

# **memeDB – meemitietokanta**

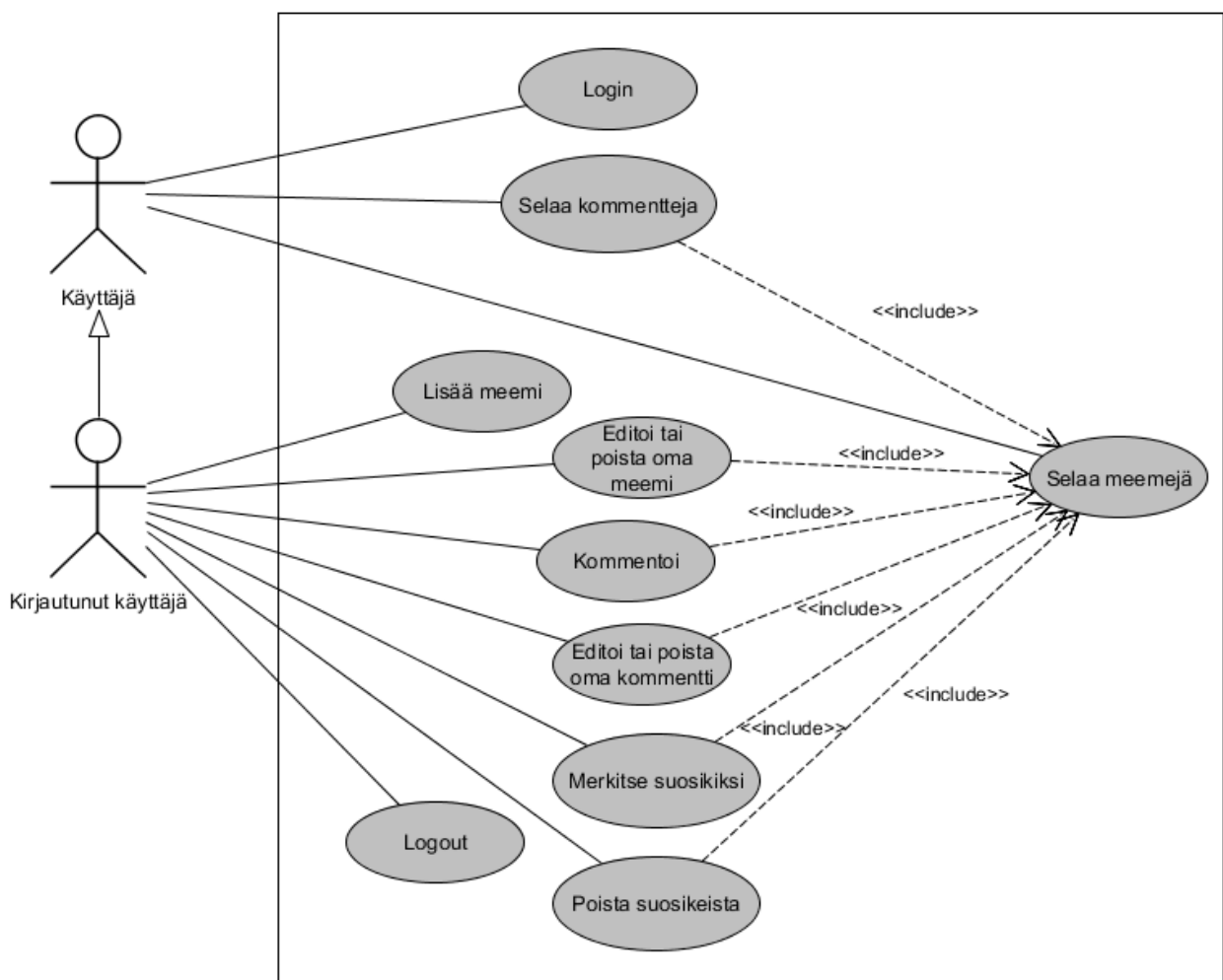
Aineopintojen harjoitustyö: Tietokantasovellus

# 1. Johdanto

Tarkoituksena on luoda PHP:llä järjestelmä erilaisten hauskojen internetistä löytyvien kuvien, tekstien ja videoiden varastointiin, jakamiseen ja kommentointiin. Kuvista säilytetään vain linkit palvelimen muistin säästämiseksi. Tarkoituksena ei ole perustaa sivustoa kuvien hostaamiseen.

