

# Tietokantasovellus Muistilista

*Noora Imberg*

*kevät 2014*

# Sisällys

## Sisällys

1. Johdanto

2. Yleiskuva järjestelmästä

Käyttötapauskaavio

Käyttäjäryhmät

3. Käyttötapaukset

Käyttäjän käyttötapaukset

Ylläpitäjän käyttötapaukset

4. Järjestelmän tietosisältö

Tietokohdekuvaukset

5. Relaatiotietokantakaavio

6. Järjestelmän yleisrakenne

7. Järjestelmän komponentit

8. Käytöliittymä

9. Asennustiedot

10. Käyttöohje

11. Testaus, puutteet ja jatkokehitysideoat

12. Omat kokemukset

13. Liitteet

## 1. Johdanto

Muistilista on tietokantasovellus-harjoitustyökurssin projektti, jonka tehtävänanto löytyy täältä: <http://advancedkittenry.github.io/suunnittelua/aiheet/Muistilista.html> .

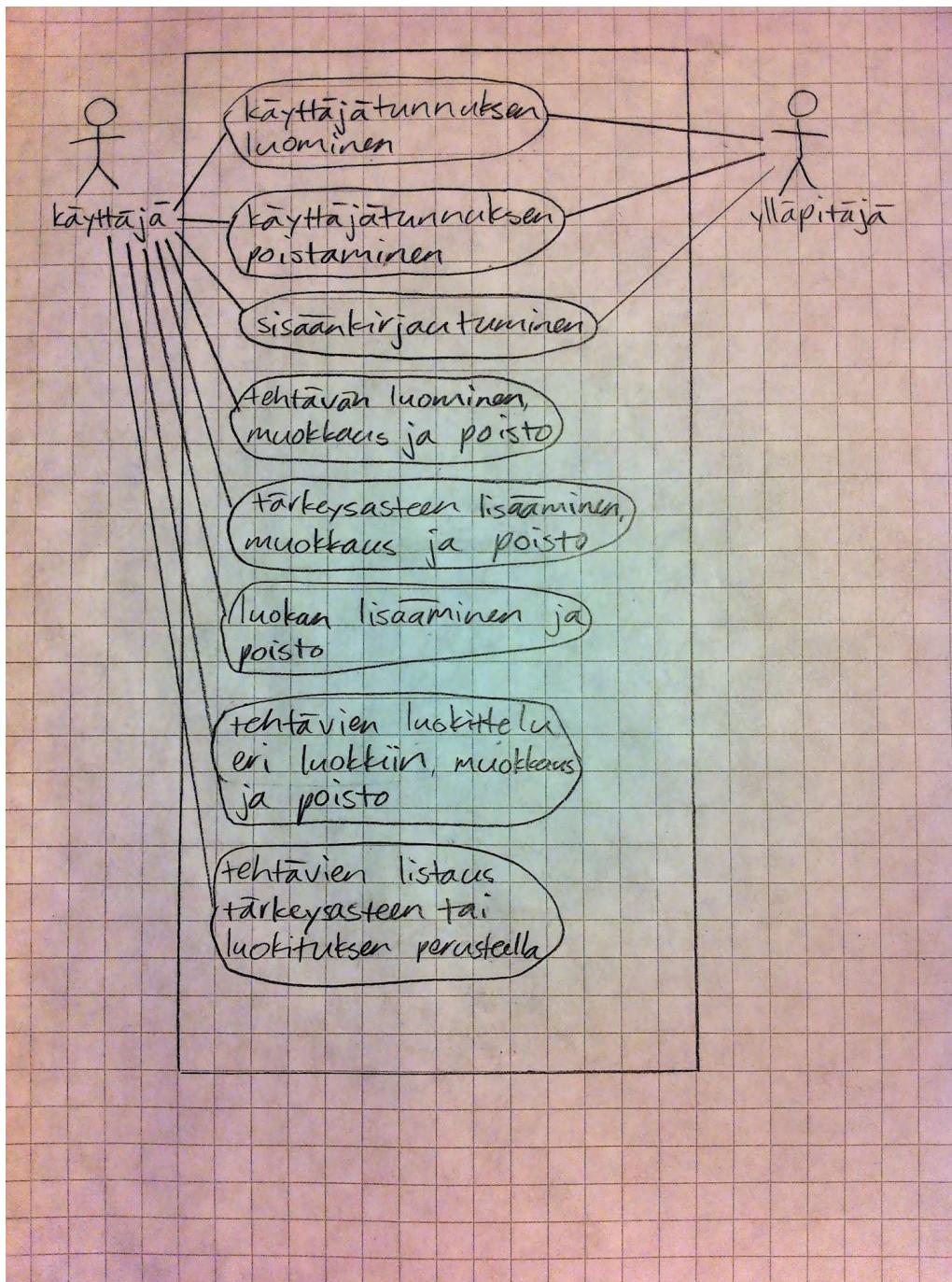
Muistilista on palvelu, johon pystyy sisäänkirjautumisen jälkeen lisäämään itselleen tehtäviä. Tehtävät voivat olla esimerkiksi asioita, jotka käyttäjän pitäisi muistaa tehdä. Muistilistaan lisättyjä tehtäviä on mahdollista muokata ja poistaa sekä antaa niille tärkeysasteita. Tehtävät voi lajitella tärkeysasteen perusteella tai niitä voi ryhmitellä erilaisiin luokkiin. Yksi tehtävä voi kuulua useampaan eri luokkaan.

