

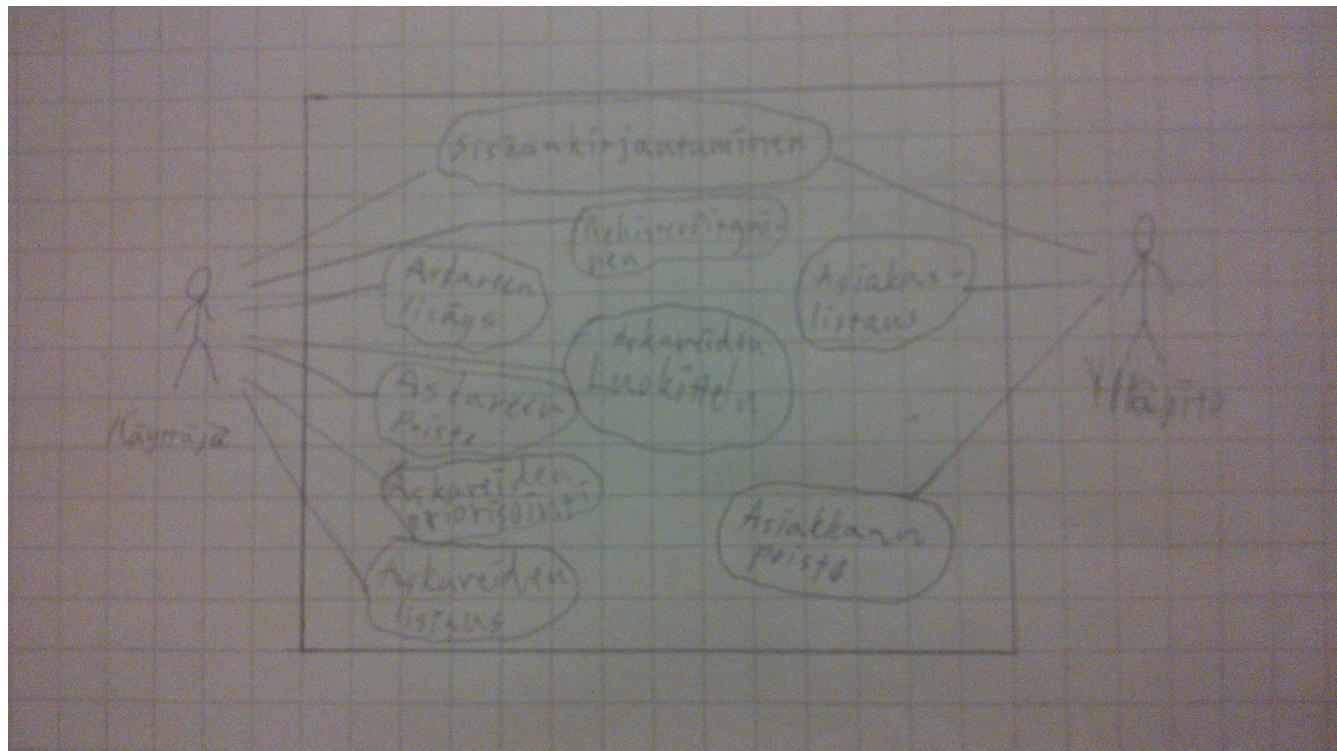
Tietokantasovellus – Muistilista

Johdanto

Tämän harjoitustyön tarkoituksena on ohjelmoida toimiva tietokantasovellus. Sovellus käsittelee eri käyttäjien muisti- tai tehtävä/askarelistoja (TO DO-lists). Sovellus sisältää mahdollisuuden sisäänkirjautumiseen ja uuden käyttäjätunnuksen luomiseen; uusien askareiden lisäys ja poisto; askareiden tärkeysjärjestyksen muuttaminen ja askareiden luokittelu. Askareilla tulisi olla myös mahdollisuus kuulua moneen luokkaan.

Sovellus toteutetaan php-ohjelmointikielellä ja postgreSQL-kyselykielellä. Sovelluksen toteutusympäristönä on Helsingin yliopiston users-palvelin.

