
TIETOKANTASOVELLUS KESKUSTELUFOORUMI

Johdanto

Projektina toteutetaan keskustelufoorumi käyttäen PHP-kieltä ja postgresql-tietokantaa. Järjestelmän tulisi mahdollistaa keskustelujen käyminen erillisissä viestiketjuissa sekä viestien lainaamisen ja muokkaamisen. Järjestelmän on pystyttävä näyttämään mitä ketjuja käyttäjät ovat jo lukeneet ja onko näihin ketjuihin tullut uusia viestejä. Ketjut mahdollisesti jaetaan myös erillisiin alueisiin keskusteluaiheiden mukaan.

Foorumi vaatii käyttäjältä käyttäjätunnukset jotta viestejä voidaan kirjoittaa tai muokata. Järjestelmän on näin mahdollistettava sekä uusien tunnusten luonti että kirjautuminen sisään vanhoilla tunnuksilla. Uusien tunnusten luonti ei vaadi sähköpostivarmistusta tai captcha-kyselyä toteutuksen yksinkertaistamiseksi.

Järjestelmän tukee myös tukea eritasoisia käyttöoikeuksia – normaali käyttäjä voi muokata vain omia viestejään kun taas moderaattorin tulee pystyä muokata kaikkia viestejä ja tarvittaessa poistaa viestejä. Tarvittaessa käyttäjälle voi asettaa porttikiellon jolloin hän ei enää voi osallistua keskusteluun. Järjestelmällä on myös erilliset hallinto-tunnukset (admin) jotka voivat asettaa tai poistaa moderaattoreja ja admineja.

Foorumi toteutetaan laitoksen users-kansioon käyttäen Apache-palvelinta. Javascriptiä käytetään hieman esimerkiksi vahvistusikkunoiden näyttämiseen viestien poiston yhteydessä. Lisäksi projektissa käytetään Bootstrap-css-kirjastoa ulkonäyn toteutukseen sekä jquery-kirjastoa asynkronisten operaatioiden mahdollistamiseksi.

Järjestelmän yleiskuva

Käyttäjryhmiä ovat tunnuksen käyttäjä, peruskäyttäjä, moderaattori ja ylläpitäjä eli admin.

Tunnuksen käyttäjä on käyttäjä joka selaa foorumia kirjautumatta sisään. Nämä käyttäjät voivat vain lukea olemassa olevia ketjuja, luoda uuden käyttäjätunnuksen tai kirjautua sisään olemassa olevilla tunnuksilla. Lisäksi tunnuksen käyttäjä voi hakea foorumilta viestejä joko käyttäjätunnuksen tai avainsanojen avulla haluamaltaan aikaväliltä.

Peruskäyttäjä on henkilö joka on luonut käyttäjätunnukset ja kirjautunut sisään. Hän voi lukemisen lisäksi luoda uusia viestiketjuja, kirjoittaa uusia viestejä sekä muokata omia vanhoja viestejään. Peruskäyttäjä ei voi luoda uusia tunnuksia tai kirjautua sisään koska hän on jo kirjautunut sisään; peruskäyttäjän on kirjauduttava ulos jotta nämä ominaisuudet tulevat käyttöön. Peruskäyttäjä voi myös lähettää yksityisviestejä muille käyttäjille.

Moderaattori on käyttäjä jolla on samat käyttötapaukset kuin peruskäyttäjällä. Lisäksi moderaattori voi muokata kaikkia viestejä sekä poistaa viestejä tai viestiketjuja. Moderaattori voi antaa myös käyttäjälle porttikiellon. Moderaattorit voivat myös luoda uusia keskustelualueita tai muokata tai poistaa olemassa olevia alueita.

Ylläpitäjä on käyttäjä jolla on samat käyttötapaukset kuin moderaattorilla. Lisäksi ylläpitäjä voi lisätä tai poistaa moderaattoreja ja admineja.

Porttikiellon saanut käyttäjä vastaa tasoltaan tunnuksenonta käyttäjää.

Käyttötapauskaavio:



Lisäksi admin on automaattisesti myös moderaattori
sekä moderaattori on automaattisesti peruskäyttäjä

Käyttötapaukset

Käyttäjätilin luominen

1. Käyttäjä painaa käyttäjätilin luomislinkkiä.
2. Järjestelmä pyytää käyttäjältä käyttäjänimen, sähköpostitunnuksen ja salasanan kahdesti.
3. Käyttäjä antaa järjestelmän pyytämät tiedot.
4. Järjestelmä luo tietokantaan uuden käyttäjän annetuilla tiedoilla ja ilmoittaa tästä käyttäjälle.

Virhetilanne: Käyttäjätunnus on jo käytössä

1. Kuten edellä
2. Kuten edellä.
3. Kuten edellä.
4. Järjestelmä havaitsee että kyseinen käyttäjätunnus on jo käytössä ja antaa virheilmoituksen.
5. Prosessi palaa kohtaan 2.

