

# LahjaSovellus

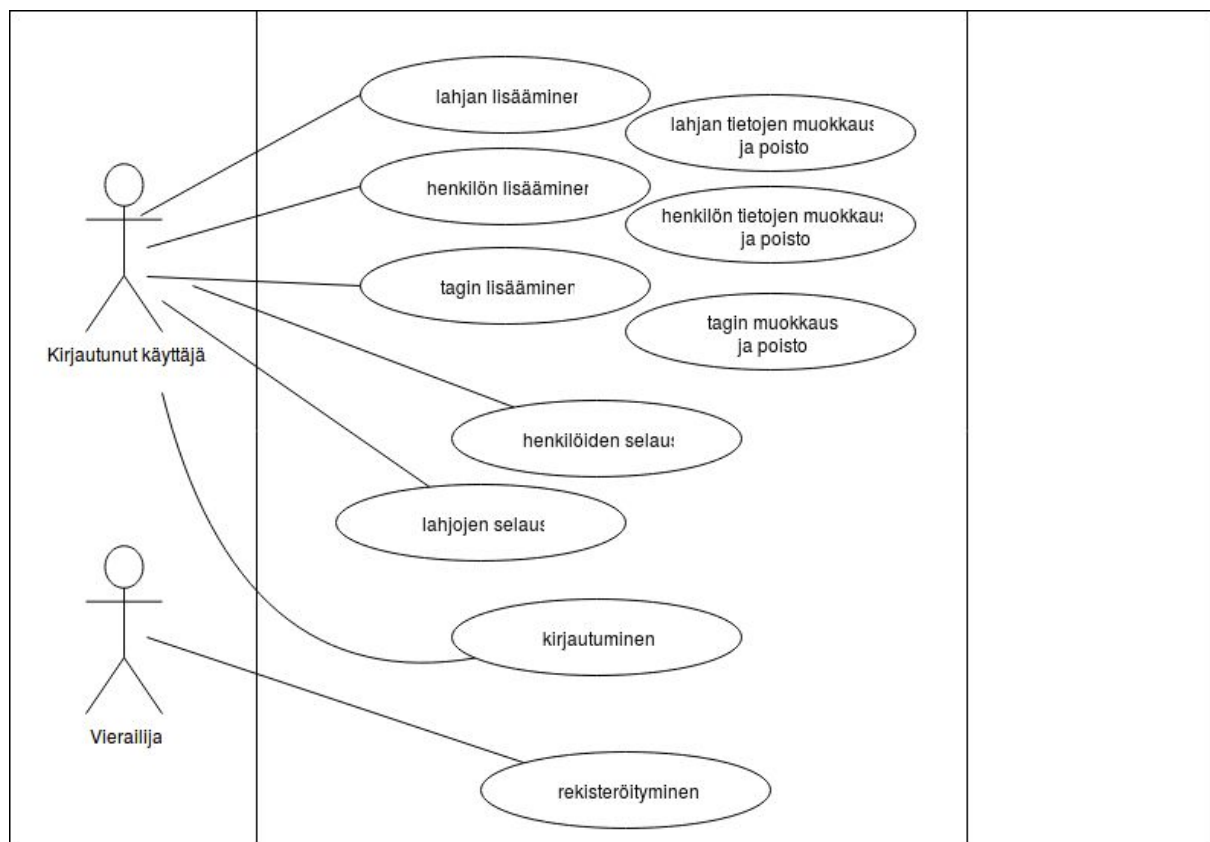
## 1. Johdanto

Harjoitustyön tarkoituksena on luoda sovellus lahjojen suunnittelun hallintaan. Ideana on, että käyttäjä voi sovelluksessa lisätä siihen tuntemiaan henkilöitä, ja yhdistää heihin esimerkiksi syntymäpäivä- tai joululahjoja. Lahjalle asetetaan luontivaiheessa tageja, sekä muut tarpeelliset tiedot. Käyttäjä voi uusien kohteiden luomisen jälkeen tarkastella suunniteltuja lahjaideoita, sekä muokata ja poistaa niitä. Henkilöiden poisto onnistuu vain, jos henkilöön ei liity mitään lahjoja. Lahjojen saajat, lahjat ja tagit tallennetaan tietokantaan käyttäjäkohtaisesti.