Muistilistan avulla käyttäjä voi helposti ylläpitää listaa vielä tekemättömistä tehtävistä ja etsiä sieltä erilaisten kriteerien perusteella tehtäviä, jotka hänen pitäisi seuraavaksi tehdä. Kriteerinä voi olla tehtäville annettu tärkeysaste tai luokat, joihin ne kuuluvat.

Muistilista on toteutettu käytäen php:tä ja PostgreSQL-tietokantaa. Sovellus on laitoksen users-palvelimella.

## 2. Yleiskuva järjestelmästä

### Käyttötapauskaavio



## Käyttäjäryhmät

Käyttäjä on kuka tahansa, joka haluaa käyttää muistilistapalvelua omien tehtäviensä tallentamiseen.

Ylläpitäjä on palvelun ylläpitäjä, joka vastaa palvelun toiminnasta.

## 3. Käyttötapaukset

### Käyttäjän käyttötapaukset

Kun uusi käyttäjä haluaa aloittaa palvelun käyttämisen, hänen on luotava itselleen käyttäjätunnus. Myöhemmin palvelua käyttäessään käyttäjän on ensin kirjauduttava sisään käyttäen omaa käyttäjätunnustaan ja salasanaansa. Sisäänkirjautumisen jälkeen käyttäjä voi tehdä seuraavia asioita:

**Uuden tehtävän lisäys:** Käyttäjä voi lisätä itselleen uuden tehtävän ja halutessaan antaa sille tärkeysasteen. Jos käyttäjä on aiemmin luonut luokkia, joihin tehtäviä voi luokitella, hän voi myös lisätä uuden tehtävän näihin luokkiin.

**Tehtävän muokkaus ja poisto:** Käyttäjälle näytetään lista tehtävistä. Hänen voi valita jonkin näistä ja muokata sen kuvausta, tärkeysastetta tai luokkia. Hänen voi myös poistaa koko tehtävän.

**Luokkien lisäys ja muokkaus:** Käyttäjälle näytetään lista olemassaolevista tehtäväluokista. Käyttäjä voi lisätä itselleen uuden tehtäväluokan, johon voi myöhemmin lisätä omia tehtäviään. Luokka voi myös muokata ja poistaa.

**Tehtävien luokittelu:** Käyttäjä voi lisätä tehtäväluokkaan haluamansa tehtävät tai poistaa tehtävät tietystä luokasta.