Virhetilanne: Salasanakentät eivät vastaa toisiaan

1. Kuten edellä
2. Kuten edellä
3. Käyttäjä syöttää käyttäjätunnuksen ja salasanat, mutta salasanat eroavat toisistaan.
4. Järjestelmä havaitsee että salasanakentät eivät vastaa toisiaan ja antaa virheilmoituksen.
5. Prosessi palaa kohtaan 2

Sisäänkirjautuminen

1. Käyttäjä painaa sisäänkirjautumislinkkiä.
2. Järjestelmä pyytää käyttäjältä käyttäjätunnusta ja salasanaa.
3. Käyttäjä syöttää käyttäjätunnuksensa ja salasanansa.
4. Järjestelmä kuittaa kirjautumisen ja ohjaa käyttäjän takaisin aloitussivulle.

Virhetilanne: Käyttäjätunnusta tai salasanaa ei annettu

1. Kuten edellä
2. Kuten edellä
3. Käyttäjä ei anna käyttäjänimeä, salasanaa tai kumpaakaan.
4. Järjestelmä havaitsee että joko käyttäjätunnus- tai salasanakenttä tai molemmat ovat tyhjiä.
5. Järjestelmä antaa tilanteeseen sopivan virheilmoituksen.
6. Prosessi palaa kohtaan 2.

Virhetilanne: Käyttäjätunnus tai salasana on virheellinen

1. Kuten edellä
2. Kuten edellä
3. Käyttäjä syöttää virheellisen käyttäjätunnuksen tai salasanan.
4. Järjestelmä huomaa että joko salasana on virheellinen tai käyttäjätunnusta ei löydy.
5. Järjestelmä antaa virheilmoituksen. Virheilmoituksessa ei täsmennetä virheen tarkkaa syytä jotta kirjautumisyrityksillä ei voi kalastella olemassa olevien käyttäjätunnusten nimiä.
6. Prosessi palaa kohtaan 2.

Viestien hakeminen

1. Käyttäjä painaa haku-linkkiä sivun yläreunassa.
2. Järjestelmä näyttää käyttäjälle hakusivun.
3. Käyttäjä kirjoittaa joko viestin avainsanoja, käyttäjänimen tai kummatkin. Hän myös mahdollisesti asettaa aikavälin jolta viestit noudetaan
4. Järjestelmä noutaa ja näyttää kriteerin täyttävät viestit.

Virhetilanne: Hakukriteerejä ei annettu

1. Kuten yllä
2. Kuten yllä
3. Käyttäjä yrittää hakea tietoa antamatta käyttäjänimeä tai hakutermejä.
4. Järjestelmä näyttää virheilmoituksen.

Uloskirjautuminen

1. Käyttäjä painaa uloskirjautumislinkkiä.
2. Järjestelmä kirjaa käyttäjän ulos ja palauttaa tämän takaisin aloitussivulle.

Ketjujen lukeminen

1. Käyttäjä navigoi itsensä jollekin aihealueelle
2. Käyttäjä painaa haluamansa ketjun nimeä.
3. Järjestelmä lataa ketjun viestit ja näyttää ne.

Ketjujen luonti

5. Käyttäjä navigoi itsensä jollekin aihealueelle.
6. Käyttäjä painaa ketjun luontilinkkiä.
7. Järjestelmä pyytää käyttäjältä ketjun nimeä sekä aloitusviestiä.
8. Käyttäjä syöttää ketjun nimen ja aloitusviestin.
9. Järjestelmä luo uuden ketjun ja ohjaa käyttäjän tähän ketjuun.

Virhetilanne: Aloitusviesti tai nimi puuttuu

5. Kuten yllä
6. Kuten yllä
7. Kuten yllä
8. Käyttäjä ei syötä joko nimeä tai aloitusviestiä.
9. Järjestelmä havaitsee aloitusviestin tai nimen puuttumisen ja antaa virheilmoituksen asiasta.
10. Prosessi palaa kohtaan 3.

Viestin kirjoittaminen

1. Käyttäjä navigoi itsensä ketjuun johon hän haluaa vastata
2. Käyttäjä vastaa viestiin joko
 - 2.1. lainaten jonkun toisen kirjoittamaa viestiä painaen tämän viestin yhteydessä olevaa lainausviestiä
 - 2.2. painaen sivun alareunassa olevaa uusi viesti-linkkiä
3. Järjestelmä pyytää käyttäjältä viestiä.
4. Käyttäjä kirjoittaa viestin.
5. Järjestelmä lisää ketjun perälle uuden viestin.
6. Käyttäjä palaa takaisin viestiketjuun.