Sovellus on toteutettu PHP:llä ja se sijaitsee laitoksen users-palvelimella. Sovellus käyttää tietokantanaan PostgreSQL-tietokantaa.

## 2. Käyttötapaukset

### 2.1 Käyttötapauskaavio



## 2.2 Käyttäjäryhmät

### *Kirjautunut käyttäjä*

Kirjautunut käyttäjä on henkilö, jolla on olemassa rekisteröity tunnus sovellukseen ja joka on kirjautunut sisään tunnuksillaan. Kirjautunut käyttäjä pystyy tarkastelemaan sovelluksen kaikkia ominaisuuksia.

### *Vierailija*

Vierailija on käyttäjä, jolla ei ole tunnusta sovellukseen. Vierailija pystyy tarkastelemaan etusivun lisäksi ainoastaan rekisteröitymis- ja kirjautumissivuja.

## 2.3 Käyttötapaukset

### 2.3.1 Kirjautuneen käyttäjän käyttötapaukset

#### *Kirjautuminen*

Käyttäjä voi kirjautua sisään aiemmin luodulla tunnuksella, jonka jälkeen käyttäjällä on kaikki sovelluksen toiminnallisuudet käytössä. Sisäänkirjautunut käyttäjä voi myös kirjautua ulos, jolloin käyttäjä näkee sovelluksen jälleen vierailijana.

#### *Henkilöiden tarkastelu*

Käyttäjä voi tarkastella lahjojen saajiksi lisättyjä henkilöitä listassa, sekä tarkastella yksittäisen henkilön tietoja ja lahjoja.

#### *Henkilön lisääminen*

Käyttäjä voi lisätä uuden henkilön lahjojen saajaksi. Henkilölle annetaan lisäyksen yhteydessä nimen lisäksi syntymäpäivä, sekä vapaamuotoinen kuvaus.

#### *Henkilön tietojen muokkaus*

Käyttäjä voi muokata olemassa olevan henkilön tietoja.

#### *Henkilön poistaminen*

Käyttäjä voi poistaa henkilön, jos henkilöön ei liity yhtäkään lahjaa.

#### *Lahjojen tarkastelu*

Käyttäjä voi tarkastella kaikkia sovellukseen tallentamia lahjojaan. Yksittäistä lahjaa tarkasteltaessa lahjasta näkyy enemmän tietoja.

#### *Lahjan lisääminen*

Käyttäjä voi lisätä uuden lahjan, joka liittyy yhteen sovelluksessa olemassa olevaan henkilöön. Lahjalle annetaan lisäyksen yhteydessä nimen lisäksi kuvaus sekä mahdollisia tageja. Lahjalle annetaan myös status sen mukaan, onko lahja jo annettu vai ei.

#### *Lahjan muokkaus*

Käyttäjä voi muokata olemassa olevan lahjan tietoja.

#### *Lahjan poisto*

Käyttäjä voi halutessaan poistaa lahjan.

#### *Tagin tarkastelu*

Käyttäjä voi tarkastella kaikkia tageja. Yksittäistä tagia tarkastelemalla käyttäjälle listataan myös kaikki lahjat, joihin kyseinen tagi liittyy.

#### *Tagien lisääminen*

Käyttäjä voi lisätä uuden tagin. Tagille ei syötetä nimen lisäksi muita tietoja.

#### *Tagin muokkaus*

Käyttäjä voi muokata olemassa olevan tagin nimeä. Muutos vaikuttaa kaikkiin lahjoihin, joihin tagi liittyy.

#### *Tagien poistaminen*

Tagin voi poistaa, jos siihen ei liity yhtäkään lahjaa.

