

# Muistilista

Tietokantasovelluksen harjoitustyö 2017  
Helsingin Yliopisto Tietojenkäsittelytieteenlaitos

Ville Huotarinen  
02.02.2017

<b>1. Johdanto</b>	<b>3</b>
<b>2. Käyttötapaukset</b>	<b>4</b>
2.1 Käyttötapauskaavio	4
2.2 Käyttötapaukset	4
2.3 Käyttötapauskuvaukset	5
<b>3. Järjestelmän tietosisältö</b>	<b>6</b>
3.1 Operator	6
3.2 Task	6
3.3 Priority	7
3.4 Task_category	7
3.5 Category	7
<b>4. Relaatietietokantakaavio</b>	<b>9</b>
<b>5. Käynnistys- ja käyttöohje</b>	<b>10</b>
<b>6. Järjestelmän yleisrakenne</b>	<b>11</b>
<b>7. Käyttöliittymä ja järjestelmän komponentit</b>	<b>12</b>
<b>8. Asennustiedot</b>	<b>13</b>

# 1. Johdanto

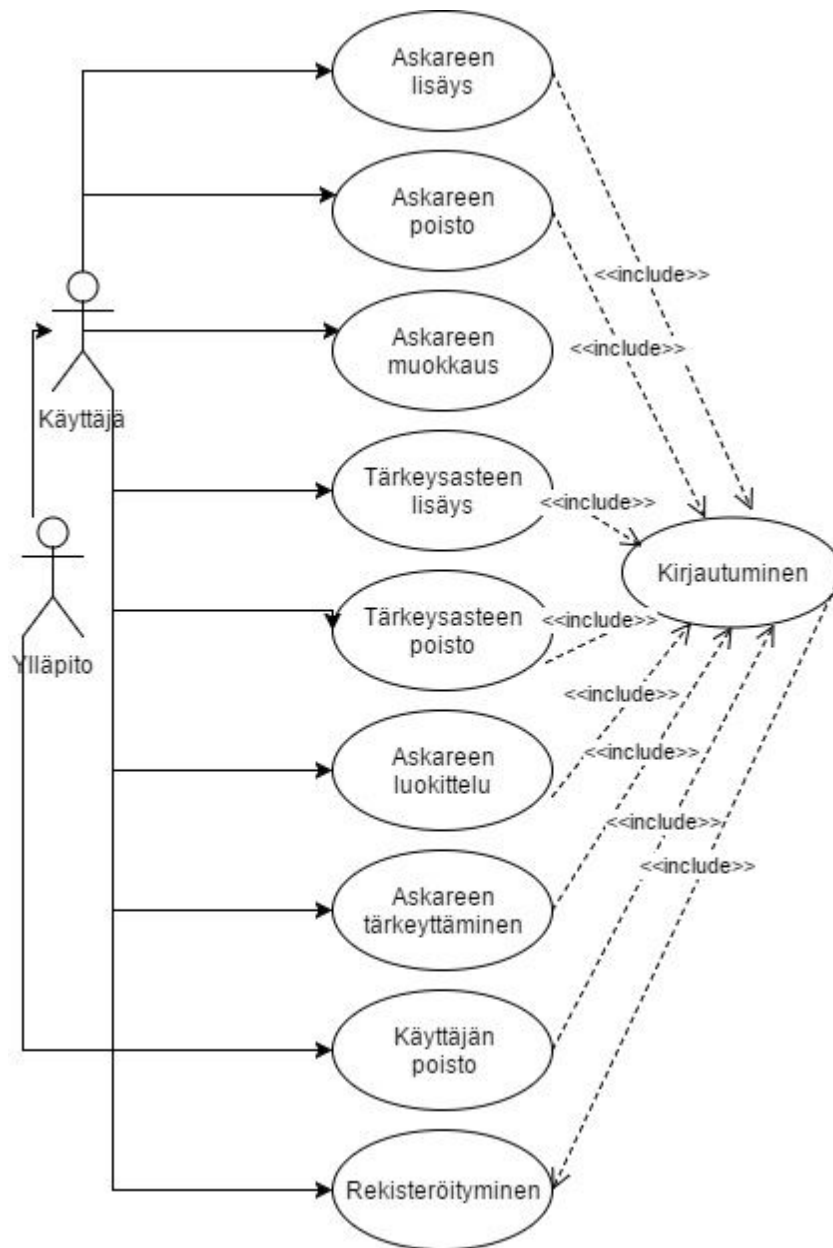
Ohjelman tarkoituksena on avustaa ihmisiä arkiaskareissa pitämällä kirjaa erilaisista tehtävistä, mitä henkilö haluaisi muistaa, ja saada ehkäpä tehdyksi. Ohjelmalla voidaan luoda, poistaa ja muokata tehtäviä.