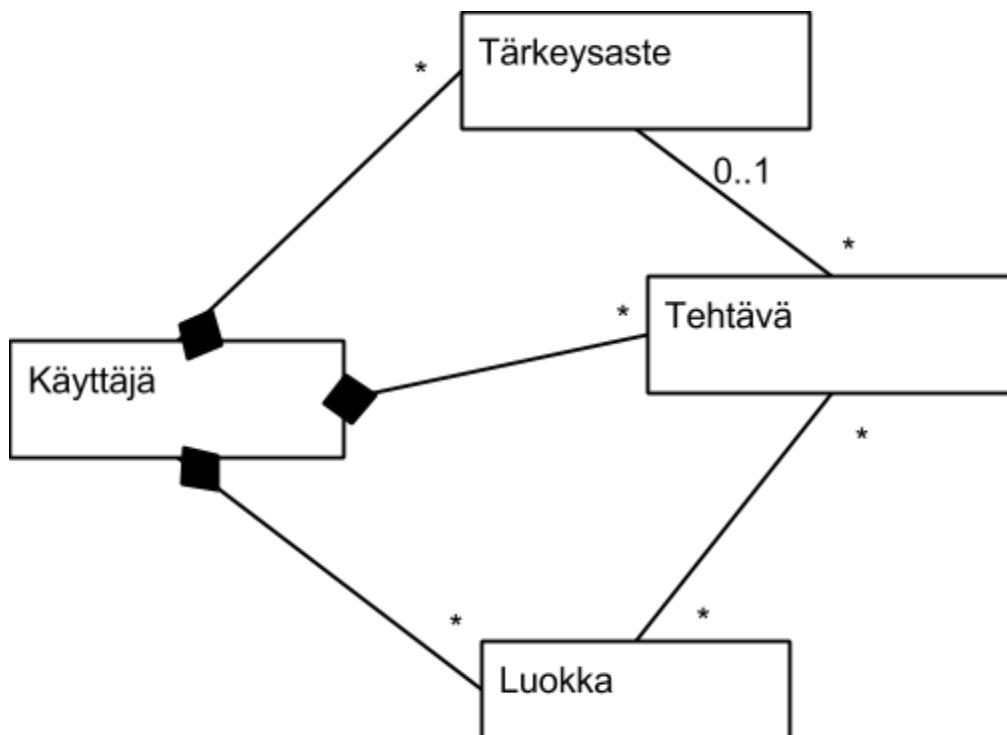
**Tehtävien listaus:** Käyttäjä voi listata tehtävänsä tiettyjen kriteerien perusteella. Hänen voi tarkastella esimerkiksi kaikkia tehtäviään, tietyn tärkeysasteen tehtäviä tai tiettyyn luokkaan kuuluvia tehtäviä. Käyttäjälle näytetään lista tehtävistä, jotka vastaavat annettuja kriteereitä.

### Ylläpitäjän käyttötapaukset

Ylläpitäjän on kirjauduttava sisään omalla käyttäjätunnukseellan ja salasanallaan. Tämän jälkeen hän voi tehdä seuraavia asioita:

**Käyttäjien listaus ja muokkaus:** Ylläpitäjä voi tarkastella listaan kaikista käyttäjistä ja muokata sitä esimerkiksi poistamalla jonkin käyttäjän.

#### 4. Järjestelmän tietosisältö



Käyttäjällä voi olla useita tehtäviä, luokkia ja tärkeysasteita. Kaikki tehtävät, luokat ja tärkeysasteet kuuluvat yhdelle käyttäjälle. Tehtävällä voi olla yksi tärkeysaste ja se voi kuulua yhteen tai useampaan luokkaan. Tärkeysasteeeseen ja luokkaan voi kuulua useampia tehtäviä.

#### Tietokohdekuvaukset

**Tietokohde: Käyttäjä:**

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku (32-bit)	käyttäjän tunnistenumero

nimi	merkkijono, max 20 merkkiä	käyttäjän tervehtimiseen käytetty nimi
käyttäjätunnus	merkkijono, max 20 merkkiä	käyttäjän käyttäjätunnus, jota käytetään sisäänsijautumiseen
salasana	merkkijono, max 20 merkkiä	käyttäjän salasana, jota käytetään sisäänsijautumiseen

Käyttäjä on palvelun käyttäjä, jolla on nimi, käyttäjätunnus ja salasana. Nimeä käytetään käyttäjän tervehtimiseen. Tunnusta ja salasanaa käytetään sisäänsijautumiseen.

