

# Tietokantasovelluksen dokumentaatio

## Aihe: Muistilista

### Johdanto

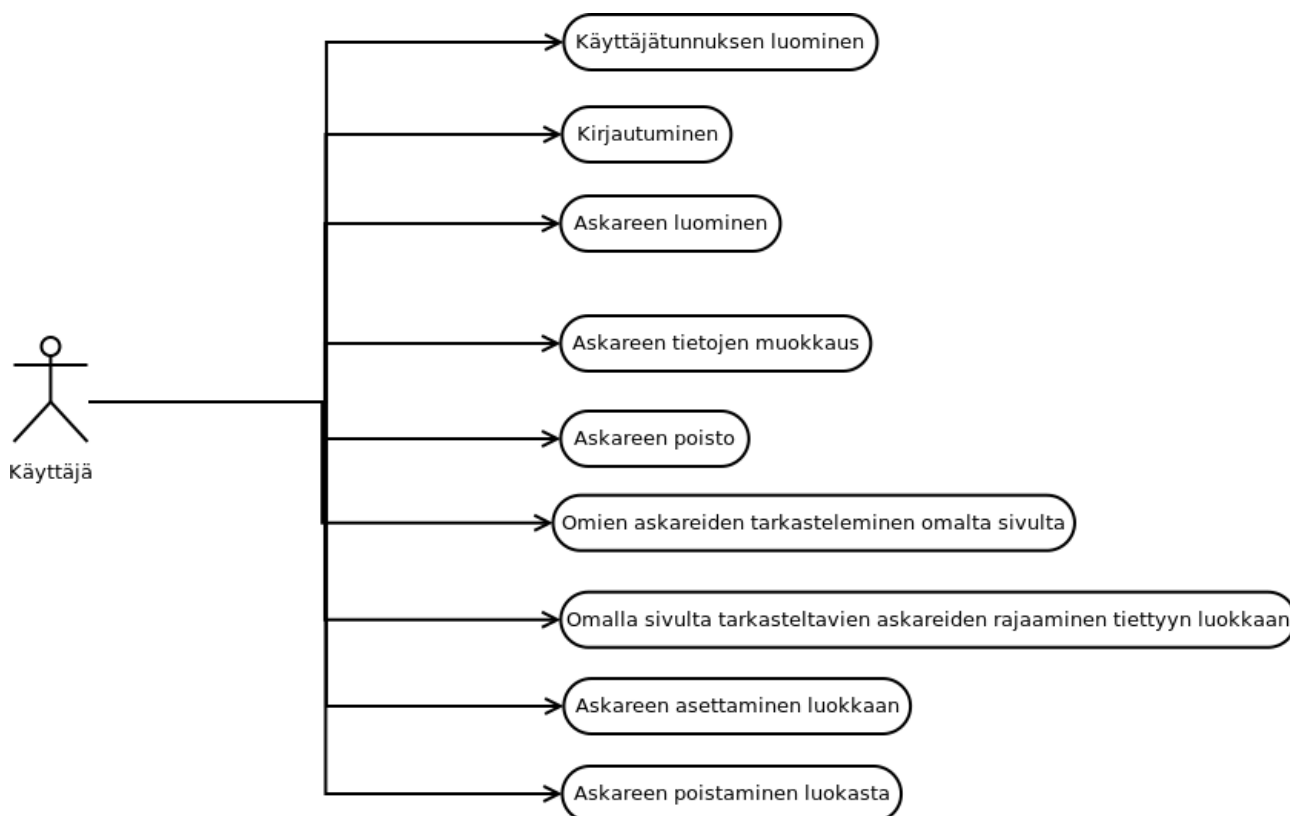
Tarkoituksena on luoda tietokantasovellus, johon eri käyttäjät voivat kirjautua ja lisätä askareitaan muistiin. Askareille asetetaan omat nimet, askareisiin liittyvät tiedot, deadline ja tärkeysasteet.

Askareet voidaan asettaa yhteen tai useampaan luokkaan(opiskelu, kotiaskareet, yms...) tai poistaa luokasta. Askareiden tietoja(nimi, tiedot, deadline, tärkeysaste) voidaan muuttaa tai askare voidaan poistaa.