Viestin muokkaaminen

1. Käyttäjä painaa viestin yhteydessä olevaa muokkausnappulaa.
2. Järjestelmä näyttää olemassa olevan viestin käyttäjälle.
3. Käyttäjä tekee haluamansa muutokset.
4. Järjestelmä tallentaa muutokset viestiin.

Viestin poistaminen

1. Käyttäjä painaa viestin yhteydessä olevaa poistonappulaa.
2. Järjestelmä varmistaa että viesti todella halutaan poistaa.
3. Käyttäjä vastaa myöntävästi
4. Viesti merkitään tietokannassa poistetuksi.
5. Sivu ladataan uudelleen, poistetun viestin kohdalla näytetään ilmoitus poistosta.

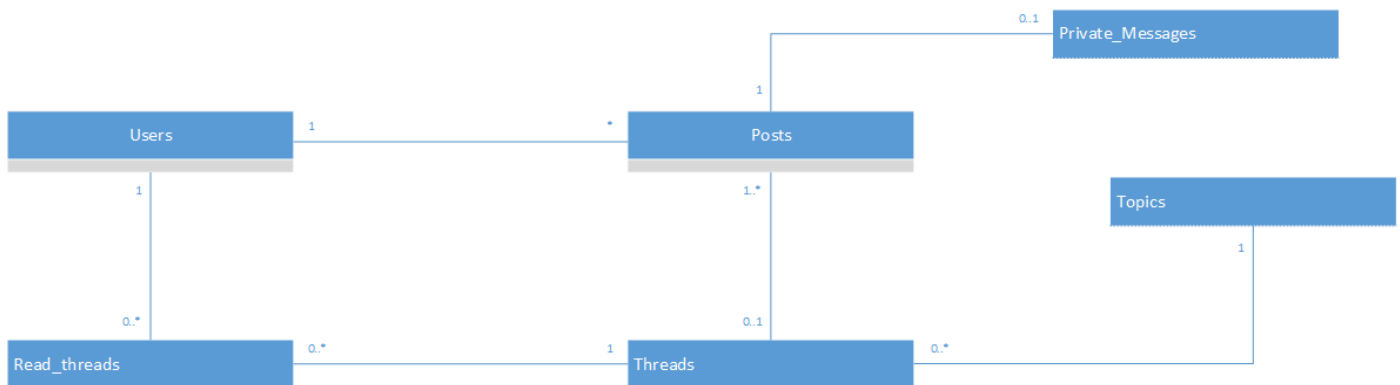
Ketjun poistaminen

1. Käyttäjä painaa ketjun yhteydessä olevaa poistonappulaa.
2. Järjestelmä varmistaa että ketju todella halutaan poistaa.
3. Käyttäjä vastaa myöntävästi.
4. Järjestelmä poistaa ketjun ja ketjuun liittyvät viestit.