### 2.3.2 Vierailijan käyttötapaukset

#### *Etusivun tarkastelu*

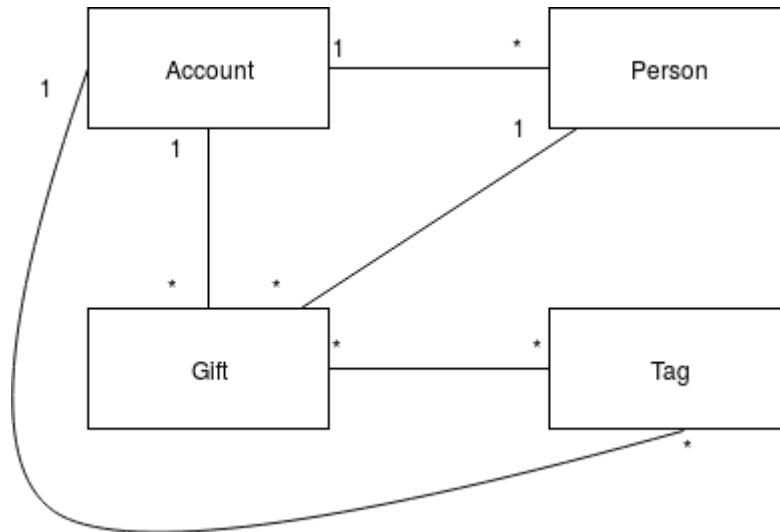
Käyttäjä voi tarkastella etusivua, mutta ei muiden kirjautuneiden käyttäjien tallentamia lahjaideoita.

#### *Rekisteröityminen ja kirjautuminen*

Vierailija voi rekisteröityä sovellukseen, jonka jälkeen hänelle avautuvat kirjautuneen käyttäjän toiminnallisuudet.

### 3. Järjestelmän tietosisältö

#### 3.1 Käsitekaavio



#### 3.2 Tietokohteet

##### *Account*

Attribuutti	Arvojoukko	Kuvailu
Username	Merkkijono, 5-30 merkkiä	Käyttäjän nimimerkki
Password	Merkkijono, 8-30 merkkiä	Käyttäjän salasana

Sovelluksen käyttäjästä tallennetaan käyttäjänimi ja salasana.

##### *Person*

Attribuutti	Arvojoukko	Kuvailu
Name	Merkkijono, max 50 merkkiä	Henkilön nimi
Birthday	Päivämäärä	Henkilön syntymäpäivä
Description	Merkkijono, max 500 merkkiä	Vapaamuotoinen kuvaus henkilöstä

Henkilöitä eli lahjan saajia voidaan lisätä tietokantaan, kun käyttäjätunnus on rekisteröity ja käyttäjä on kirjautunut sovellukseen. Henkilö liittyy yhteen käyttäjään, mutta käyttäjään voi liittyä useita henkilöitä.

### *Gift*

Attribuutti	Arvojoukko	Kuvailu
Name	Merkkijono, max 100 merkkiä	Lahjan nimi
Status	Boolean	Lahjan status, annettu=true ja ei annettu=false
Added	Päivämäärä	Päivämäärä, jolloin lahja on lisätty sovellukseen
Description	Merkkijono, max 500 merkkiä	Vapaamuotoinen kuvaus lahjasta