Järjestelmä pyörii TKTL:n users-palvelimella ja käyttää samalla palvelimella sijaitsevaa PostgreSQL tietokantaa tarvittavien linkkien, käyttäjätietojen jne. säilömiseen. Sovelluksen rakentamiseen tullaan käyttämään Slim reitityskirjastoa, Twig sivupohjamootoria ja Bootstrap frameworkkia sekä JavaScriptiä. Syötteiden validointiin käytetään Valitron kirjastoa ja riippuvuushallintaan käytetään Composeria.

## 2. Yleiskuva järjestelmästä



Käyttötapauskaavio

## Käyttäjäryhmät

### Käyttäjä

Oletusryhmä, johon myös muut ryhmät kuuluvat. Käyttäjä on tässä tapauksessa henkilö, joka tulee sivulle, eikä välttämättä ole kirjautunut. Pystyy selaamaan sivuston sisältöä.

### Kirjautunut käyttäjä

Käyttäjä, joka on kirjautunut sivustolle. Pystyy mm. tuomaan sivustolle omaa sisältöään ja muokkaamaan sitä.

## Käyttötapauskuvaukset

### Käyttäjä:

Meemien selaaminen:

Käyttäjä navigoi sivuston osaan missä meemit ovat listattuna, jolloin käyttäjällä näytetään lista kaikista tietokannassa olevista meemeistä.

Kommenttien selaaminen:

Käyttäjä valitsee meemilistasta haluamansa meemin, jonka omalla sivulla käyttäjälle näytetään meemi ja siihen liittyvät kommentit.

Muut käyttötapauskuvaukset: kirjautuminen.

### Kirjautunut käyttäjä:

Meemien lisääminen:

Käyttäjä navigoi sivun osalle, jolla meemien lisääminen tietokantaan tapahtuu, jonka jälkeen käyttäjä syöttää sivuston tekstikenttiin meemityypille olennaiset tiedot ja painaa lisää-nappia.

Omien meemien editoiminen ja poistaminen:

Käyttäjä navigoi jonkin lisäämänsä meemin sivulle ja painaa editoi-nappia. Tämän jälkeen käyttäjälle näytetään sivu, jossa kohdetta voidaan muokata tai poistaa kokonaan.

Kommentoiminen:

Käyttäjä navigoi jonkin meemin sivulle, syöttää tekstikenttään kommenttinsa ja painaa lähetä-nappia.

Omien kommenttien editoiminen ja poistaminen:

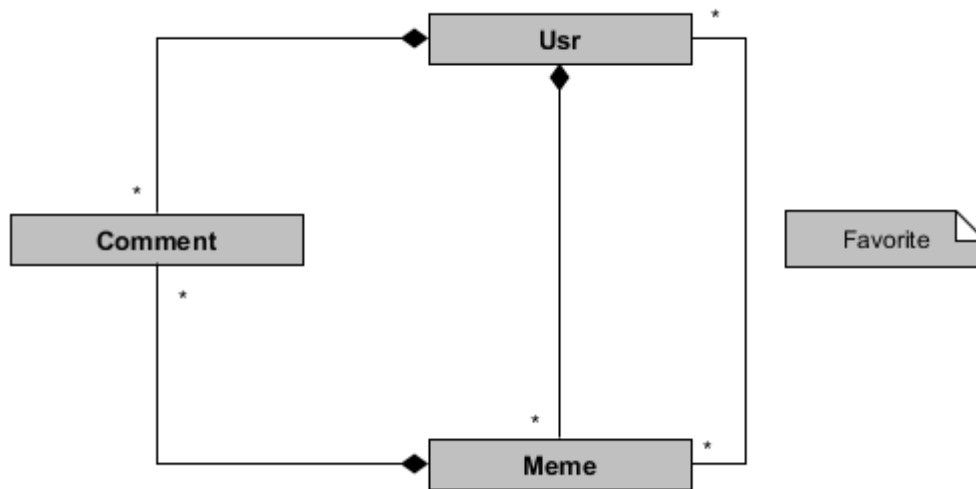
Käyttäjä navigoi haluamansa kommentin luokse ja painaa editoi-nappia. Tämän jälkeen käyttäjälle näytetään sivu, jossa kohdetta voidaan muokata tai poistaa kokonaan.