Käyttäjän oikeuksien muuttaminen

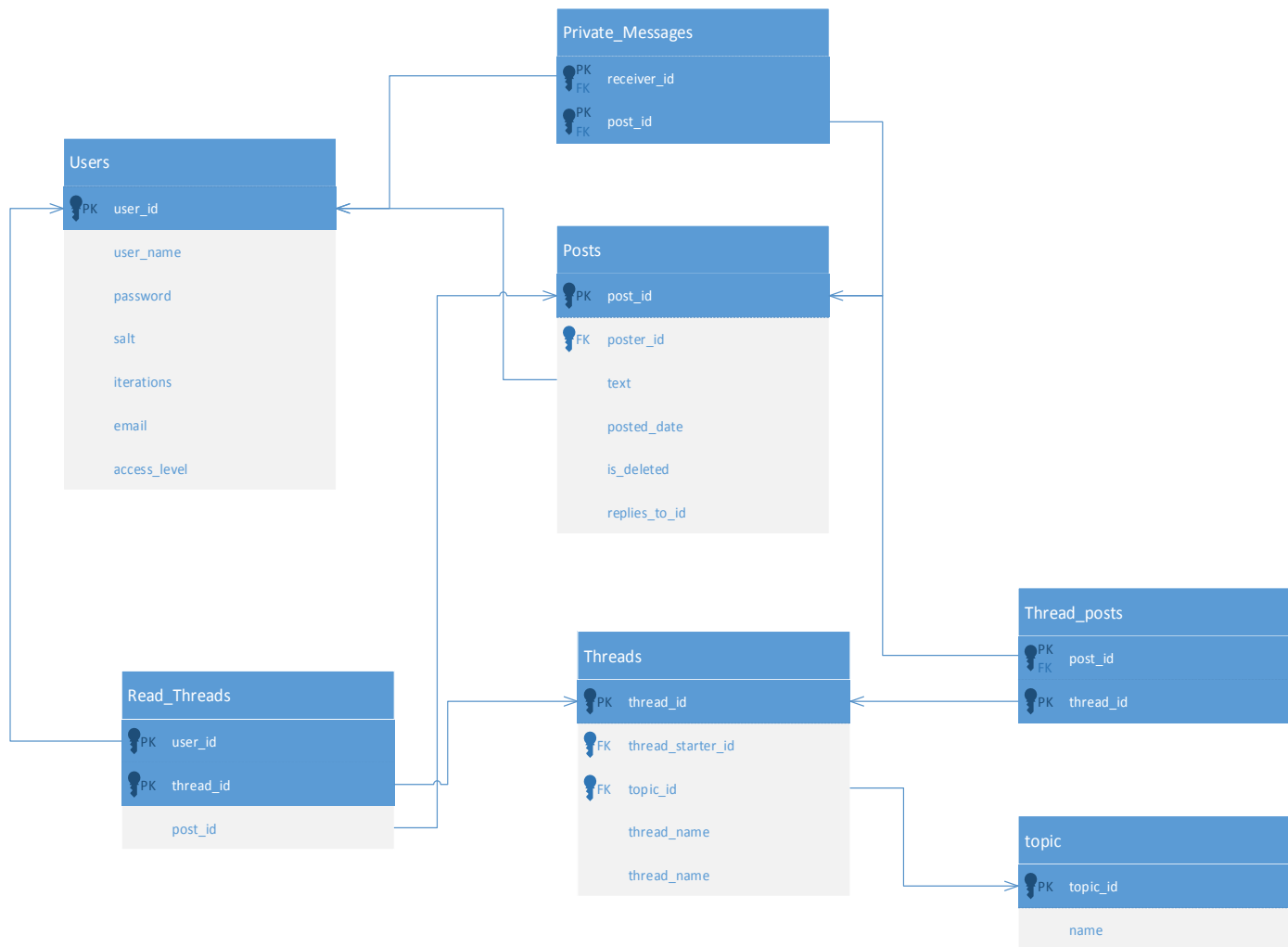
1. Käyttäjä avaa käyttäjähallinnan.
2. Käyttäjä etsii listasta haluamansa käyttäjän.
3. Käyttäjä valitsee alasvetovalikosta sopivan roolin ja valitsee sen.
4. Järjestelmä muuttaa roolin ja ilmoittaa asiasta.

Järjestelmän tietosisältö



Käyttäjällä voi olla useita viestejä, mutta yhteen viestiin liittyy aina yksi käyttäjä. Viestin on oltava joko yksityisviesti tai jollekin ketjulle kuuluva viesti, muttei yhtä aikaa kumpikin. Yksityisviestiin liittyy aina yksi viesti. Ketjulla voi olla useita viestejä; kuitenkin ainakin yksi (aloitusviesti). Ketju kuuluu johonkin alifoorumiin\aiheeseen mutta aina vain yhteen. Aiheella\alifoorumilla voi olla useita ketjuja. Lisäksi ylläpidetään listaa käyttäjän lukemista ketjuista. Käyttäjällä voi olla useita luettuja ketjuja, ja ketjuilla voi olla useita lukijoita.

Relaatiotietokantakaavio



User-taulu

Attribuutti	Arvojoukko	Kuvaus
user_id	serial (integer)	Käyttäjän tunnusluku
user_name	varchar(40)	Käyttäjän nimi, max. 40 merkkiä
email	varchar(40)	Käyttäjän sähköposti, max 40. merkkiä
user_password	char(64)	Käyttäjän salasanan tiiviste, sha-2
user_salt	char(64)	Salasanan suola; tiiviste sha-2 sessio-id:stä ja kellonajasta
iterations	integer	monta kertaa tiivistefunktio(salasana + suola) on laskettava
access_level	integer	Käyttäjän tyyppi; normaalikäyttäjä, admin jne

Posts-taulu

Attribuutti	Arvojoukko	Kuvaus
post_id	serial (integer)	Viestin tunnusluku
poster_id	integer	Kirjoittaneen käyttäjän id.
text	varchar(4096)	Viestin tekstikenttä; max 4 kilotavua.
posted_date	datetime	Päivämäärä ja kellonaika jolloin viesti on kirjoitettu. Muotoa yyyy-mm-dd hh:mm:ss
is_deleted	boolean	Onko viesti poistettu. Jos viesti on poistettu, ketjussa näkyy tällöin vain ilmoitus viestin poistosta
replies_to	integer	Jos vastaa johonkin viestiin niin sisältää viestin id:n. Muutoin null.

Private_messages-taulu

Attribuutti	Arvojoukko	Kuvaus
post_id	integer	Yksityisviestiin liittyvä viesti.
receiver_id	integer	Yksityisviestin vastaanottaja.