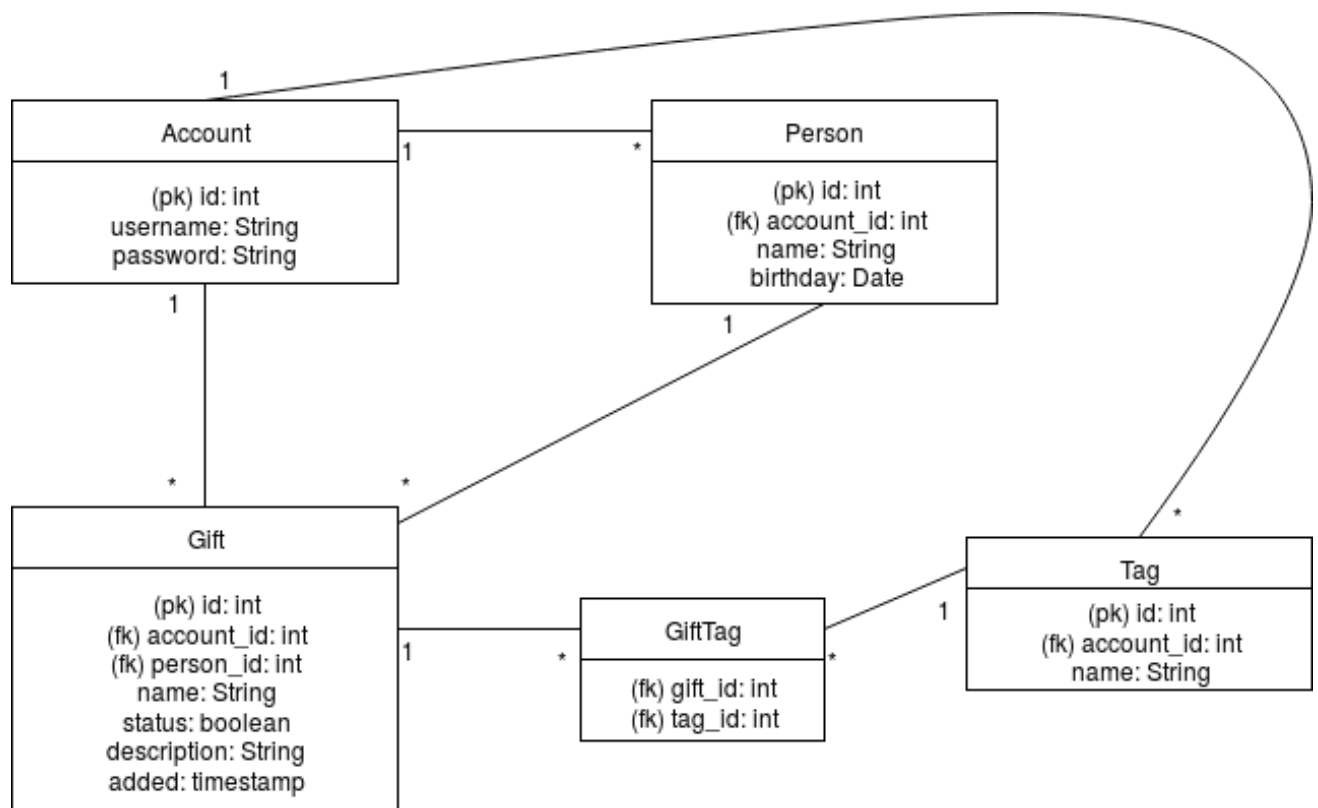
Lahjoja voi lisätä tietokantaan, kun käyttäjätunnus on rekisteröity ja sovelluksessa on lisättynä vähintään yksi käyttäjä. Henkilön tapaan lahja liittyy vain yhteen käyttäjään, mutta käyttäjällä voi olla lisättynä monia lahjoja. Lahja liittyy yhteen henkilöön eli lahjan saajaan, mutta henkilöön voi liittyä useampia lahjoja.

### *Tag*

Attribuutti	Arvojoukko	Kuvailu
Name	Merkkijono, max 50 merkkiä	Tagin nimi

Tagit luodaan käyttäjäkohtaisesti, eli yksi tagi liittyy yhteen käyttäjään mutta käyttäjään voi liittyä useita tageja. Yksi tagi voi liittyä moneen lahjaan ja lahjalla voi olla monia tageja.