#### Tietokohde: Tehtävä:

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku (32-bit)	tehtävän tunnistenumero
kuvaus	merkkijono, max 1000 merkkiä	tehtävän kuvailu
käyttäjä_id	kokonaisluku (32-bit)	kertoo kenen käyttäjän tehtävästä on kyse
tärkeysaste_id	kokonaisluku (32-bit)	kertoo tehtävään liittyvän tärkeysasteen

Tehtävät liittyvät aina johonkin käyttäjään. Tehtävillä on kuvaus, jossa kerrotaan mitä käyttäjän pitäisi muistaa tehdä. Tehtävällä voi olla yksi tärkeysaste.

#### Tietokohde: Tärkeysaste:

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku (32-bit)	tärkeysasteen tunnistenumero
nimi	merkkijono, max 50 merkkiä	tärkeysasteen nimi
käyttäjä_id	kokonaisluku (32-bit)	kertoo kenen käyttäjän tärkeysasteesta on kyse

Tärkeysaste liittyy aina johonkin käyttäjään. Tärkeysasteella on nimi, joka kertoo joitain tehtävien tärkeydestä.

**Tietokohde: Luokka:**

Attribuutti	Arvojoukko	Kuvailu
id	kokonaisluku (32-bit)	luokan tunnistenumero
nimi	merkkijono, max 50 merkkiä	luokan nimi
käyttäjä_id	kokonaisluku (32-bit)	mille käyttäjälle luokka kuuluu

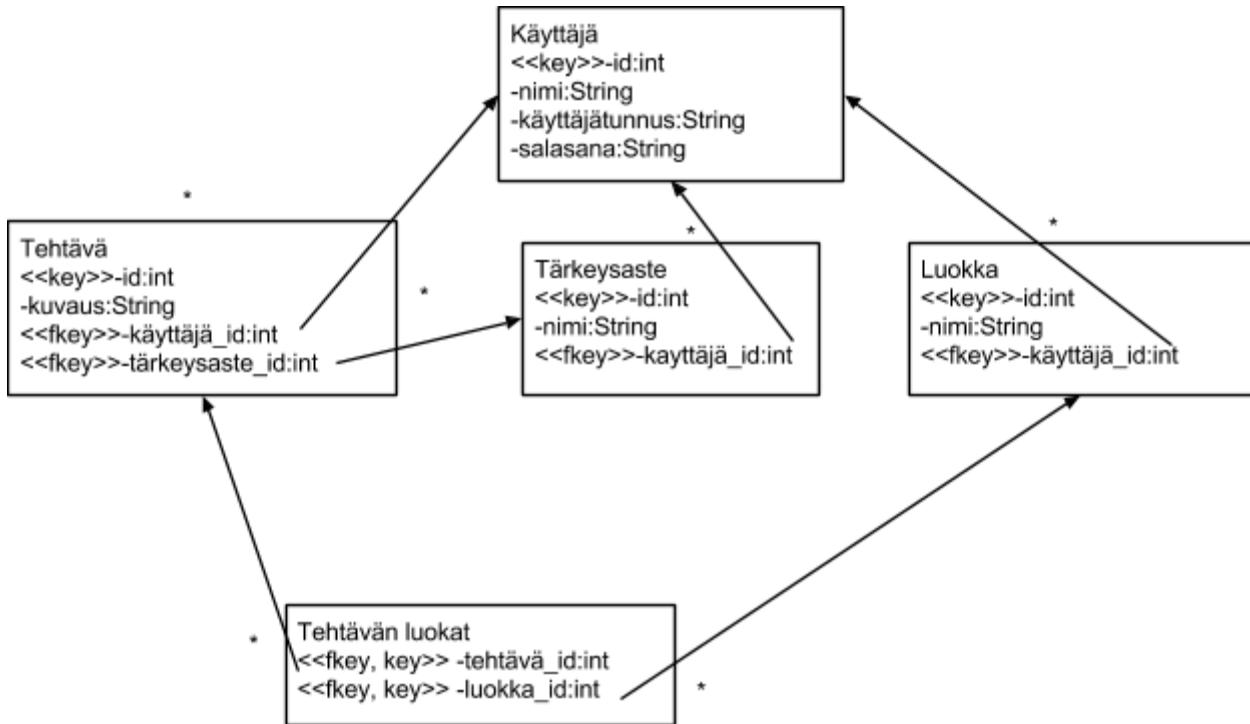
Luokka liittyy aina johonkin käyttäjään. Luokalla on nimi, joka kuvaa luokkaa jotenkin.