Työ toteutetaan osoitteeseen [hanranti.users.cs.helsinki.fi/tsoha](http://hanranti.users.cs.helsinki.fi/tsoha). Järjestelmä toteutetaan käyttäen PostgreSQL -tietokantapalvelinta ja PHP:tä.

### Yleiskuva järjestelmästä

Käyttötapauskaavio:



Käyttäjäryhmät:

Käyttäjä

Käyttäjä on kuka tahansa henkilö, joka voi rekisteröityä sivun käyttäjäksi ja tämän jälkeen kirjautua sisään käyttämään muita sivun toimintoja.

## Käyttäjän käyttötapauskuvaukset:

### Askareen luominen

Kirjautunut käyttäjä voi luoda askareen, joka näkyy käyttäjän omalla sivulla. Askareelle annetaan nimi, askareeseen liittyvät tiedot, deadline eli viimeinen päivämäärä, jolloin askare on voimassa, ja tärkeysaste, joka on askareen tärkeyden verrattuna muihin askareisiin määrittävä luku. Käyttäjän tulee olla kirjautunut sisään luodakseen askareen.

### Askareen tietojen muokkaus

Askareen nimi, tiedot, deadline ja tärkeysaste voidaan muuttaa askareen muokkaussivulla olevalla lomakkeella, kun käyttäjä on kirjautunut sisään.

### Askareen poisto

Askare voidaan poistaa omalta sivulta omien askareiden listasta käyttäjän ensin kirjaututtua sisään.

### Omien askareiden tarkasteleminen omalta sivulta

Käyttäjä voi kirjaututtuaan sisään tarkastella omia askareitaan listasta omalta sivulta. Tällöin käyttäjälle näkyvät askareiden nimet, tiedot, deadlinet ja tärkeysasteet.

### Omalla sivulla tarkasteltavien askareiden rajaaminen tiettyyn luokkaan

Kirjautunut käyttäjä voi rajata omalla sivulla näkyvät askareet johonkin luokkaan, jota käyttäjä on käyttänyt jossakin askareessa. Tällöin oma sivu listaa vain tähän luokkaan kuuluvat askareet.

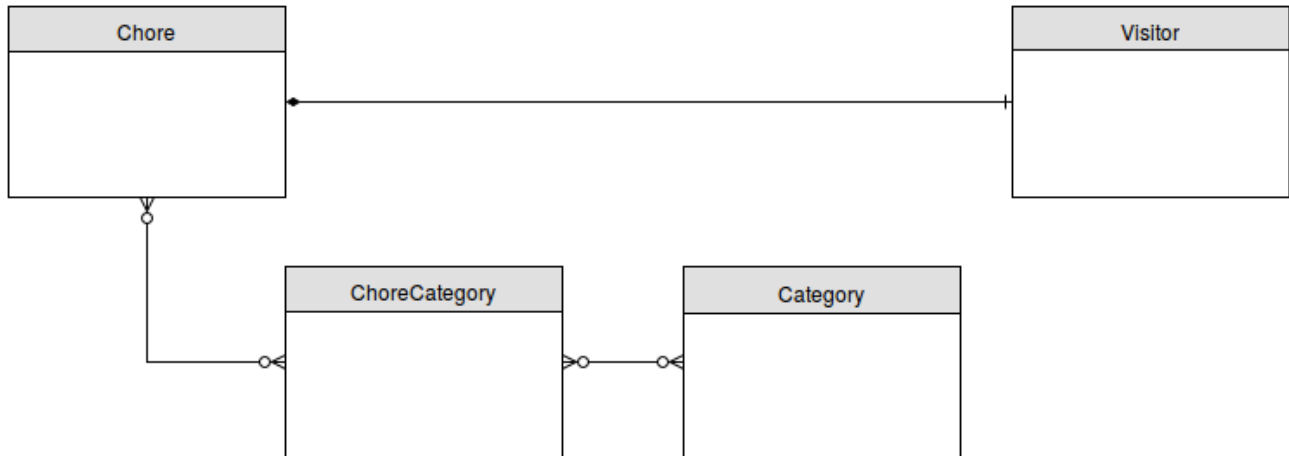
### Askareen asettaminen luokkaan

Askare voidaan asettaa kuulumaan johonkin luokkaan askareen tietojen muokkaussivun lomakkeella.