Käyttäjä voi kirjautumisen jälkeen tarkastella omia listattuja askareitaan, sekä lisätä ja muokata niitä tarvittaessa. Käyttäjä voi myös luokitella askareitaan tärkeysjärjestykseen. Tavoitteena on lisätä käyttäjien tehokkuutta arjessa toimimalla avustavana muistina ja muistuttajana.

Työ on selainpohjainen ja se toteutetaan laitoksen users-palvelimella. Ohjelmointikielenä toimii PHP, joten käyttäjän selaimen täytyy tukea sitä.

## 2. Käyttötapaukset

### 2.1 Käyttötapauskaavio



### 2.2 Käyttötapaukset

**Ylläpitäjä:** Käyttäjä, jolla oikeuksia normaalia käyttäjää enemmän. Mahdollistaa sovelluksen ylläpidon.

**Käyttäjä:** Sovellusta käyttävä rekisteröitynyt henkilö.

## 2.3 Käyttötapauskuvaukset

**Askareen lisääminen, muokkaaminen ja poisto:**

Käyttäjä voi lisätä muistilistalle askareen, muokata/poistaa sen.

**Tärkeysasteen lisäys ja poisto:**

Käyttäjä voi priorisoida muistilistalla olevia askareitaan lisäämällä niille tärkeysasteen.

Tärkeysasteen voi myös poistaa.

**Askareen luokittelu:**

Käyttäjä voi luokitella askareitaan erilaisiin kategorioihin.

**Luokkien luominen ja poisto:**

Käyttäjä voi luoda tai poistaa halutessaan luokkia, jotka ovat henkilökohtaisia.

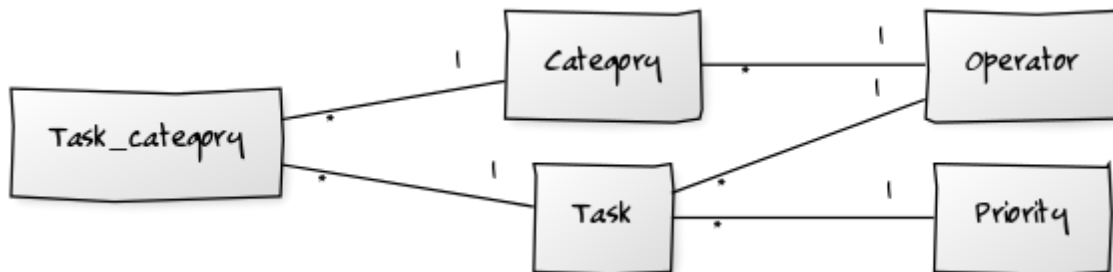
**Käyttäjän muokkaus:**

Ylläpitäjällä on oikeudet poistaa käyttäjän tunnukset.

Ylläpitäjälle kuuluvat myös kaikki käyttäjän käyttötapauskset.

Muita käyttötapausksia ovat rekisteröityminen ja kirjautuminen.

### 3. Järjestelmän tietosisältö



#### 3.1 Operator

Atribuutti	Arvojoukko	Kuvailu
id	serial	id
name	merkkijono, max 50. merkkiä	Kirjautumiseen käytettävä tunnus
password	merkkijono, max 50. merkkiä	Kirjautumiseen käytettävä salasana
status	integer, max 50. merkkiä	Arvo 1 jos ADMIN ja 0 jos käyttäjä eli USER

Operator on rekisteröitynyt ohjelman käyttäjä. Käyttäjä voi luoda itselleen askareita (Task). Käyttäjän status voi olla joko ADMIN tai USER. ADMIN-statuksen omaava käyttäjä voi selata ja poistaa muita käyttäjiä.

