

**Muistilista**

# 1. Johdanto

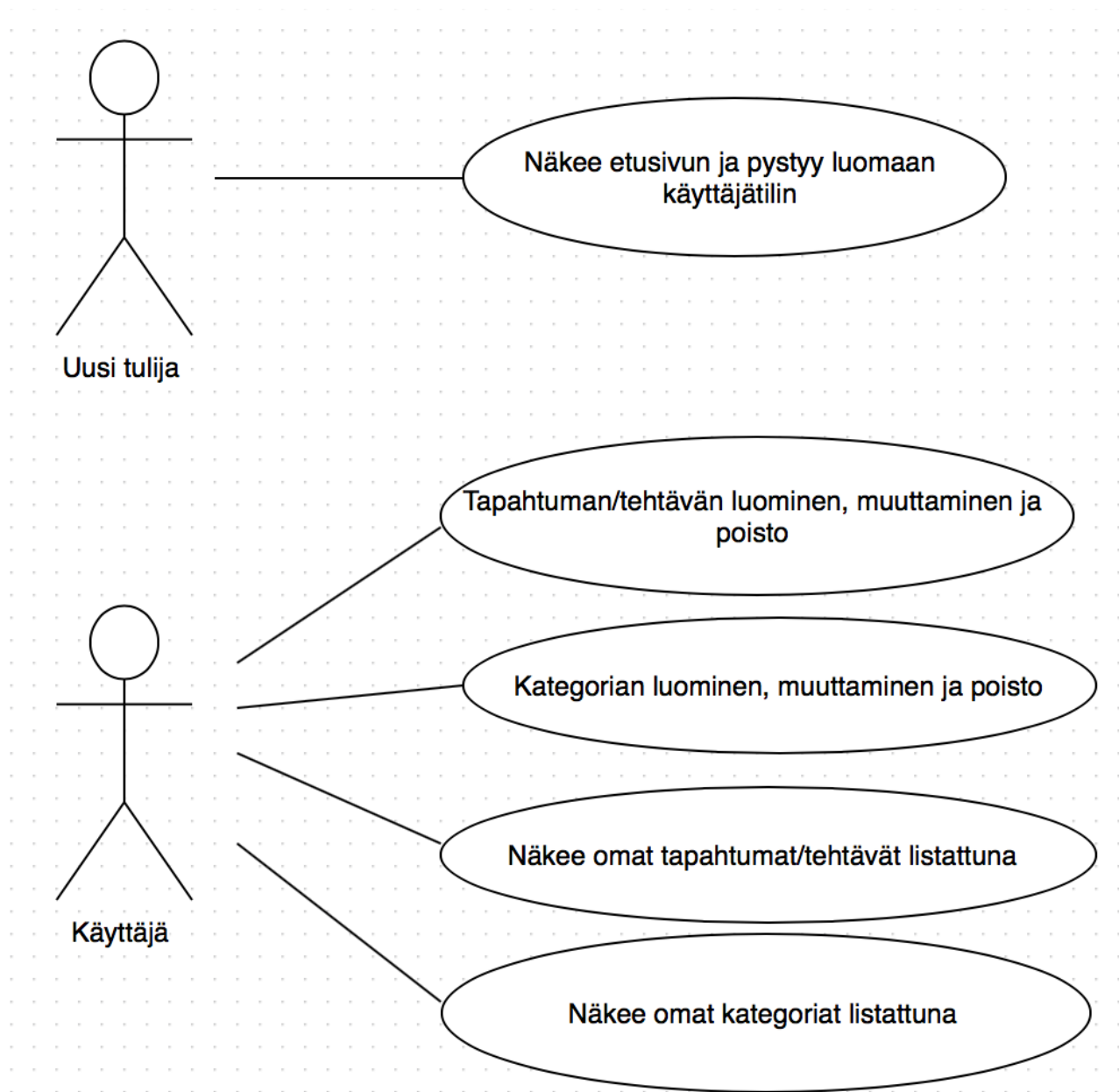
Projektin tarkoituksena on tuottaa muistilista-tietokantasovellus, joka auttaa omien tehtävien ja tapahtumien muistamisessa, priorisoinnissa ja aikatauluttamisessa.

Sovellus ohjelmoidaan user-palvelimelle Apache web-palvelimen taakse PHP-kieltä käyttäen. Tietokantana käytetään PostgreSQL-tietokantaa laitoksen tietokantapalvelimella.