**Tietokohde: Tehtävän luokat:**

Attribuutti	Arvojoukko	Kuvailu
tehtävä_id	kokonaisluku (32-bit)	mihin tehtävään luokka kuuluu
luokka_id	kokonaisluku (32-bit)	mihin luokkaan tehtävä kuuluu

Yksi tehtävä voi kuulua useisiin luokkiin ja yhteen luokkaan voi kuulua useita tehtäviä.

## 5. Relaatiotietokantakaavio



## 6. Järjestelmän yleisrakenne

Sovellus noudattaa MVC-mallia. Kontrollerit on sijoitettu projektin juureen, näkymät views-hakemistoon ja mallit libs/models-hakemistoon. Yleiskäytöiset apuluokat löytyvät libs-hakemistosta. Tiedostonimet on kirjoitettu pienellä.

## 7. Järjestelmän komponentit

Tällä hetkellä githubista löytyy myös tiedostoja, joita tässä ei mainita, mutta lopulliseen palautukseen on tarkoitus jättää, vain oikeasti käytössä olevat tiedostot.

### Kontrollerit

login.php

Käsittelee sisäänkirjautumisen varmistamalla, että käyttäjätunnus ja salasana vastaavat toisiaan ja löytyvät käyttäjä-tietokannasta. Kun oikea käyttäjätunnus ja salasana on annettu, käyttäjä

asetetaan session käyttäjäksi ja käyttäjälle näytetään tehtävälista-näkymä. Ilman niitä käyttäjä näkee vain kirjautumisnäkymän.

#### logout.php

Käsittelee uloskirjautumisen poistamalla kirjautuneen käyttäjän tiedot sessiosta ja siirtymällä sisäänkirjautumissivulle.

#### muokkaatehtavaa.php

Käsittelee tehtävänmuokkausnäkymässä annetut kyseistä tehtävää muokkaavat tiedot.

Tarkastaa mallilta, että tarvittavat parametrit on annettu ja käskee mallin sitten muokata tietokantaa. Onnistuneen muokauksen jälkeen siirrytään takaisin tehtävälistaan.

#### tarkeysasteenmuokkaus.php

Näyttää tärkeysasteenmuokkausnäkymän kirjautuneelle käyttäjälle ja välittää myös tiedon kirjautuneesta käyttäjästä ja hänen tärkeysasteistaan.

#### tarkeysasteennimenmuokkaus.php

Näyttää kirjautuneelle käyttäjälle tärkeysasteennimenmuokkausnäkymän ja käsittelee tärkeysasteennimenmuokkausnäkymässä annetut tiedot eli tärkeysasteen uuden nimen. Kysyy tärkeysaste-mallilta, onko nimi kelvollinen ja jos on, pyytää mallia päivittämään sen tietokantaan. Lopuksi siirrytään tärkeysasteenmuokkausnäkymään.

#### tarkeysasteenpoisto.php

Käsittelee tärkeysasteenmuokkausnäkymän tapahtumat, joissa painetaan tärkeysasteen poisto -nappia. Pyytää tärkeysastemallia poistamaan kyseisen tärkeysasteen tietokannasta. Lopuksi siirrytään takaisin tärkeysasteenmuokkausnäkymään.