Suosikkeihin lisääminen:

Käyttäjä navigoi haluamansa meemin sivulle ja painaa "Favourite" -nappia, jonka jälkeen kyseinen meemi näytetään tulevaisuudessa käyttäjän suosikit-listassa.

Muut käyttötapauskuvaukset: uloskirjautuminen, suosikeista poistaminen.

### 3. Järjestelmän tietosisältö



Käsitekaavio

Tietokohde: Usr

Attribuutti	Arvojoukko	Kuvaus
Username	Merkkijono, max 20 merkkiä	Käyttäjän nimi
Password	Merkkijono, max 50 merkkiä	Käyttäjän salasana

Käyttäjään voi liittyä useita meemejä ja kommentteja sekä käyttäjällä voi olla useita meemejä suosikkinaan. Jos käyttäjä tuhoetaan, tuhoutuu kaikki käyttäjän liittyvä tämän mukana.

Tietokohde: Meme

Attribuutti	Arvojoukko	Kuvaus
Title	Merkkijono, max 50 merkkiä	Meemin otsikko
Type	Merkkijono, max 10 merkkiä	Meemin tyyppi. Mahdollisia tyypejä ovat cypypasta, image ja video.
Content	Merkkijono, max 5000 merkkiä	Copypastalle tämä on pelkää tekstiä, kuville linkki kuvaan ja videoille YouTube:n videoid.

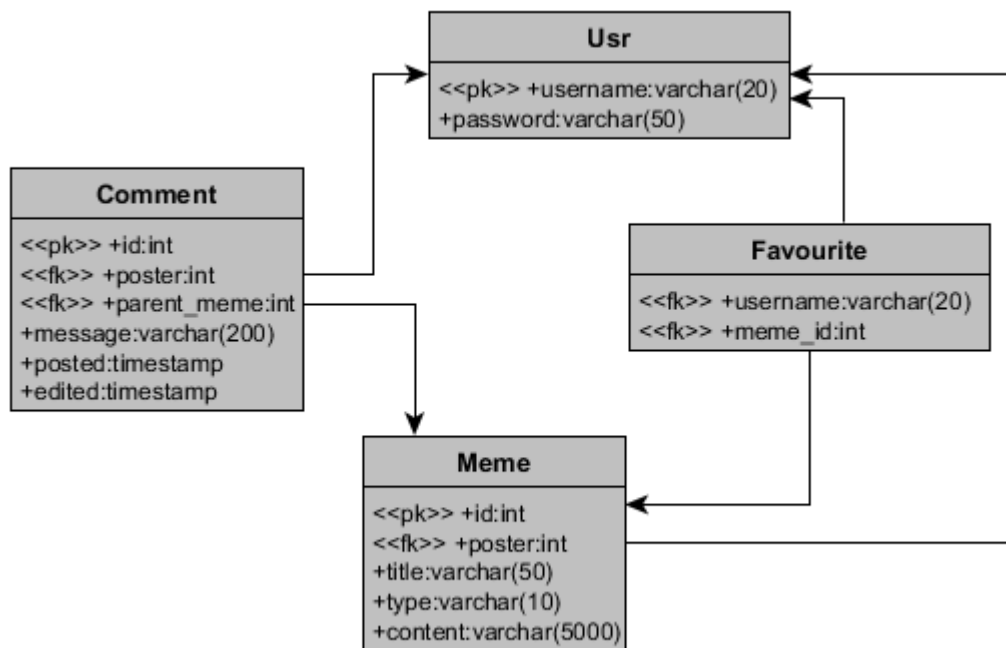
Jokaisella meemillä on yksi käyttäjä kohteen luojana. Meemin voi liittyä useita kommentteja ja meemit voivat olla usean käyttäjän suosikkeja. Jos meemi poistetaan, tuhoutuvat siihen liittyvät kommentit ja suosikkisuhteet. Jokaisella käyttäjällä voi myös olla useita suosikkimeemejä, mikä synnyttää monesta-moneen -suhteen kohteiden välille.

