

# Muistilista

## 1. Johdanto

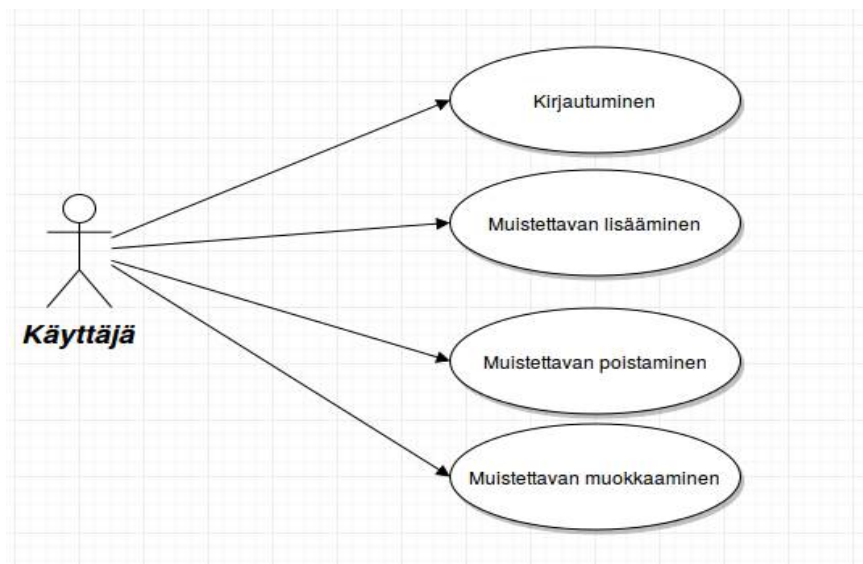
Valitsin aiheekseni Muistilistan. Järjestelmän tarkoitus on tehdä lista muistettavista asioista. Käyttäjä haluaa lisätä asioita listaan, priorisoida, ja luokitella niitä. Käyttäjä haluaa myös kavereidensa voivan käyttää järjestelmää.

Tavoitteena on saada toimiva muistilista, jossa voi lisätä, poistaa ja muokata tehtäviä. Muita toimintoja ovat kirjautuminen, luokkien lisäys ja poisto, sekä ne voivat olla sisäkkäisiä. Tavoitteena on saada mahdollisimman monta näistä toiminnoista toteutettua.

Toteutan järjestelmän php:llä. Työ toteutetaan laitoksen users-palvelimelle.

## 2. Yleiskuva järjestelmästä

### Käyttötapauskaavio



Käyttäjät ovat Ossi ja hänen kaverinsa, kaikilla on samat oikeudet.

### Käyttäjärühmät

Pääkäyttäjä

Ossi Opiskelija, kirjautunut

Muut käyttäjät, kirjautunut

Ossin kaverit

**Käyttötapauskuvaukset**

Kirjautuminen

Syötetään käyttäjätunnus ja salasana, ja painetaan kirjaudu-nappia.

Askareen lisäys

Lisätään askare, tallennetaan se. Luokitellaan haluttaessa askare ja priorisoidaan se.

Askareen poisto

Valitaan askare ja poistetaan se.

Askareen muokkaus

Valitaan askare, mennään muokkaus tilaan, muokataan ja tallennetaan.

Askareen tärkeysjärjestyksen lisäys

Lisätään luodessa

Askareen tärkeysjärjestyksen poisto

Valitaan askare, muokkaus tilaan ja poistetaan priorisointi.

Luokan vaihtaminen

Valitaan askare, muokkaus tilaan ja vaihdetaan luokka.

Askareiden luokittelu

Tehdään lisäyksessä.

### 3. Järjestelmän tietosisältö



#### Tietokohde: Henkilo

Attribuutti	Arvojoukko	Kuvaus
id	int	Käyttäjän numero-tunnus
nimi	String	Käyttäjän nimi
salasana	String	Käyttäjän salasana

Henkilolla voi olla monta tai ei yhtään Muistettavaa.

**Tietokohde: Muistettava**

Attribuutti	Arvojoukko	Kuvaus
id	int	Muistettavan asian numero-tunnus
henkilo_id	int	Viiteavain Henkilo-luokkaan
luokka_id	int	Viiteavain Luokka-luokkaan
nimi	String	Muistettava asia
prioriteetti	int	Muistettavan asian tärkeysjärjestys
kuvaus	String	Lisätietoa muistettavasta asiasta
pvm	DATE	Muistettavan asian päivämäärä, deadline

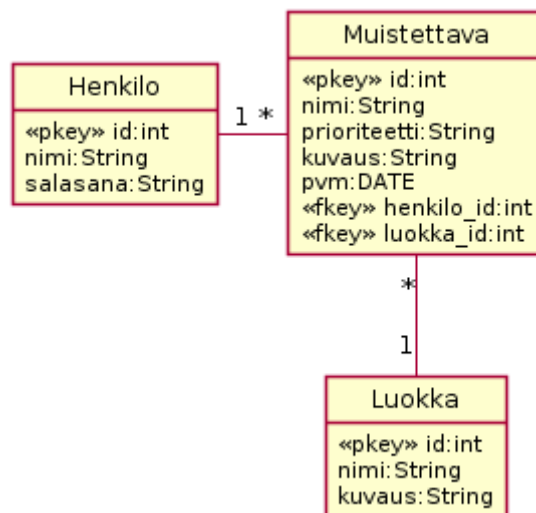
Muistettavalla voi olla vain yksi Henkilo ja Luokka.