### Askareen poistaminen luokasta

Askare voidaan poistaa mistä tahansa luokasta, johon se on lisätty, askareen muokkaussivulla olevasta listasta.

## Järjestelmän tietosisältö



Tietokohde: Visitor

Atribuutti	Arvojoukko	Kuvailu
id	Kokonaisluku	Yksilöllinen luku, joka toimii tämän visitorin tunnisteena
name	Merkkijono, max. 50 merkkiä	Kayttajatunnus palveluun
password	Merkkijono, max. 50 merkkiä	Salasana palveluun

Jokainen chore tuntee tasan yhden visitorin. Tällöin chore kuuluu tälle visitorille.

Tietokohde: Chore

Atribuutti	Arvojoukko	Kuvailu
id	Kokonaisluku	Yksilöllinen luku, joka toimii tämän choren tunnisteena
name	Merkkijono, max. 50 merkkiä	Choren nimi
info	Merkkijono, max. 350 merkkiä	Tietoa choresta
deadline	Päiväys yyyy-mm-dd	Viimeinen päivä, jolloin chore on voimassa
importancedegree	Kokonaisluku	Luku, joka määrittää, kuinka tärkeä chore on
visitorid	Merkkijono, max. 50 merkkiä	Viittaus visitoriin, joka omistaa tämän choren

Jokainen chore tuntee tasan yhden visitorin. Tällöin visitor omistaa tämän choren. ChoreCategory voi ydistää monta tai ei yhtään samaa tai eri chorea moneen tai ei yhteenkään samaan tai eri categoryyn. Chorella on siis monen suhde moneen suhde Categoryyn.

## Tietokohde: Category

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max 50. merkkiä	Categoryn nimi

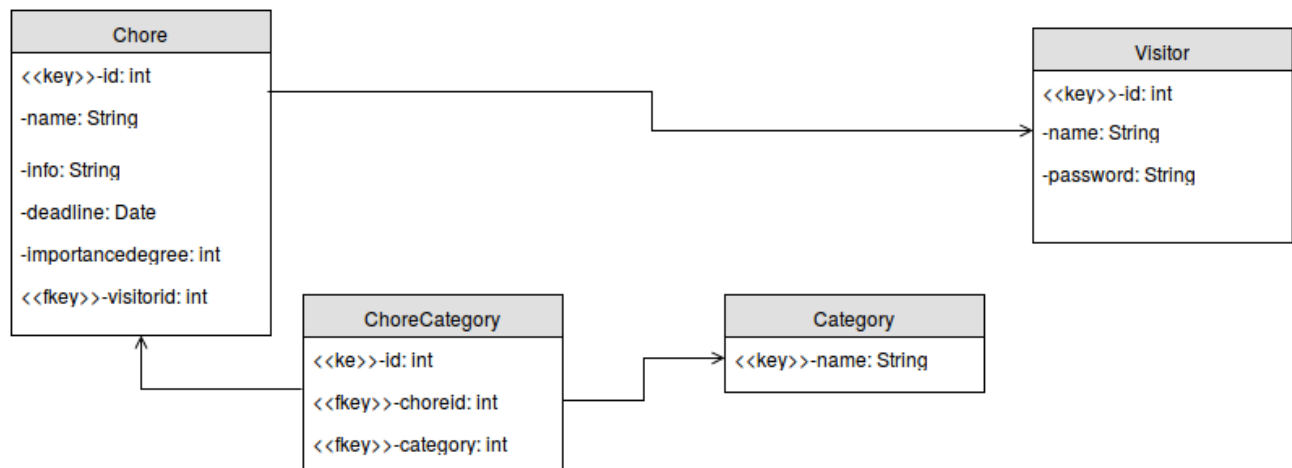
Categoryllä on monen suhde moneen suhde Choreen.

## Tietokohde: ChoreCategory

Attribuutti	Arvojoukko	Kuvailu
id	Kokonaisluku	Yksilöllinen luku, joka toimii tämän choreCategoryn tunnisteena
choreid	Kokonaisluku	Viittaus choreen, joka kuuluu toisena attribuuttina olevaan categoryyn
category	Kokonaisluku	Viittaus categoryyn, johon toisena attribuuttina oleva chore kuuluu.