Topic-taulu

Attribuutti	Arvojoukko	Kuvaus
topic_id	serial (integer)	Aiheen id.
name	varchar(40)	Aiheen Nimi

Thread-taulu

Attribuutti	Arvojoukko	Kuvaus
thread_id	serial (integer)	Ketjun id
starter_id	integer	Käyttäjän id joka aloitti ketjun
topic_id	integer	Aiheen id jolle ketju kuuluu
thread_name	varchar(128)	Ketjun nimi

Thread_posts-taulu

Attribuutti	Arvojoukko	Kuvaus
thread_id	integer	Ketjun id jolle viesti kuuluu
post_id	integer	Viestin id

Read_threads-taulu

Attribuutti	Arvojoukko	Kuvaus
thread_id	integer	Ketjun id joka on luettu
user_id	integer	Lukijan id
post_id	integer	Viimeisen luetun viestin id

Järjestelmän yleisrakenne

Järjestelmä on toteutettu käyttäen mvc-mallia php-kielellä. Lisäksi järjestelmässä käytetään hieman javascriptiä erilaisten viestien näyttämiseksi sekä tiettyjen operaatioiden toteuttamisessa asynkronisesti.

Kontrollerit sijaitsevat projektin juurihakemistossa. Näkymät ovat views-kansiossa, kun taas mallit löytyvät libs/models-kansiosta. Lisäksi libs-kansiossa on yleisiä php-funktioita. Käytetyt css-määritteet ovat css-kansiossa ja javascript-tiedostot ovat js-kansiossa. Tietokannan alasajamiseen ja pystyttämiseen tarvittavat skriptit löytyvät sql-kansiosta.

Näkymien nimeämisessä on käytetty camelcase-tyyliä kun taas kontrollerien nimissä eri sanat on eritelty alaviivoilla. Mallien nimet vastaavat tietokantataulujen nimiä.

Järjestelmän komponentit

Näkymä\kontrolleriparit

base.php

Sivu joka ladataan aina ensiksi kun jokin näkymä näytetään. Näyttää riippuen siitä onko käyttäjä kirjautunut sisään log in, log out, register, control panel, search ja main page –linkit. Lisäksi Css-määritetiedostot ja javascript –tiedostot ladataan tällä sivulla.

index.php

index.php on sovelluksen etusivu. Sivulla listataan aihealueet ja niiden sisältämien ketjujen määrän. Aihealueen nimi toimii linkkinä josta avautuu sivu joka näyttää aihealueeseen liittyvät ketjut. Jos käyttäjä on kirjautunut

sisään ja hän on admin tai moderaattori, näytetään hänelle napit joiden avulla käyttäjä voi luoda, poistaa ja uudelleennimetä alueita.

register.php

Sivu jolla käyttäjä voi luoda uuden käyttäjätunnuksen. Sivulla on kentät käyttäjänimelle, sähköpostille, ja kahdesti salasanalle sekä nappi joka aloittaa rekisteröitymisen. Virhetilanteessa käyttäjälle näytetään virheilmoitus.

login.php

Sivu jolla näytetään käyttäjälle kirjautumisnäkyvä. Sivulla on kenttä käyttäjänimelle ja salasanalle sekä nappi joka aloittaa sisäänkirjautumisen. Mikäli kirjautuminen epäonnistuu, näytetään käyttäjälle sopiva virheilmoitus.

search.php

Sivu jolla käyttäjä voi hakea viestejä tietyin hakukriteerein. Käyttäjä voi antaa käyttäjänimen ja/tai hakutermejä joiden perusteella viestejä haetaan. Lisäksi käyttäjä voi rajata aikaväliä jolta viestejä haetaan. Haun jälkeen käyttäjä ohjataan näkymään jossa viestit listataan. Viestiä klikkaamalla käyttäjä viedään ketjuun jossa viesti on

register.php

Sivu jolla käyttäjä voi luoda uuden käyttäjätunnuksen. Sivulla on kentät käyttäjänimelle, sähköpostille, ja kahdesti salasanalle sekä nappi joka aloittaa rekisteröitymisen. Virhetilanteessa käyttäjälle näytetään virheilmoitus.

control_panel.php

Sivu jolla listataan linkit käyttäjän yksityisviesteihin sekä listaan rekisteröityneistä käyttäjistä.

user_management.php

Sivu jolla käyttäjä näkee listan rekisteröityneistä käyttäjistä. Jos käyttäjä on moderaattori, hän näkee lisäksi alasvetovalikon jossa hän voi asettaa käyttäjän tilan joko normaalikäyttäjäksi tai bannatuksi jos kyseinen käyttäjä ei ole moderaattori tai admin. Jos käyttäjä on admin, hän voi lisäksi muuttaa muiden moderaattoreiden ja adminien oikeuksia.

private_message_list.php

Sivu jolla näytetään käyttäjän yksityisviestit. Tämä sivu käyttää threadView.php-näkymää eli samaa näkymää kuin thread.php. Käyttäjälle näytetään myös poistonappula viestin yhteydessä.