#### tehtavalista.php

Käsittelee tehtävälistänäkymän tapahtumat, joissa vaihdetaan sivua tai haetaan tehtäviä tietyyn tärkeysasteen perusteella. Täällä voi myös vaihtaa sitä, kuinka monta tehtävää yhdellä sivulla näytetään. Hakee tehtävämallilta annettujen hakuehtojen (sivunumero tai tärkeysaste) mukaisen listan tehtäviä ja antaa sen tehtävälistänäkymälle, jos käyttäjä on kirjautunut sisään.

#### tehtavanmuokkaus.php

Huolehtii tehtävänmuokkausnäkymän esittämisestä. Saa tiedon siitä, mitä tehtävää käyttäjä haluaa muokata ja hakee tämän tehtävän tehtävämallilta ja välittää sen tehtävänmuokkausnäkymälle.

#### tehtavanpoisto.php

Huolehtii tehtävien poistosta, jos käyttäjä painaa tehtävänpoistonappia tehtävälistänäkymässä. Pyytää tehtävämallia poistamaan kyseisen tehtävän sen id:n perusteella tietokannasta ja siirtää käyttäjän lopuksi takaisin tehtävälistaan.

uusitarkeysaste.php

Käsittelee tärkeysasteenmuokkausnäkymästä tulevat uuden tärkeysasteen luotipyynnöt. Jos tärkeysasteennimikenttään on kirjoitettu jotain, pyytää tärkeysastemallia lisäämään tietokantaan sen nimisen tärkeysasteen. Välittää tärkeysastemallille myös tiedon käyttäjästä.

uusitehtava.php

Kysyy tehtävämallilta kyseisen käyttäjän tärkeysasteet ja välittää ne uusitehtävänäkymälle. Ottaa vastaan uusitehtävänäkymästä tulevat tiedot uudesta tehtävästä ja tarkastaa tehtävämallilta niiden kelvollisuuden. Jos ne ovat kunnossa, pyytää tehtävämallia lisäämään tietokantaan kyseisen uuden tehtävän. Onnistuneen tehtävänlisäyksen jälkeen siirtyy tehtävälistanäkymään.

## Näkymät views/

kirjautuminen.php

Näyttää sisäänkirjautumislomakkeen, johon voi syöttää käyttäjätunnuksen ja salasanan.

pohja.php

Näyttää kaikilla sivuilla olevan yläpalkin. Sisäänkirjautumisen jälkeen palkista löytyy linkit tehtävälistaan, uuden tehtävän lisäykseen, tärkeysasteiden muokkaukseen ja uloskirjautumiseen. Huolehtii myös mahdollisten sessioon tallennettujen ilmoitusten ja datataulukkoon tallennettujen virheiden näyttämisestä.

tarkeysasteenmuokkaus\_view.php

Esittää käyttäjän tärkeysasteet taulukkomuodossa. Jokaisen tärkeysasteen kohdalla on nappi, josta tärkeysasteen voi poistaa ja toinen nappi, josta siirtyy sivulle, jossa on mahdollista muokata tärkeysasteen nimeä. Lisäksi sivulla on lomake, jonka avulla voi lisätä uuden tärkeysasteen.

tarkeysasteennimenmuokkaus\_view.php

Näyttää lomakkeen, jossa on mahdollista muuttaa kyseisen tärkeysasteen nimeä.

tehtavalista\_view.php

Näyttää käyttäjän tehtävät taulukkomuodossa. Jokaisen tehtävän kohdalla on myös nappi, josta tehtävän voi poistaa ja toinen josta voi siirtyä kyseisen tehtävän muokkaussivulle. Lisäksi näkymässä on nappulat, joiden avulla voi vaihtaa sivua, jos tehtäviä on useampia sivuja. Näkymässä on myös valikko, josta voi valita vain tietyn tärkeysasteen tehtävät.

tehtavanmuokkaus\_view.php

Näyttää lomakkeen, jonka avulla voi muokata olemassa olevaa tehtävää eli vaihtaa sen tärkeysastetta tai kuvausta.

uusitehtava\_view.php

Näyttää lomakkeen, jonka avulla voi luoda uuden tehtävän eli antaa tehtävälle kuvauksen ja tärkeysasteen.