ChoreCategory liittää choreja ja categoryja toisiinsa. Sama tai eri chore voidaan liittää useaan eri categoryyn ja sama tai eri category voidaan liittää useaan eri choreen.

## Relaatiotietokantaavaio



## Käynnistys- / käyttöohje

Työ löytyy osoitteesta <http://hanranti.users.cs.helsinki.fi/tsoha/>

Kirjautuminen sivulle onnistuu oikeasta yläkulmasta käyttäjätunnuksella ”Kalle” ja salasanalla ”Kalle123”.

Kirjautuneena sisään voi oikeasta yläkulmasta siirtyä omalle sivulle, joka listaa omat askareet.

Omalla sivulla olevat askareet voi rajata johonkin käyttäjän käyttämään luokkaan tai näyttää kaikki askareet oikeassa alakulmassa olevasta valikosta.

Omia askareita voi poistaa, askareen tietoja voi tarkastella sen nimessä olevan linkin kautta ja käyttäjälle voi lisätä uusia askareita muistilistan alla olevasta linkistä.

Askareen omalta sivulta voi siirtyä askareen tietojenmuokkaussivulle, jossa voi muuttaa askareen tietoja, lisätä askareen uuteen luokkaan tai poistaa askareen luokasta.

## Järjestelmän yleisrakenne

Tietokantasovellusta tehdessä on noudatettu MVC-mallia. Kontrollerit, näkymät ja mallit sijaitsevat hakemistoissa controllers, views- ja models. UserController käsittelee rekisteröitymiseen ja kirjautumiseen liittyviä sivuja. Chore controller käsittelee askareiden ja näihin liittyvien luokkien listausta, lisäystä, muokkaamista ja poistoa. FrontController käsittelee etusivua.

views -hakemiston juuressa on kaikkien html -sivujen perusta default.html ja etusivu front.html. Hakemistosta chores löytyy askareiden ja luokkien käsittelyyn liittyvät html -sivut ja hakemistosta user löytyy kirjautumista käsittelevät sivut. macros -hakemistosta löytyy uudelleen käytettävät html -formit.

Kansio assets sisältää käytetyt css- ja javascript-tiedostot.

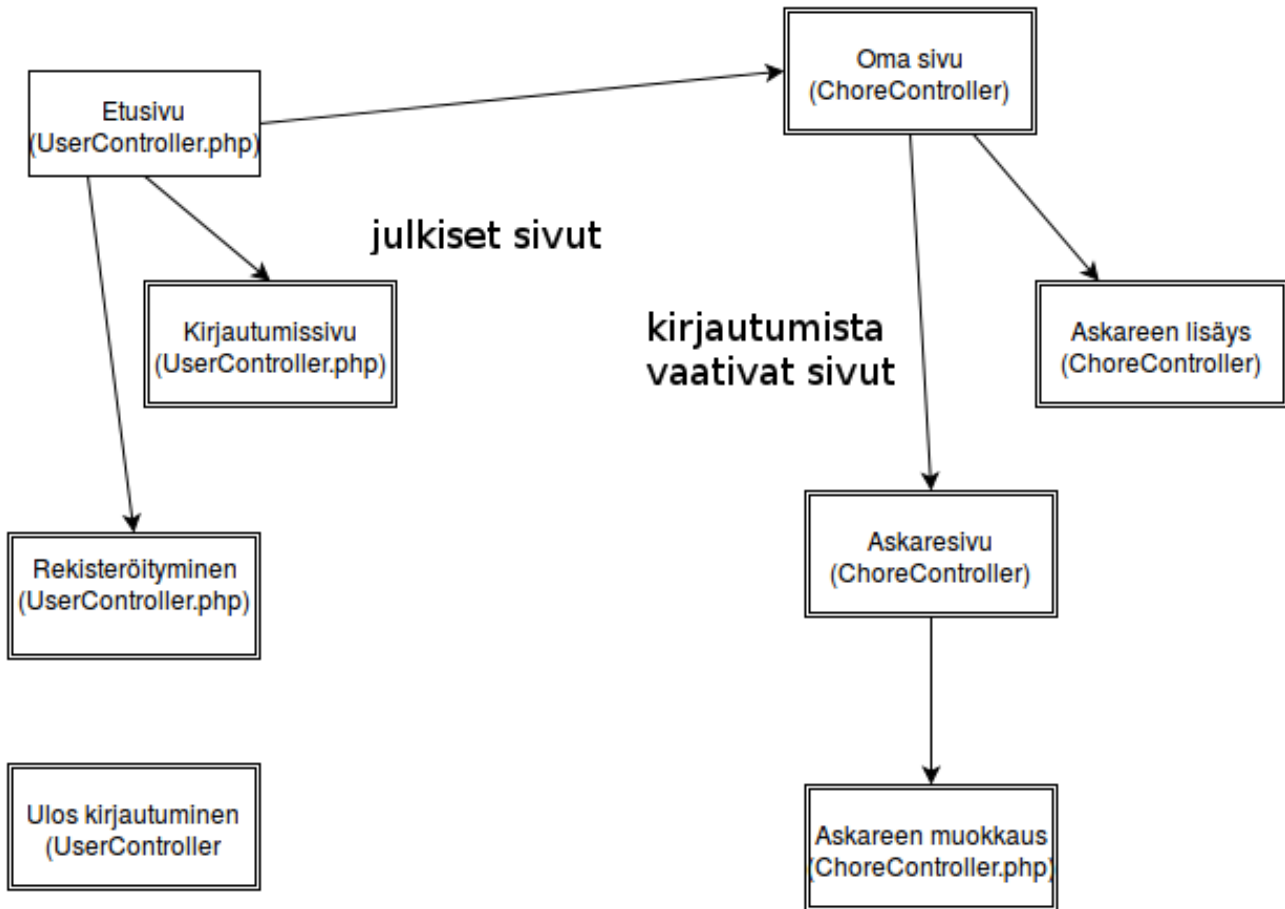
config sisältää muun muassa routes.php:n, joka käsittelee sovelluksen reitit.