threads.php

Sivu jolla näytetään valitun aihealueen ketjut. Jokaisen ketjun kohdalla näytetään ketjun nimi, kuka loi ketjun, monta viestiä on ketjussa, sekä milloin viimeisin viesti ketjuun on lisätty. Ketjun nimi toimii linkkinä sivulle jossa näkyy ketjun viestit. Lisäksi jos käyttäjä on kirjautunut sisään ja häntä ei ole bannittu, näytetään onko ketjussa lukemattomia viestejä ja tarjotaan linkki joka vie viimeiseen luettuun viestiin sekä nappi jota painamalla käyttäjä ohjataan sivulle jolla voidaan luoda uusi ketju. Jos käyttäjä on lisäksi admin tai moderaattori, näytetään hänelle napit joiden avulla käyttäjä voi luoda, poistaa tai uudelleennimetä ketjuja.

thread.php

Sivu jolla näytetään valitun ketjun viestit. Näkyvillä on aina kaksi linkkiä joista ensimmäisellä näkee ketjun lukijat ja toisella pääsee takaisin ketjusivulle. Lisäksi sivulla on näkyvissä jokainen ketjun viesti. Jokaisen viestin kohdalla näytetään viestin lähettäjän nimi, lähetysaika ja viestin sisältö. Viestin yhteydessä näytetään myös mahdollisesti lainattu viesti. Jos käyttäjä on kirjautunut sisään ja häntä ei ole bannittu, hän voi klikata viestinlähettäjän nimeä jolloin hänet ohjataan yksityisviestin lähetysivulle. Sisäänkirjautunut käyttäjä näkee aina myös kaksi nappia joista ensimmäistä painamalla hänet ohjataan sivulle jossa hän voi kirjoittaa uuden viestin, toista painamalla hänet ohjataan sivulle jossa hän voi kirjoittaa vastauksen valitulle viestille. Käyttäjä näkee myös omien viestiensä kohdalla napin jolla hänet ohjataan sivulle jossa viestiä voi muokata, mutta vain jos viestiä ei ole merkitty poistetuksi. Sivun alareunassa on linkit edelliselle\seuraavalle sivulle jos se on tarpeellista.

Jos käyttäjä on lisäksi moderaattori tai admin, hän näkee muokkausnapin jokaisen viestin kohdalla. Lisäksi käyttäjä näkee tällöin viestin poistonapin jokaisen viestin kohdalla. Nappia painamalla viesti merkitään poistetuksi jolloin sen sisältö korvataan ilmoituksella poistosta. Poistettua viestiä ei voi enää poistaa tai muokata.

reply.php

Sivulla näytetään tekstilaatikko ja submit-nappula. Tekstilaatikkoon kirjoitettu teksti lisätään uutena viestinä ketjuun josta sivulle tultiin alun perin. Tekstiä ei ole pakko kirjoittaa vaan tyhjänkin viestin voi jättää.

edit.php

Sivu on vastaavanlainen kuin reply.php. Erona viestin teksti on lisätty tekstikenttään valmiiksi. Lisäksi linkit johtavat eri sivuille\kontrollereille. Näkymä on erillinen vastausnäkymästä koska sivu on erittäin yksinkertainen joten koodin toisto ei ole valtavaa ja koin että eroavaisuuksien määrittely php:n avulla johti if/else-spagettiin jota on vaikea hallinnoida. Jos sivut muuttuvat monimutkaisemmiksi kuin nyt ja koodin toistaminen aiheuttaa ylläpito-ongelmia, kannattaa näkymien yhdistämistä harkita.

quote.php

Samankaltainen sivu kuin reply.php ja edit.php. Eroina ovat linkkien osoitteet ja se, että käyttäjälle näytetään lainattava viesti. Tämä on erillinen sivu samasta syystä kuin yllä on selitetty, ja sama huomio pätee myös tälle sivulle.

private_message.php

Sivu yksityisviestien lähettämistä varten. Tämän näkymän kanssa on sama logiikka käytössä kuin quote\edit\reply-näkymien.

readers.php

Sivu jolla listataan ketjun lukeneet henkilöt. Yksinkertainen listanäkymä. Jos käyttäjä on kirjautunut sisään, nimien (paitsi oman) klikkaaminen vie yksityisviestin lähettämisenäkymään.

new_thread.php

Sivu jolla käyttäjä luo uuden ketjun. Käyttäjälle näytetään kentät ketjun nimeä ja aloitusviestiä varten sekä nappi joka luo uuden ketjun. Sekä ketjun nimi että aloitusviesti ovat pakollisia.

Kontrollerit ilman näkymää

Nämä kontrollerit ladataan asynkronisesti javascriptin avulla eikä näihin liity näkymiä.