## Tietokantataulut sql/

add-test-data.sql

Sisältää testaamiseen käytettyä tietokantoihin syötettävää dataa. Tällä hetkellä esimerkiksi uusia käyttäjiä saa vain lisäämällä tiedot tänne.

create-tables.sql

Sisältää tietokantataulujen luomiseen tarvittavat create table -lauseet.

drop-tables.sql

Sisältää tietokantataulujen poistoon tarvittavat lauseet.

## Yleiskäyttöiset tiedostot libs/

common.php

Sisältää yleiskäyttöisiä funktioita, joiden avulla esimerkiksi tarkastetaan, onko käyttäjä kirjautunut, tai siirrytään tietylle sivulle tai näytetään tietty näkymä.

config.php

Sisältää täältä: <http://stackoverflow.com/questions/1475297/phps-white-screen-of-death>  
kopioidun virheidenkäsittelyn, jota on käytetty apuna koodissa olevien virheiden etsimisessä.  
Ilman tästä virheet johtavat lähes aina valkoiseen sivuun.

tietokantayhteys.php

Sisältää tietokantayteyden luomista varten tarvittavat tiedot ja luo yhteyden.

## Mallit libs/models/

tarkeysaste.php

Sisältää tärkeysasteen getterit, setterit, konstruktorin ja funktiot, joiden avulla tietokannasta voi etsiä tärkeysasteen sen id:n perusteella tai käyttäjän id:n perusteella. Sisältää myös funktiot, joilla tärkeysastetauluun voi lisätä uuden tärkeysasteen, poistaa tärkeysasteen tai muokata olemassa olevaa tärkeysastetta. Lisäksi sisältää funktion, jolla voi tarkastaa uuden tärkeysasteen kelvollisuuden.

tehtava.php

Sisältää tehtävän getterit, setterit, konstruktorin ja funktiot, joiden avulla tietokannasta voi etsiä tehtävän sen id:n perusteella tai tehtävät käyttäjän id:n perusteella. Sisältää myös funktiot, joilla tehtävätauluun voi lisätä uuden tehtävän, poistaa tehtävän tai muokata olemassa olevaa tehtävää. Lisäksi sisältää funktion, jolla voi tarkastaa uuden tehtävän kelvollisuuden ja funktiot joilla voi laskea käyttäjän tehtävien lukumäärän tai käyttäjän tiettyyn tärkeysasteeseen liittyvien tehtävien lukumäärän.

## JavaScript js/

bootstrap.js

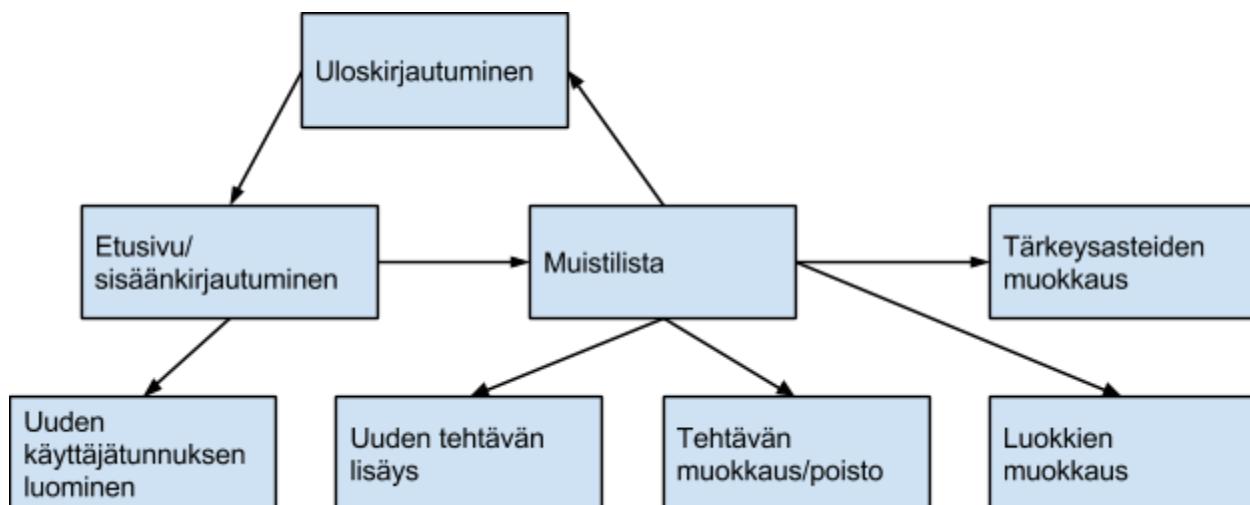
Tarvitaan, jotta tehtävälistanäkymässä oleva pudotusvalikko toimii.