Käyttötapauskaavio



Käyttötapaukset

Asiakkaan käyttötapaukset:

Asiakas rekisteröityy järjestelmään: Hän luo käyttäjätunnuksen ja saa oikeudet käyttää muistilistan palveluita.

Asiakas luo uuden askareen muistilistaansa: Askareelle kirjoitetaan kuvaus, sekä voidaan asettaa prioriteetti, suoritusajankohta sekä erilaisia luokkia.

Asiakas muokkaa askareitaan: Askareen prioriteettia, ajankohtaa tai kuvasta voi muokata. Askareille voidaan myös lisätä luokkia.

Asiakas noutaa listan askareistaan: Askareiden listan voi järjestää prioriteetin, ajankohdan tai luokkien mukaan.

Asiakas poistaa askareen: Tarvittaessa askareen voi myös poistaa muistilistasta. Hävitetyä tietoa ei enää saa takaisin.

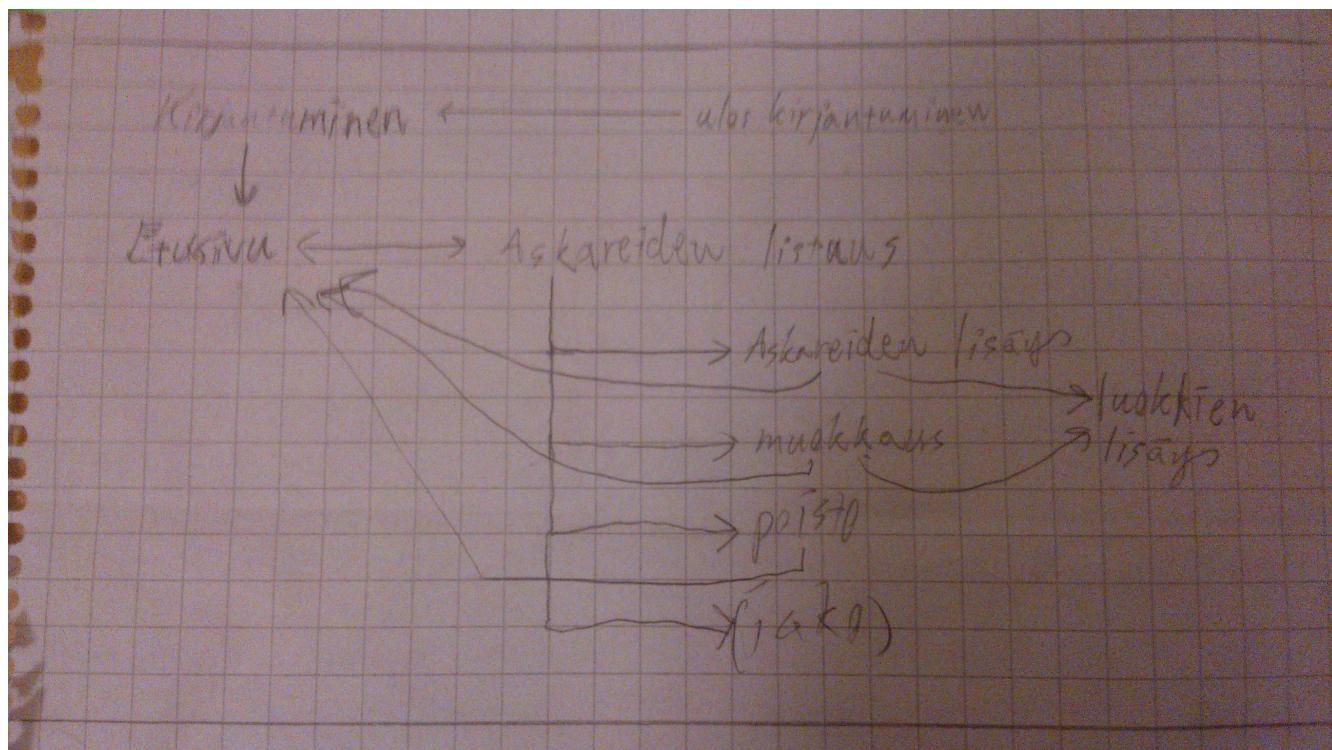
Asiakas poistaa käyttäjätilinsä: Halutessaan käyttäjätilin voi poistaa, jolloin kaikki asiakkaan tiedot ja askareet hävitetään.