`delete_post.php`

Kontrolleri joka poistaa annetun viestin jos käyttäjällä on oikeudet viestin poistoon.

`delete_thread.php`

Kontrolleri joka poistaa annetun ketjun ja ketjun viestit jos käyttäjällä on oikeudet ketjun poistoon.

`logout.php`

Kontrolleri joka kirjaa käyttäjän ulos.

`rename_thread.php`

Kontrolleri joka uudelleennimeää ketjun jos käyttäjällä on oikeudet uudelleennimeämiseen.

`role_change.php`

Kontrolleri joka vastaa käyttäjien roolin vaihtamisesta jos kutsuvalla käyttäjällä on oikeudet roolien muokkaamiseen.

`topic_management.php`

Kontrolleri joka vastaa aiheiden luonnista\poistosta\uudelleennimeämisestä jos käyttäjällä on oikeudet näihin operaatioihin.

Mallit

post.php

Malli joka vastaa yksittäisistä viesteistä. Sisältää viestin kentät ja CRUD-operaatioita viesteille.

thread.php

Kuten yllä, mutta ketjuille.

topic.php

Kuten yllä, mutta aihealueille.

user.php

Kuten yllä, mutta käyttäjälle. Lisäksi sisältää salasanan hashaukseen liittyvät funktiot.

Muita tiedostoja

database.php

Luokka joka vastaa tietokannan käytöstä ja yhteyden luonnista ja ylläpidosta. Kaikki tietokantaoperaatiot kulkevat tämän luokan kautta.

utility.php

Tiedosto joka sisältää yleiskäyttöisiä funktioita. Tämä tiedosto myös käynnistää käyttäjän session.

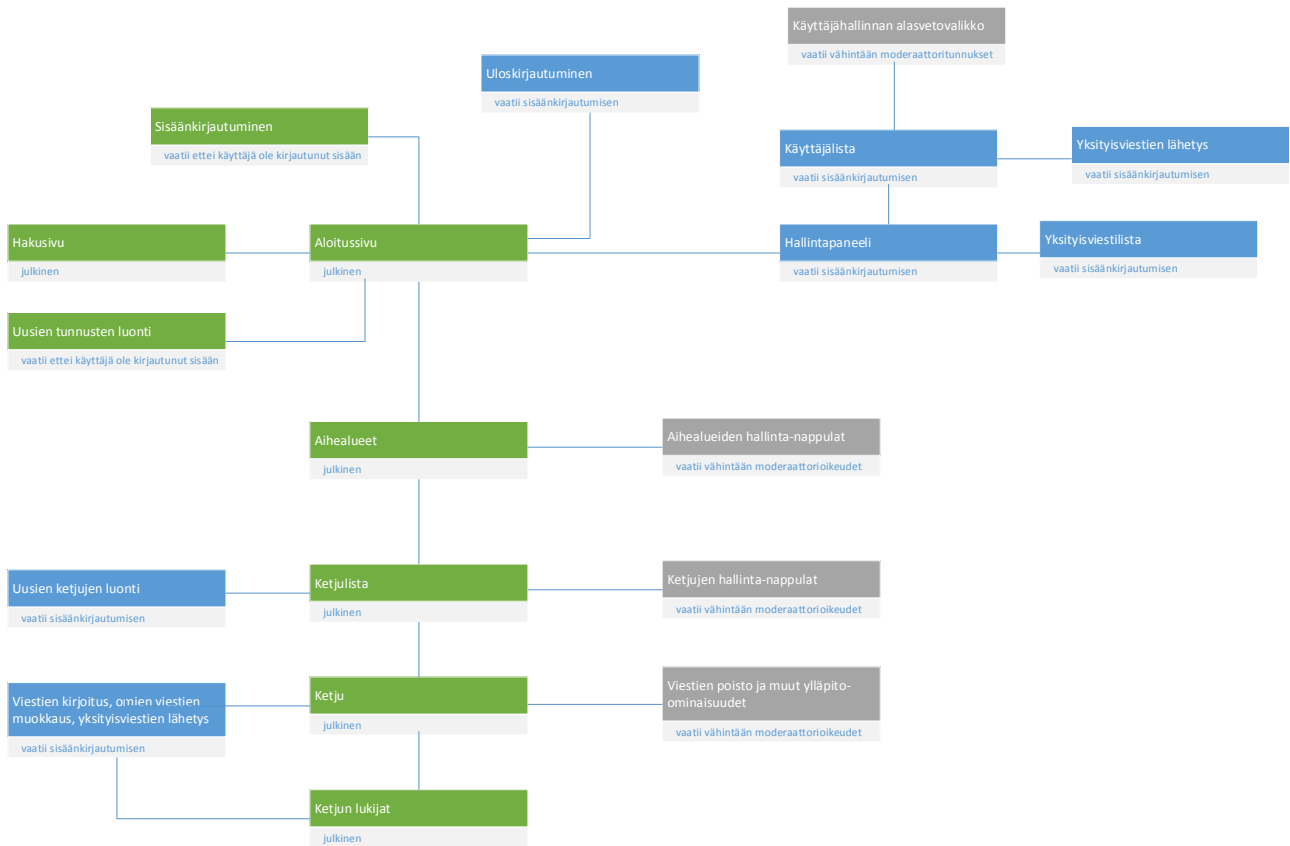
utility.js

Yleiskäyttöisiä javascript-funktioita.