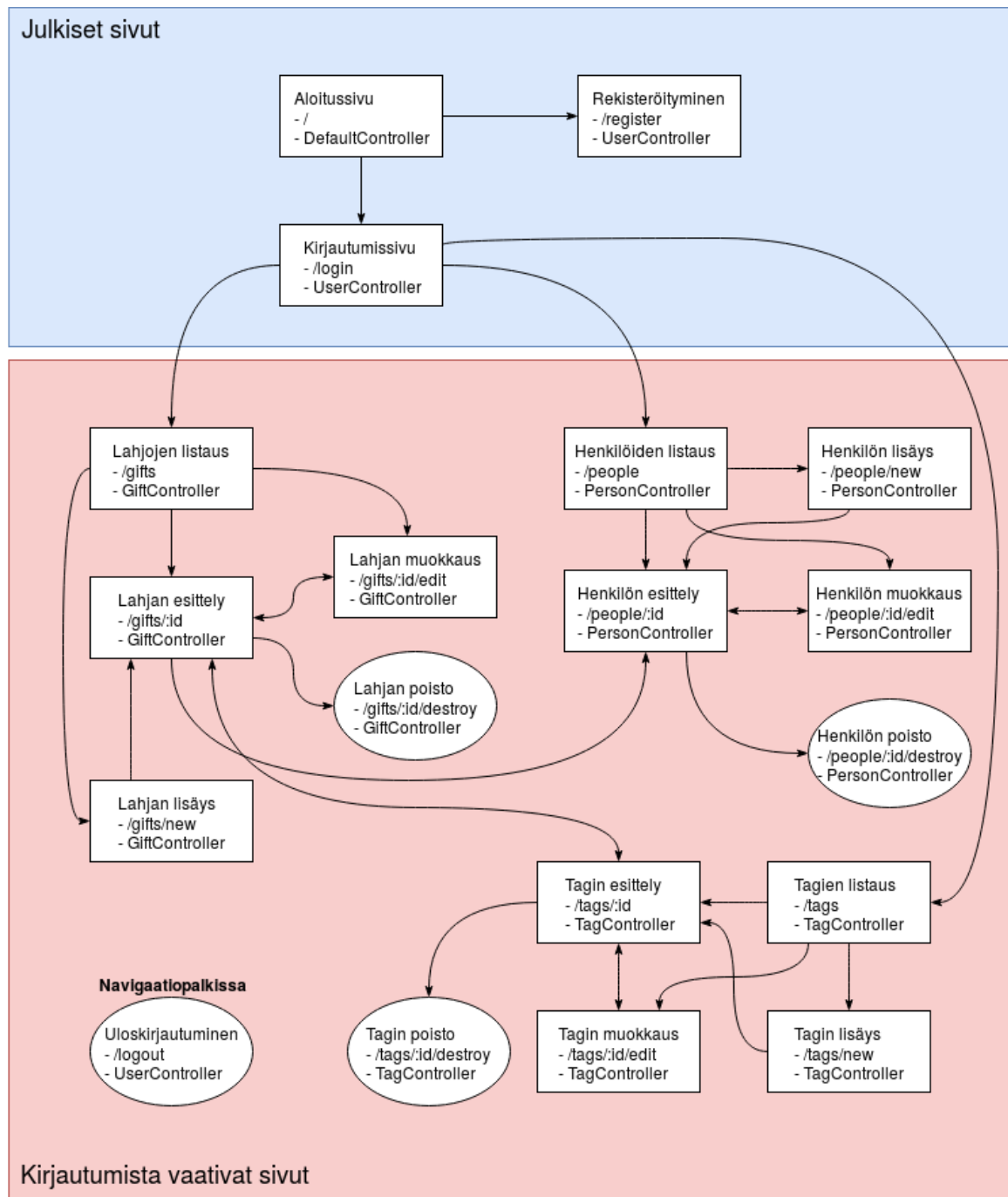
## 4. Relaatiotietokantakaavio



## 5. Järjestelmän yleisrakenne

Sovellus on toteutettu MVC-mallin mukaisesti. Kontrolleriluokat on jaettu vastuualueen mukaan (käyttäjä, lahjat, saajat, tagit) ja ne löytyvät kansioista controllers. Vastaavat mallit löytyvät kansioista models ja html-sivuina toteutetut näkymät kansioista views. Tietokannan luovat ja testidatan lisäävät kyselyt löytyvät sql-hakemistosta.

## 6. Käyttöliittymä ja järjestelmän komponentit



## 7. Asennustiedot

Sovelluksen asentamiseen on ensiksi ladattava tiedostot tästä reposta kloonamalla ne omalle koneelle. Sovellus voidaan pystyttää alustalle, jossa on otettu käyttöön PostgreSQL ja PHP-tuki. Tiedostot kopioidaan palvelimen netissä näkyvään hakemistoon, joka users-palvelimen tapauksessa on htdocs-hakemisto. Users-palvelimen toimintaa varten muokataan vielä config-hakemiston tiedostoa "environment.sh" vastaamaan omaa yliopiston käyttäjätunnusta ja haluttua projektikansion nimeä. Tämän jälkeen muutokset siirretään palvelimelle komentoriviltä komennolla "bash bootstrap.sh".

Tietokantataulut saa alustettua komentoriviltä komennolla "bash create\_tables.sh" ja testidatan lisättyä komennolla "bash add\_test\_data.sh". Mikäli sovellukseen tehdään muutoksia, saadaan ne siirrettyä palvelimelle komennolla "bash deploy.sh".

## 8. Käyttöohje

Sovellus löytyy osoitteesta <http://mihamaki.users.cs.helsinki.fi/tsoha/>. Kirjautumaton käyttäjä pääsee valmiissa sovelluksessa ainoastaan etusivulle, joten käyttäjän on rekisteröidyttävä sovellukseen päästäkseen käyttämään sen toimintoja.

*Testikäyttäjä*

Käyttäjänimi: UkkoNooa

Salasana: ukkonooa