Tietokohde: Comment

Attribuutti	Arvojoukko	Kuvaus
Message	Merkkijono, max 200 merkkiä	Viesti itse
Posted	Aikaleima	Lähetämisajankohta
Edited	Aikaleima	Viimeisin editoimisajankohta. Tyhjä jos ei ikinä editoitu.

Kommenteilla on yksi käyttäjä tekijänä ja yksi meemi isäntänä.

## 4. Relaatietietokantakaavio



Relaatietietokantakaavio

## 5. Järjestelmän yleisrakenne

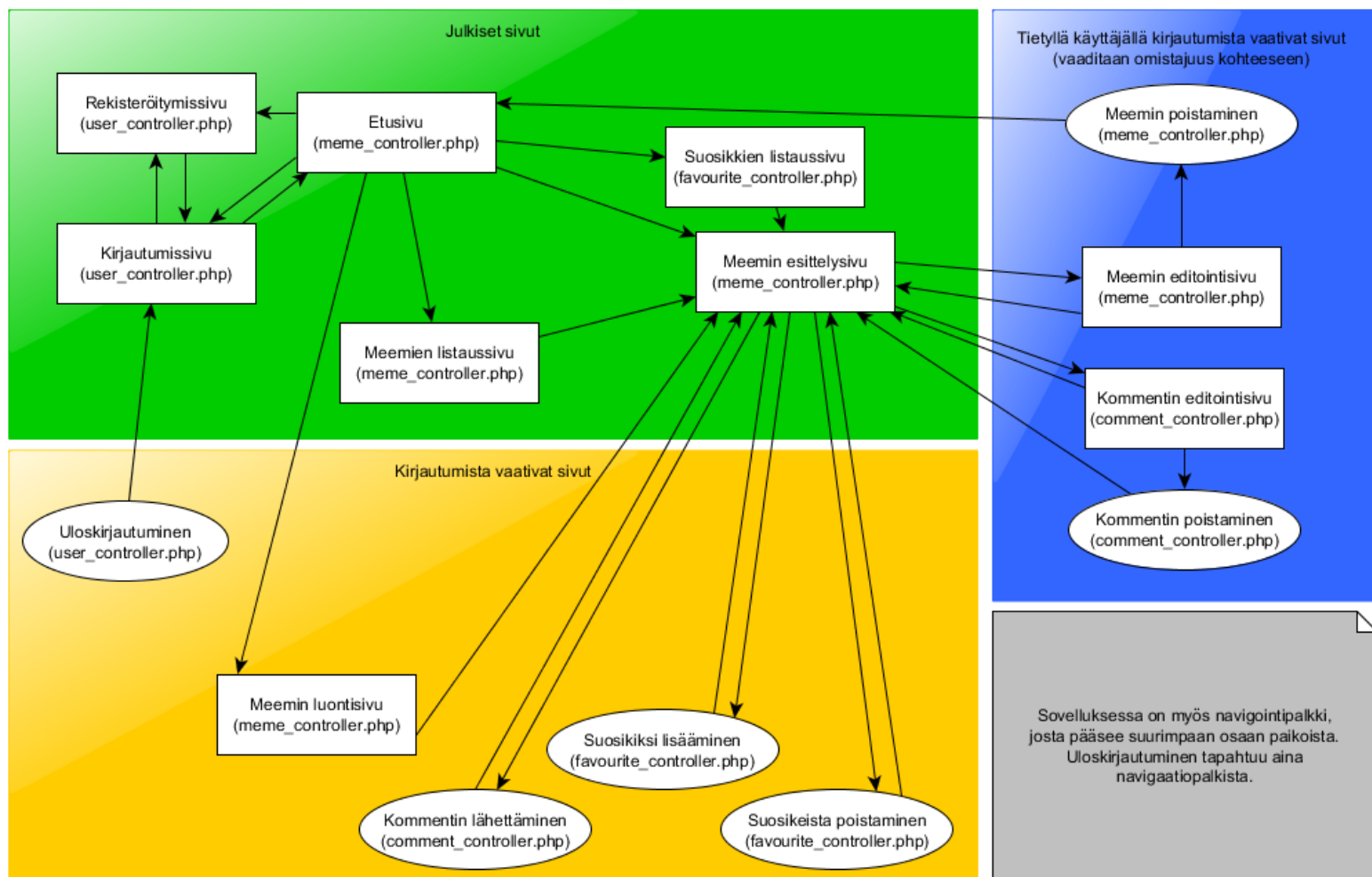
Ohjelma noudattaa MVC-mallia eli ohjelman toiminnot on jaettu kontrollereihin, näkymiin ja malleihin, jotka sijaitsevat kansioissa controllers, views ja models. Edellä mainitut kansiot sijaitsevat app kansiossa. Php:n apu- ja pohjaluokat ovat lib kansiossa ja html:n pohjatiedosto on views kansiossa. Varsinaista asetustiedostoa ei ole, mutta sivulla kerralla näytettävien kohteiden lukumäärää voi muuttaa muuttamalla `entries_per_page` -muuttujaa `base_model.php` tiedostossa. CSS-tyylitiedosto ja JavaScript skriptitiedosto sijaitsevat assets hakemistossa omissa kansioissaan.