## css-tyylitiedostot css/

bootstrap.css

Parantaa joidenkin näkymien ulkonäköä.

## 8. Käyttöliittymä



Uuden käyttäjätunnuksen luonti -sivulta palataan sisäänkirjautumissivulle. Muilta sivuilta, joista ei lähde nuolia, palataan muistilista-sivulle eli nuolet ovat kaksisuuntaisia kaikissa muissa tapauksissa, paitsi uloskirjautumisessa. Uloskirjautumisen jälkeen palataan

sisäänkirjautumissivulle. Ilman sisäänkirjautumista näkee vain sisäänkirjautumissivun ja uuden käyttäjätunnusluonti -sivun.

## 9. Asennustiedot

Sovellus on nyt tallennettu laitoksen users-palvelimen htdocs-hakemistoon, missä se on näkyvissä nettiin. Sovelluksen voi asentaa kopioimalla tiedostot kyseisestä paikasta esimerkiksi omaan htdocs-hakemistoonsa. Sovellus vaatii PostgreSQL-tietokannan, jonka saa laitoksen usersilla käyttöön skriptin (wanna-postgres) avulla. Lopuksi tietokannan yhteystiedot on muokattava oikeiksi tiedostoon libs/tietokantayteys.php.

## 10. Käyttöohje

Sovelluksen sisäänkirjautuminen löytyy osoitteesta

<http://imberg.users.cs.helsinki.fi/Muistilista/login.php>. Sovellusta voi testata kirjautumalla sisään seuraavilla tiedoilla:

Käyttäjätunnus: testi

Salasana: salasana

Sisäänkirjautumisen jälkeen voi tarkastella tehtävälistaa. Tehtäviä voi lisätä yläreunassa olevasta Lisää uusi tehtävä -napista. Sen vieressä olevasta Kirjaudu ulos -nappulasta voi kirjautua ulos.

## 11. Testaus, puutteet ja jatkokehitysideoat

## 12. Omat kokemukset

Sovelluksen tekoon oli varattu niin vähän aikaa, että jatkuvasti piti todella miettiä, mitä kannattaa edes yrittää toteuttaa. Tiesin kyllä, ettei harjoitustyökurssien kanssa kannattaisi ottaa samanaikaisesti muita kursseja, mutta koska tästä kurssista ei saa 15 opintopistettä, oli muitakin kursseja pakko ottaa sopivasti, jotta pääsi lähemmäs tuota vaadittua opintopistemäärää. Tällä kurssilla tuntikirjanpito olisi ollut mielenkiintoista, sillä tuntuu että kurssiin sai käyttää loputtomasti aikaa, mutta silti lopputulos on surkea. Aikaa kului hyvin paljon myös dokumentin kirjoittamiseen, kuvien piirtelyyn, koodikatselointiin ja muuhun sellaiseen. Näissäkin kaikissa olisi pitänyt varmaan vain katsoa, mistä voi saada eniten pisteytä, tehdä se ja unohtaa muu. Voisiko esimerkiksi tästä dokumentoinnista karsia muutaman kohdan, jotta varsinaiselle koodaamiselle jäisi edes pari tuntia enemmän aikaa.

## 13. Liitteet

Tehtäväankuvaus löytyy täältä: <http://advancedkittenry.github.io/suunnittelu/aiheet/Muistilista.html>