lib sisältää luokat BaseController, jonka kaikki kontrollerit perivät ja BaseModel, jonka kaikki mallit perivät. BaseController sisältää sessioihin liittyviä funktioita, ja BaseModel sisältää mallien konstruktorin ja virheiden käsittelyn. lib sisältää myös uudelleenohjaukseen liittyvän redirect.php:n ja näkymän näyttämiseen liittyvän view.php:n.

sql sisältää tietokantataulujen luontiin poistoon ja testidatan lisäämiseen liittyviä tiedostoja.

vendor sisältää muun muassa bootstrapin css- ja javascript-tiedostot ja jqueryn javascript-tiedoston.

## Käyttöliittymä ja järjestelmän komponentit



Vasemmalla puolella ovat julkiset sivut ja oikealla kirjautumista vaativat sivut.

Sivulla on navigaatio palkki, joka mahdollistaa kirjautumisen tai rekisteröitymisen aina kun käyttäjä ei ole kirjautunut sisään. Kun käyttäjä on kirjautunut sisään, navigaatiopalkista voi kirjautua ulos tai siirtyä omalle sivulle.

## Asennustiedot

Asennukseen vaaditaan cs-tunnus palvelimelle users.cs.helsinki.fi ja, että PHP-tuki ja PostgreSQL on otettu käyttöön palvelimella.

Kloonaa sovelluksen github-repositorio(<https://github.com/hanranti/Tsoha-Bootstrap>) omalle koneellesi komennolla: `git clone git@github.com:hanranti/Tsoha-Bootstrap.git`

Korvaa repositorion config-kansiossa sijaitsevassa environment.sh tiedostossa kohtaan `USERNAME="hanranti"` oma cs-tunnuksesi hanrantin tilalle ja kohtaan `PROJECT_FOLDER="tsoha"` haluamasi polku sovellukseen osoitteessa `USERNAME.users.cs.helsinki.fi/PROJECT_FOLDER` tsohan tilalle.

Suorita projektin juuressa komento `bash bootstrap.sh`.

Suorita projektin juuressa komento `bash deploy.sh`.

Suorita projektin juuressa komento `bash create_tables.sh`.

Sovelluksesi löytyy nyt osoitteesta `USERNAME.users.cs.helsinki.fi/PROJECT_FOLDER`.