#### 3.2 Task

Atribuutti	Arvojoukko	Kuvailu
id	serial	id

name	merkkijono, max. 120 merkkiä	Askareen kuvaus
date	päivämäärä	Milloin askare on lisätty
deadline	päivämäärä	Askareen määräaika

Task on käyttäjän luoma askare. Käyttäjällä voi olla monia askareita, ja ne voivat kuulua moneen luokkaan (Category), mutta sillä voi olla vain yksi tärkeysaste (Priority).

### 3.3 Priority

Atribuutti	Arvojoukko	Kuvailu
id	serial	id
name	merkkijono, max. 120 merkkiä	Tärkeysasteen nimi

Priority on askareen tärkeysaste. Tärkeysasteella voi olla useampi askare (Task).

### 3.4 Task\_category

Atribuutti	Arvojoukko	Kuvailu
task_id	kokonaisluku	Viittaus Task-luokkaan
category_id	kokonaisluku	Viittaus Category-luokkaan

Task\_category on liitostaulu askareiden (Task) ja askareen luokan (Category) välillä.

### 3.5 Category

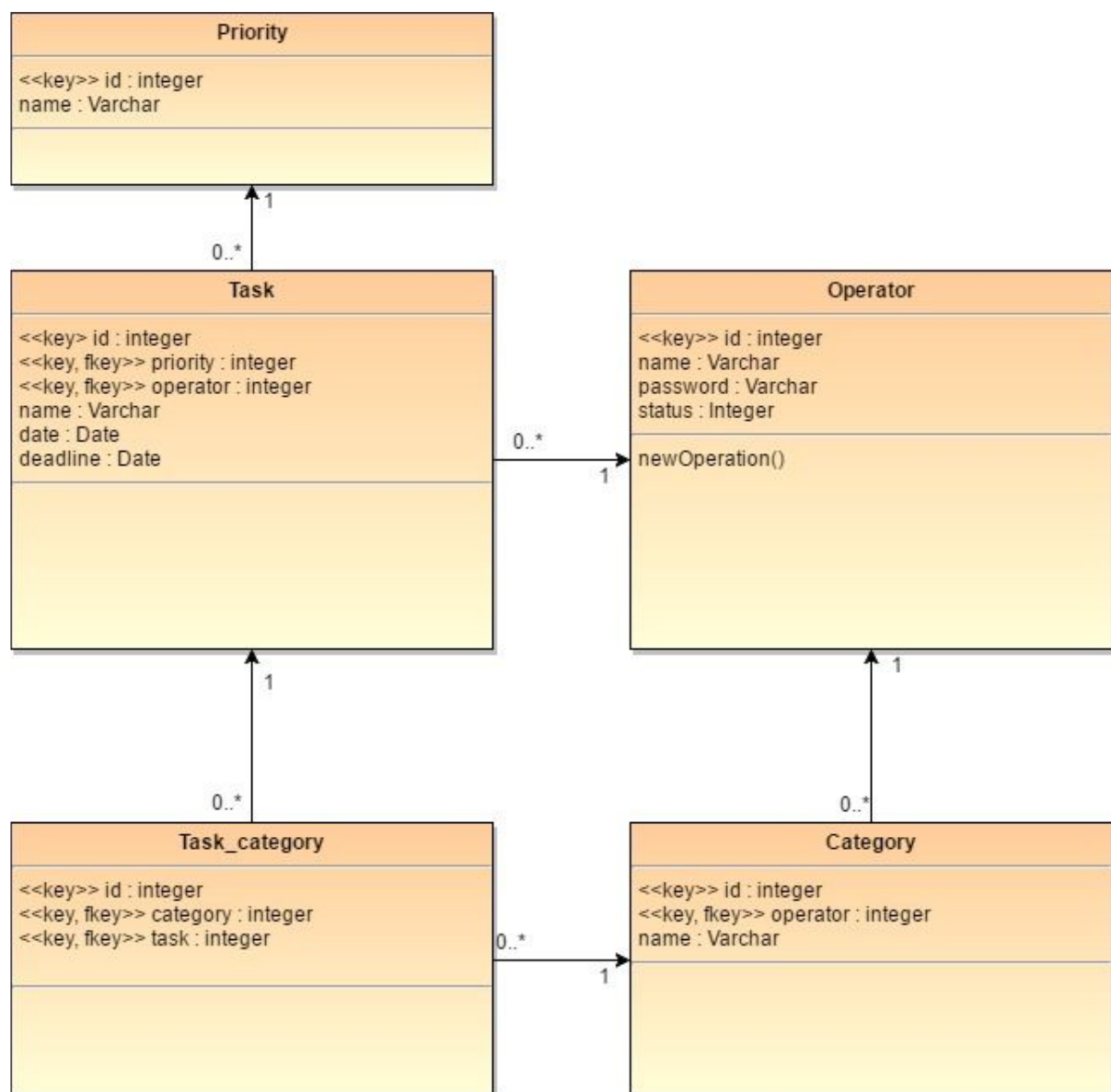
Atribuutti	Arvojoukko	Kuvailu
------------	------------	---------