### Tietokohde: Luokka

Attribuutti	Arvojoukko	Kuvaus
id	int	Luokan numero-tunnus
nimi	String	Luokan nimi
kuvaus	String	Luokan kuvaus

Luokalla voi olla monta Muistettavaa.

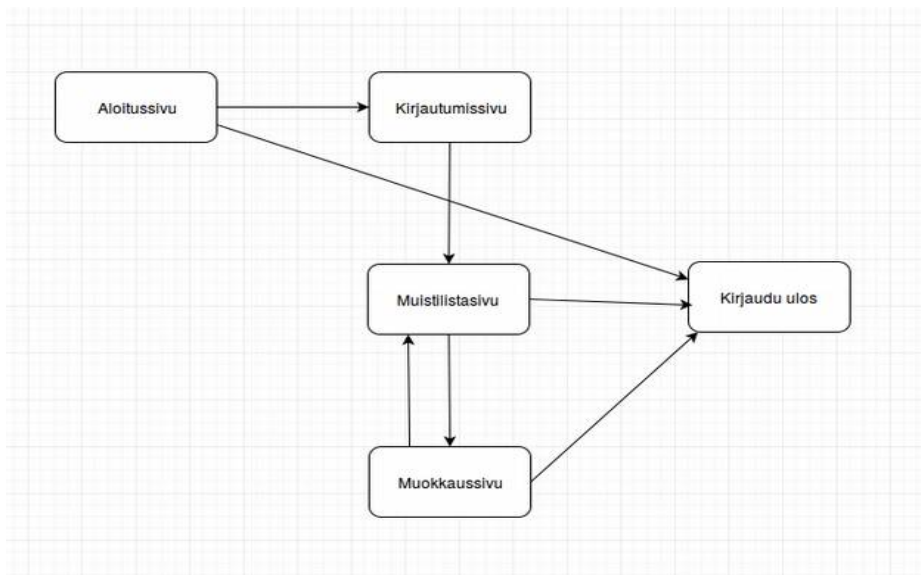
## 4. Relaatiotietokantakaavio



## 5. Järjestelmän yleisrakenne

Tietokantasovellusta tehdessä on noudatettu MVC-mallia. Kontrollerit, näkymät ja mallit sijaitsevat hakemistoissa controllers, views- ja models. Käytetyt apukirjastot on sijoitettu hakemistoon lib ja asetukset ovat tiedostossa settings.php.

## 6. Käyttöliittymä ja järjestelmän komponentit



Kirjautumista vaativat sivut Muistilistasivu, Muokkaussivu ja uloskirjautuminen.

## 7. Asennustiedot

Asenna sovellus kopioimalla sen tiedostot palvelimen nettiin näkyvään hakemistoon (esim. usersin htdocs-hakemisto).

## 8. Käyttöohje

Harjoitustyö löytyy osoitteesta: <http://heinihaa.users.cs.helsinki.fi/tsoha/todo>

Käyttäjätunnus: Ossi, salasana: Ossi.

Lisättävän muistettavan muoto tulee olla:

ID:int, nimi:String, Luokka:String, Prioriteetti:int, Kuvaus:String, Deadline:date. Nimi, Prioriteetti ja Deadline eivät saa olla tyhjiä. Muokkaus tapahtuu muokkaa sivulla ID:n perusteella.

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

Sovellusta voisi kehittää vielä lisäämällä muita käyttäjiä, eli lisäämällä ne Henkilo-tauluun. Jokaisella käyttäjällä olisi omat muistilistat, eli muistettavilla olisi määritelty henkilö\_id. Testausta on tehty kokeilemalla ohjelman toimintaa.

## 10. Omat kokemukset

Aluksi minulla oli vaikeuksia saada urers-palvelin toimimaan ja näkymien kanssa. Muutama pieni virhe löytyi ja pääsin hyvin jatkamaan ohjelman kehitystä. Tietokantojen ja yhteyden luominen oli yllättävän helppoa ja materiaaleissa oli siihen, kuten muuhunkin, hyvät ohjeet. Harjoitustyö palautti mieleeni webbipuolen ohjelmointia ja opin integroimaan tietokantoja siihen.

## 11. Muu dokumentaatio

Create table-lauseet:

```
CREATE TABLE Henkilo(
id SERIAL PRIMARY KEY, -- SERIAL tyyppinen pääavain pitää huolen, että tauluun lisätyllä
rivillä on aina uniikki pääavain. Kätevää!
nimi varchar(50) NOT NULL, -- Muista erottaa sarakkeiden määrittelyt pilkulla!
salasana varchar(50) NOT NULL
);
CREATE TABLE Luokka(
id SERIAL PRIMARY KEY,
nimi varchar(15) NOT NULL,
kuvaus varchar(50)
);
CREATE TABLE Muistettava(
id SERIAL PRIMARY KEY,
henkilo_id INTEGER REFERENCES Henkilo(id), -- Viiteavain Henkilo-tauluun
luokka_id INTEGER REFERENCES Luokka(id),
nimi varchar(50) NOT NULL,
prioriteetti varchar(2),
kuvaus varchar(50),
pvm DATE
);
Kaikki sql-koodi löytyy githupista.
```