Sovellusta käyttääkseen käyttäjän täytyy luoda itselleen henkilökohtainen käyttäjätili, jonka avulla käyttäjä tunnistetaan. Tällöin kaikki sovellukseen lisättävät tiedot näkyvät vain sille käyttäjälle, joka on sen sinne lisännyt.

Sovelluksen tarkoitus on käyttäjän elämän helpottaminen toimimalla muistilistana, johon omat tulevat tapahtumat ja tehtävät voi kirjata ylös helposti. Näitä tehtäviä ja tapahtumia pystyy myös priorisoimaan ja luokittelemaan kategorioihin, joita käyttäjä voi itse luoda. Tehtävillä ja tapahtumilla on myös deadline ja ne voidaan merkata suoritetuksi.

## 2. Yleiskuva järjestelmästä



### Käyttäjärühmät

#### Uusi tulija:

Uusi tulija on sivulla käyvä henkilö, joka ei ole rekisteröitynyt. Voidakseen käyttää palvelua uuden tulijan on rekisteröidyttävä.

#### Käyttäjä:

Käyttäjä on sivulle rekisteröitynyt ja kirjautunut sovelluksen käyttäjä

## **Käyttötapauskuvaukset**

### Uuden tulijan käyttötapaukset

Näkee etusivun ja pystyy luomaan käyttäjätilin:

Etusivulla voi kirjautua sisään ja siellä on myös linkki uuden käyttäjätilin luontiin.

### Käyttäjän käyttötapaukset

Tapahtuman/tehtävän luominen, muuttaminen ja poisto:

Kirjautunut käyttäjä pystyy luomaan uusia tehtäviä ja tapahtumia, sekä muokkaamaan ja poistamaan niitä. Tapahtumaa tai tehtävää luodessa sille annetaan ainakin nimi, deadline ja prioriteetti. Niille voi myös antaa yhden tai useamman kategorian.

Kategorian luominen, muuttaminen ja poisto:

Kirjautunut käyttäjä pystyy luomaan, muuttamaan ja poistamaan kategorioita.

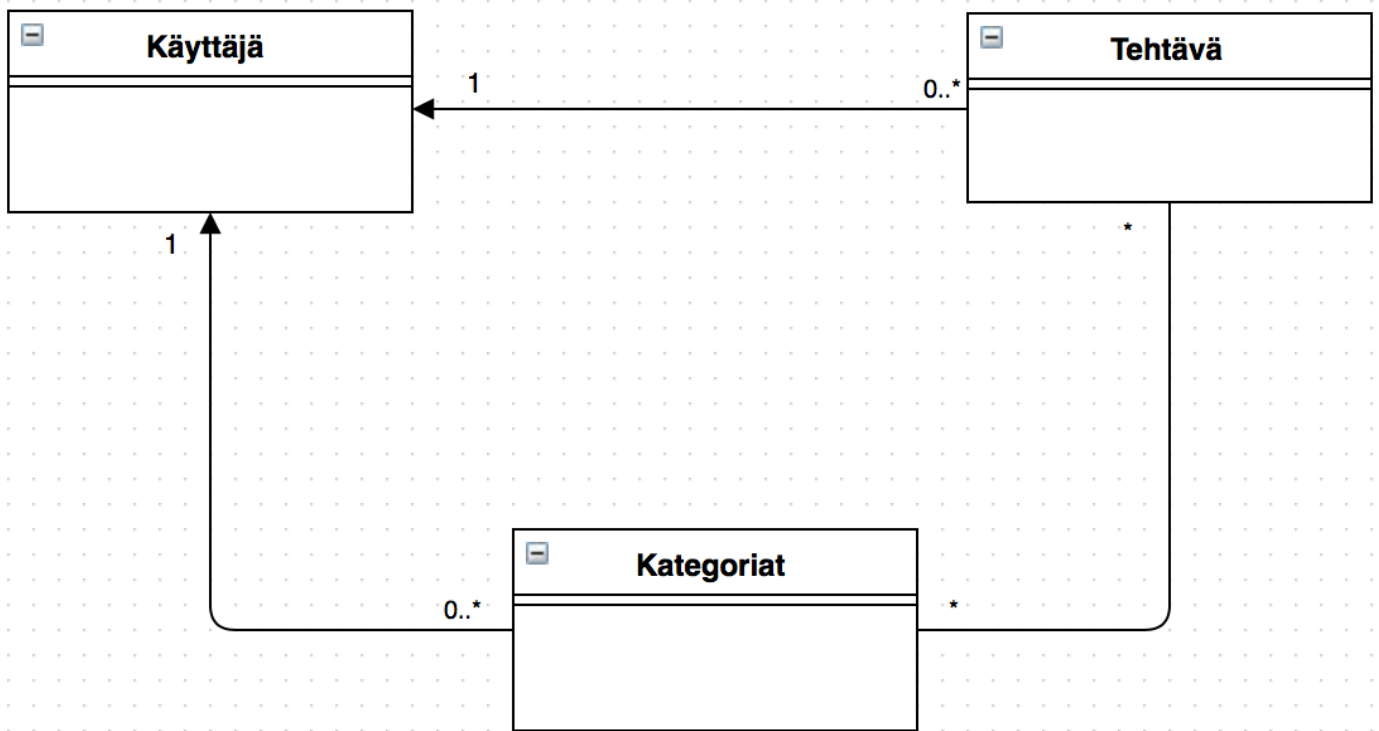
Näkee omat tapahtumat/tehtävät listattuna:

Kirjautunut käyttäjä näkee tapahtumat ja tehtävät listattuna, muistilista näkymässä.

Näkee omat kategoriat listattuna:

Kirjautunut käyttäjä näkee kategoriat listattuna, kategoria näkymässä.

### 3. Järjestelmän tietosisältö



Käyttäjällä voi olla useita tehtäviä ja kategorioita, kuitenkin tehtävät ja kategoriat kuuluvat vain yhdelle käyttäjälle. Tehtävällä voi olla useita kategorioita ja kullakin kategorialla voi olla useita tehtäviä.

#### Tietokohde: Käyttäjä

Attribuutti	Arvojoukko	Kuvailu
ID	kokonaisluku NOT NULL	pääavain
käyttäjätunnus	merkkijono max 30 NOT NULL	käyttäjän tunnus
salasana	merkkijono max 30 NOT NULL	käyttäjä salasana

### Tietokohde: Tehtävä

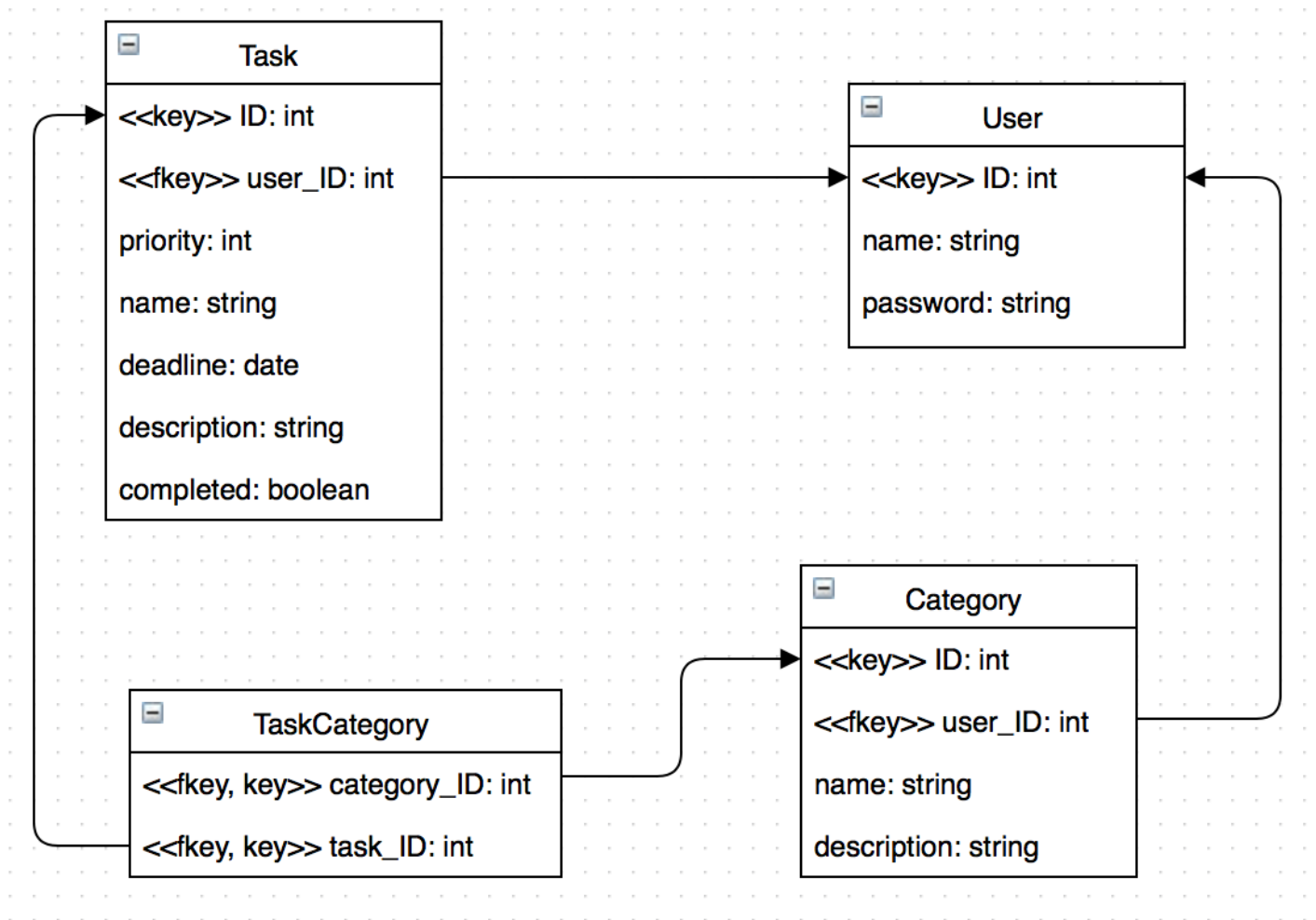
Attribuutti	Arvojoukko	Kuvailu
ID	kokonaisluku NOT NULL	pääavain
käyttäjä	kokonaisluku NOT NULL	vierasavain
nimi	merkkijono max 255 NOT NULL	tehtävän nimi
prioriteetti	kokonaisluku NOT NULL	kuinka tärkeä tehtävä on
deadline	päivämäärä NOT NULL	mihin asti on aikaa suorittaa tehtävä
suoritettu	tosi/epätosi	onko tehtävä suoritettu
kuvaus	merkkijono max 2000	tehtävän kuvaus

