelimelle ja päivittämällä config-kansion tiedostoihin konfiguraatiot vastaamaan käyttämäsi ympäristöä.

## 8. Käynnistys- / käyttöohje

Blogisovellus on asennettuna osoitteessa <http://piiahart.users.cs.helsinki.fi/tsoha/>. Voit kirjautua sisään käyttäjätunnuksella "Vieras" ja salasanalla "Vieras".

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

Jatkossa sovelluksen rakennetta voisi selkeyttää entisestään ja toteuttaa jotkin ominaisuudet palveluina (service). Esimerkiksi tagien listaaminen olisi järkevämpää tehdä sillä tavalla. On myös tutkimisen arvoista, kuinka paljon näkymissä on vielä toisteistaisuutta.

## 10. Omat kokemukset

Websovelluksen teko oli innostavaa. Dynaamisten näkymien rakentaminen palkitsi vaivannäön nopeasti ja kontrollerien sekä muiden taustalla vaikuttavien näkymättömien rakenteiden kehittyminen oli jatkuvasti näkyvillä.

Olen käynyt aikaisemmin webpalvelinohjelmointi-kurssin, joten MVC-mallin mukaiset websovellukset ovat perusperiaatteiltaan olleet minulle tuttuja. Tällä kurssilla opin pitkäjänteistä työskentelyä opiskelijan mittakaavassa ison sovelluksen parissa ja tutustuin paremmin tietokannan käyttöön sovelluslogiikan ja tietokannan pystyttämisen kannalta.

Tietokantasovellus on kurssina opastanut tasaisen rauhalliseen viikkotyöskentelyyn ja jakamaan

työurakkaa järkeviin osiin ja kokonaisuuksiin. Harjoitustyöni aihe, blogi, on minulle hyvin tuttu arjestani ja voin kuvitella, mitä olisi käynyt, jos olisin tehnyt oman blogisovellukseni ominpäin: olisin luultavasti yrittänyt saada kaiken valmiiksi kerralla, enkä olisi malttanut toteuttaa ominaisuuksia vähän kerrallaan. Viikon-parin jälkeen en olisi jaksanut jatkaa sovellukseni parissa.

## **11. Muu dokumentaatio**