id	serial	id
name	merkkijono, max. 50 merkkiä	Luokan kuvaus

Category on askareen luokka. Luokalla voi olla monta askareta, ja askareella voi olla monta luokkaa. Näiden välillä on liitostaulu Task\_category.



## 4. Relaatiotietokantakaavio



## 5. Käynnistys- ja käyttöohje

Ohjelman löytää osoitteesta <http://vihu.users.cs.helsinki.fi/tsoha>. Etusivulla voidaan kirjautua käyttäjätunnuksella ja salasanalla. Jos käyttäjätunnusta ja salasanaa ei ole, voit luoda nämä sivun yläpalkista 'Register'-nappia painamalla.

Ohjelmaa voidaan testata käyttämällä käyttäjätunnusta 'admin' ja salasanaa 'admin', joka on myös ohjelman moderointitili.

Kirjautuneena yläpalkista voidaan tarkastella omia 'taskeja' eli tehtäviä. Lisäksi voidaan kirjautua ulos 'logout'. Admin tunnuksilla voidaan myös selata/poistaa muita käyttäjiä.

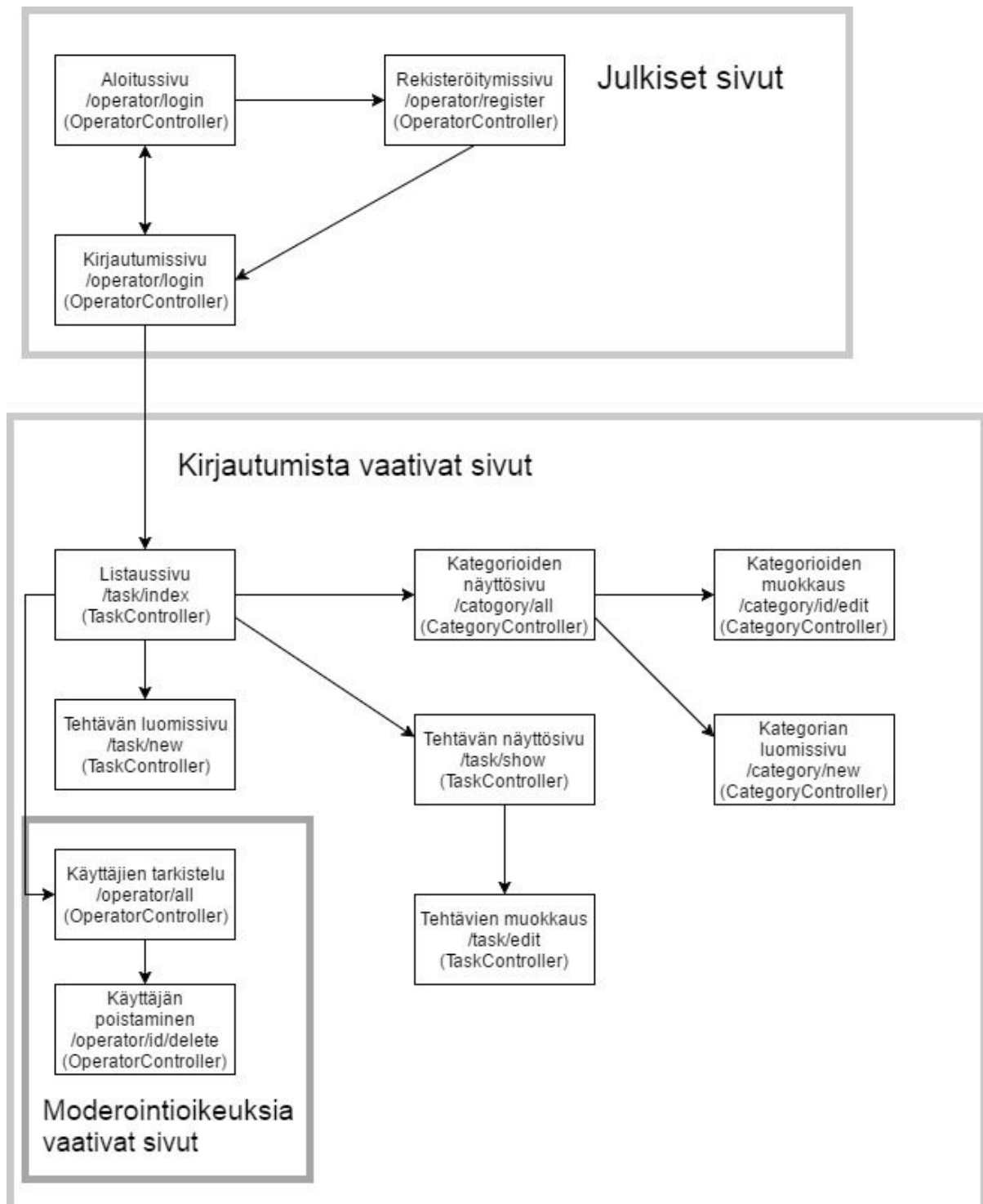
## 6. Järjestelmän yleisrakenne

Muistilista tietokantasovellusta tehdessä on noudatettu MVC-mallia. Kontrollerit, näkymät ja mallit sijaitsevat app-hakemiston alihakemistoissa: controllers, views ja models. Views hakemistosta löytyy näkymille kuuluvat alihakemistot, ja yleiskäytössä olevat sivut löytyvät sen juuresta.

Muistilistan kaikki muut tiedostot on kirjoitettu pienellä, paitsi malliluokat camelCasella.

Sovellusta päästään käyttämään kirjautumisen jälkeen, ja ennen sitä mahdollista on tehdä uudet tunnukset tai kirjautua. Admin-tunnuksilla saadaan näkyviin ylläpitoon liittyviä toimintoja, kuten käyttäjätunnusten hallinnointia.