create_table.sql, drop_table.sql, insert_data.sql, setup

Testitietokannan alasajoon ja pystyttämiseen tarvittavat tiedostot.

Käyttöliittymä



Käyttöliittymäkaaviossa käytetty seuraavaa värikoodausta:

Vihreä – julkinen

Sininen – vaatii sisäänkirjautumisen

Harmaa – vaatii moderaattori\adminioikeudet

Asennustiedot

Ohjelma asennetaan kopioimalla sen tiedostot johonkin sopivaan hakemistoon, esim. htdocs-kansion alle.

Ohjelmalla ei ole ulkoisia konfiguraatioita mutta se käyttää ulkoisia javascript-kirjastoja jotka on linkattu Googlen palvelimelta.

Käynnistys- / käyttöohje

Aloitussivu on osoitteessa <http://ekaaria.users.cs.helsinki.fi/board/index.php>. Toimintojen pitäisi itsestään selviää jos on joskus aikaisemmin käyttänyt foorumeita.

Testaus, tunnetut bugit ja puutteet & jatkokehitysideat

- Salasanaa ei voi vaihtaa
- Voi lainata vain yhtä viestiä
- Sähköpostia ei tarkisteta mitenkään (tai ylipäätään käytetä mihinkään)
- Ulkoasua voisi parantaa.
- Bugi: Yksityisviestin lähettäminen ketjun lukijalistan kautta palauttaa väärälle sivulle
- Bugi: Osa järjestelmäviesteistä ei näy (editointi\quote) – miksi?
- Bugi\puute: Samalla käyttäjätunnuksella voi kirjautua monelta koneelta yhtä aikaa.

KESKEN, poista
ranskalaiset viivat ja
korvaa tekstillä

Omat kokemukset

KESKEN

Liite 1 – aihealueen määrittely

Katso <http://advancedkittenry.github.io/suunnittelu/aiheet/Keskustelufoorumi.html>

Liite 2 – Create table-lauseet

```
CREATE TABLE users (  
  user_id serial PRIMARY KEY,  
  user_name varchar(40) NOT NULL UNIQUE,  
  email varchar(40) NOT NULL,  
  user_password char(64) NOT NULL,  
  user_salt char(64) NOT NULL,  
  iterations integer NOT NULL,  
  access_level integer NOT NULL  
);
```

```
CREATE TABLE posts (  
  post_id serial PRIMARY KEY,  
  poster_id integer REFERENCES users (user_id) ON DELETE CASCADE ON UPDATE CASCADE,  
  text varchar(4096),  
  posted_date timestamp NOT NULL,  
  is_deleted boolean NOT NULL,  
  replies_to integer REFERENCES posts (post_id) ON DELETE SET NULL ON UPDATE CASCADE
```

```
);
```

```
CREATE TABLE private_messages (  
    post_id integer REFERENCES posts (post_id) ON DELETE CASCADE ON UPDATE CASCADE,  
    receiver_id integer REFERENCES users (user_id) ON DELETE CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY (post_id, receiver_id)  
);
```

```
CREATE TABLE topics (  
    topic_id serial PRIMARY KEY,  
    name varchar(40) NOT NULL  
);
```

```
CREATE TABLE threads (  
    thread_id serial PRIMARY KEY,  
    thread_name varchar(128) NOT NULL,  
    starter_id integer REFERENCES users (user_id) ON DELETE CASCADE ON UPDATE CASCADE,  
    topic_id integer REFERENCES topics (topic_id) ON DELETE CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE thread_posts (  
    thread_id integer REFERENCES threads (thread_id) ON DELETE CASCADE ON UPDATE CASCADE,  
    post_id integer REFERENCES posts (post_id) ON DELETE CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY (thread_id, post_id)  
);
```

```
CREATE TABLE read_threads (  
    thread_id integer REFERENCES threads (thread_id) ON DELETE CASCADE ON UPDATE CASCADE,
```

```
user_id integer REFERENCES users (user_id) ON DELETE CASCADE ON UPDATE CASCADE,  
post_id integer REFERENCES posts (post_id) ON DELETE CASCADE ON UPDATE CASCADE,  
PRIMARY KEY (thread_id, user_id)  
);
```