### Tietokohde: Kattegoria

Attribuutti	Arvojoukko	Kuvailu
ID	kokonaisluku NOT NULL	pääavain
käyttäjä	kokonaisluku NOT NULL	vierasavain
nimi	merkkijono max 255 NOT NULL	kattegorian nimi
kuvaus	merkkijono max 2000	kattegorian kuvaus

Kattegorian ja tehtävän välillä on monesta moneen -suhde, joka toteutetaan liittotaululla, jossa on vierasavaimet Tehtävään ja Kattegoriaan. Yhdessä avaimet muodostavat taulun pääavaimen.

## 4. Relaatietietokantakaavio



## 5. Käynnistys- /käyttöohje

Harjoitustyö on osoitteessa <http://kauvo.users.cs.helsinki.fi/muistilista/>  
Sisään voi kirjautua tunnuksilla:

Käyttäjätunnus: Anna  
Salasana: sohvi

Käyttäjätunnus: Mikael  
Salasana: miksu

## 6. Järjestelmän yleisrakenne

Tietokanta sovellusta tehdessä on noudatettu MCV-mallia. Kontrollerit, näkymät ja mallit sijaitsevat hakemistoissa controllers, views ja models. Hakemistossa view olevat näkymät on jaettu mallien mukaisiin alihakemistoihin. Malleja vastaavat kontrollerit on nimetty tyyliin mallinnimi\_controller. Kaikki tiedostonimet on kirjoitettu pienellä. Kontrollien ja mallien yleiset apuluokat löytyvät projektin hakemistosta lib.