## 7. Käyttöliittymä ja järjestelmän komponentit



## 8. Asennustiedot

Lataa sovellus aluksi osoitteesta <https://github.com/vipeeri/Tsoha-Bootstrap/> käyttämällä GitHubin Fork-toimintoa.

Esimerkissä kerron asennusohjeet Helsingin yliopiston users-palvelimelle, käyttäen Ubuntu-käyttöjärjestelmää ja ssh-yhteyttä.

Aloitetaan luomalla työkansiomme htdocs, joka onnistuu terminaalissa komennolla `wanna-htdocs`. Tähän kansioon tallennamme kaikki sovellukseen kuuluvat tiedostot.

Tarvitsemme sovelluksen käyttöä varten myös tietokantapalvelimen, tällä kertaa käytössämme on PostgreSQL. Jos PostgreSQL on asennettu palvelimelle, saamme sen käyttöön komennolla `wanna-postgres`.

Nyt löydämme kansioista config tiedoston `environment.sh`, johon muokkaamme oman käyttäjänimen sekä projektin kansion. Minulla nämä ovat vihu ja tsoha.

Jotta voimme luoda projektikansion ja siirtää tiedostot palvelimelle, ajamme työkansion juuressa komennon `bash bootstrap.sh`. Nyt meillä on käytössä osoite: <http://vihu.users.cs.helsinki.fi/tsoha/>

Jotta voit tarkastella tekemiäsi muutoksia tiedostoihin, täytyy terminaalissa ajaa komento `bash deploy.sh`, joka päivittää muutokset myös palvelimelle.

Tietokantoja varten on komennot `bash create_tables.sh` (taulujen luonti) ja `bash add_test_data.sh` (ajaa palvelimelle taulujesi testidatan).