Kontrollerien jne. lisäksi järjestelmässä pieni JavaScript-komponentti, jolla esitetään käyttäjälle varmistusviesti meemejä ja kommentteja poistaessa.

Istuntoa järjestelmässä käytetään kirjautuneen käyttäjän tunnistamiseen, jotta kirjautumattomien ja kirjautuneiden käyttäjien pääsy heille kuulumattomiin resursseihin, kuten toisten käyttäjien sisällön muokkaamiseen, pystytään estämään.

Tiedostojen nimet on kirjoitettu pienellä, kuten metodienkin nimet. Luokkien nimet on kirjoitettu CamelCasella samoin kuten kaikki JavaScriptiin liittyvä.

## 6. Käyttöliittymä ja järjestelmän komponentit



## 7. Asennustiedot

Sovellus on tällä hetkellä asennettuna TKTL:n users-palvelimelle, jolle on otettuna käyttöön php ja psqL tuki.

Toiseen paikkaan users-palvelimella asentamiseen riittää environment.sh:n muuttaminen sekä bootstrap.sh skriptin suorittaminen (kunhan ssh jne. on valmisteltu käytettäväksi). Tämän jälkeen sovellusta voidaan päivittää deploy.sh skriptillä, joka siirtää tarvittavat tiedostot palvelimelle. Uusia kirjastoja lisättäessä täytyy ajaa, joko bootstrap.sh tai Composerin omat vaadittavat komennot. Lisäksi mukana on skriptit tietokannan alustamiselle ja testidatan lisäämiselle. Myös nämä tulee ajaa ennen käyttöönottoa.

Muille palvelimelle asennettaessa palvelimelle olennaiset tiedostot/kansiot ovat app, config, lib, vendor, sql, assets, index.php, ja composer.json. Jos haluaa käyttää yllä mainittuja valmiita skriptejä asennuksen hallintaan ja päivittämiseen tulee ne muokata ympäristöön sopiviksi. Tietokantayhteyden konfiguraatio tapahtuu config/database.php ja lib/database.php tiedostoissa. Lisäksi myös tietokannan alustamiseen käytetyt skriptit täytyy muokata sopiviksi ja ajaa tai vaihtoehtoisesti luoda kantaan create\_tables.sql tiedoston mukaiset taulut.

## 8. Käynnistys- / käyttöohje

Tällä hetkellä ohjelma pyörii TKTL:n users-palvelimella osoitteessa <http://sobackr.users.cs.helsinki.fi/memeDB/> .

Kaikki lisäys- ja muokkaustoiminnot ovat kirjautumisen takana ja vain käyttäjän tunnuksella luotuja kohteita on mahdollista muokata kyseisellä tunnuksella.

Sisään pääsee kirjautumaan osoitteessa <http://sobackr.users.cs.helsinki.fi/memeDB/login> tai etusivulla olevaa log in -nappia painamalla.

Testaamiseen tarkoitettuja tunnuksia ovat (formaatissa username, password):

- User, User
- User2, User2
- User3, User3
- User4, User4