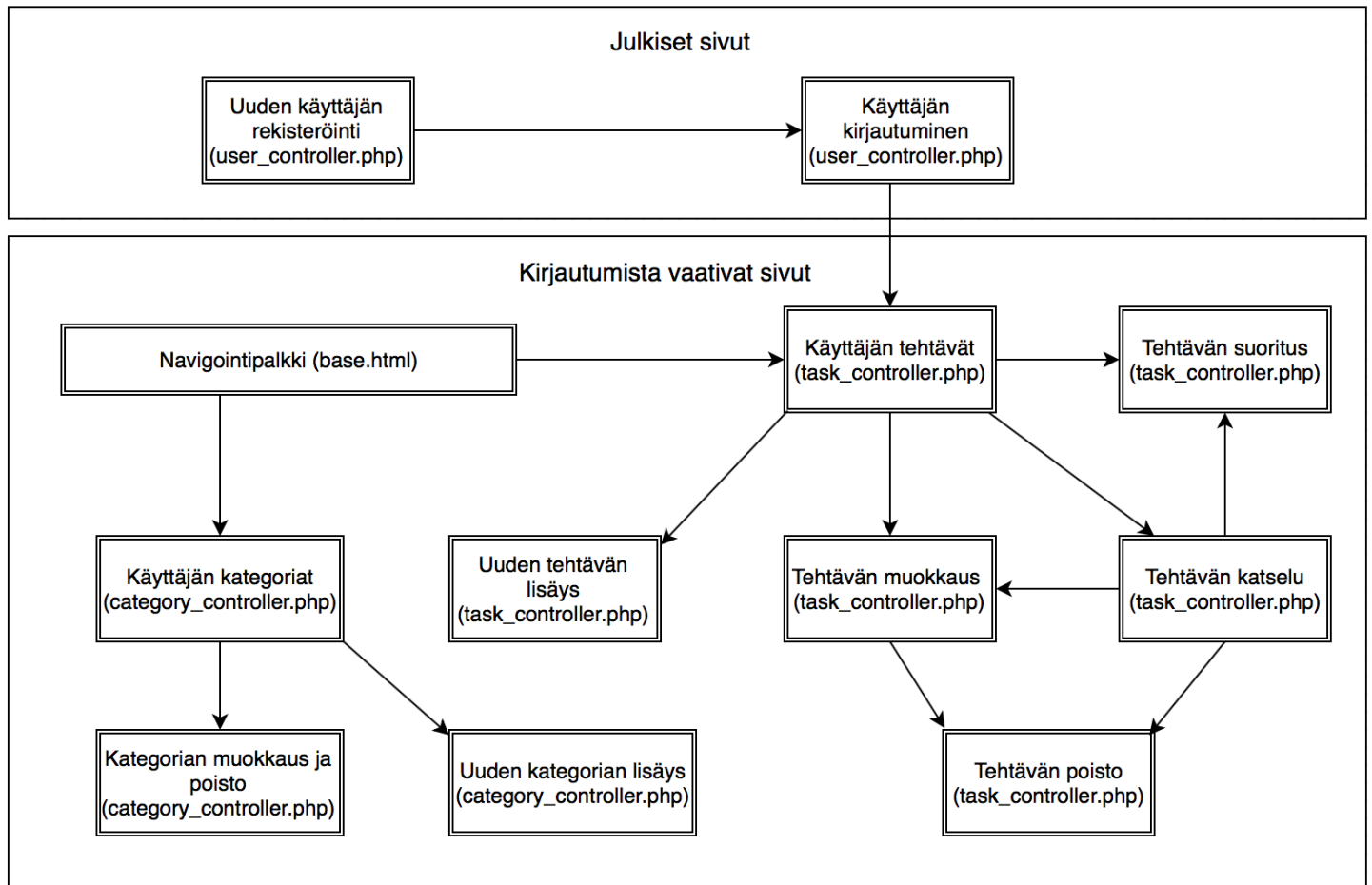
Tietokantayhteys, sovelluksen reitit ja asennusympäristö määritellään hakemiston config sisältäissä tiedostoissa.

Dokumentti löytyy hakemistosta doc

Tietokantataulujen luontiin, poistamiseen ja testidatan lisäämiseen tarvittavat SQL-kyselyt ovat sql-hakemistossa.



## 7. Käyttöliittymä ja järjestelmän komponentit



## 8. Asennustiedot

Sovellus on suunniteltu asennettavaksi tietojenkäsittelytieteen laitoksen user-palvelimelle. Asenna sovellus kopioimalla sen tiedostot palvelimen nettiin näkyvään hakemistoon, jonka jälkeen asetat tietokannan yhteystiedot oikeiksi tiedostoon `libs/config.php`.