Ylläpitäjän käyttötapaukset:

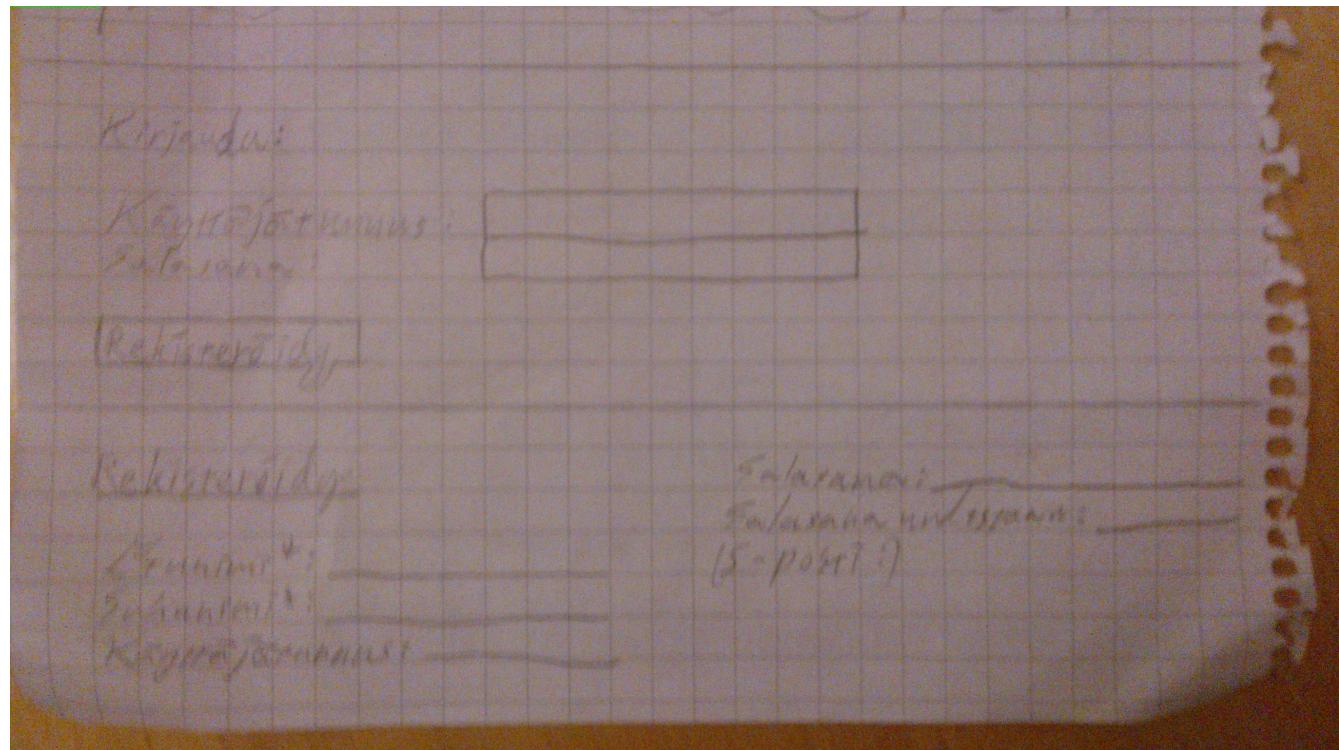
Ylläpitäjä poistaa asiakkaan. Jos asiakas ei jostain syystä voi itse poistaa käyttäjätiliään, ylläpitäjä voi sen tehdä.

Ylläpitäjä listaa asiakkaat: Ylläpitäjä voi pitää kirjaan asiakkaiden käyttötileistä ja sisäänsäkirjautumisajankohdista.

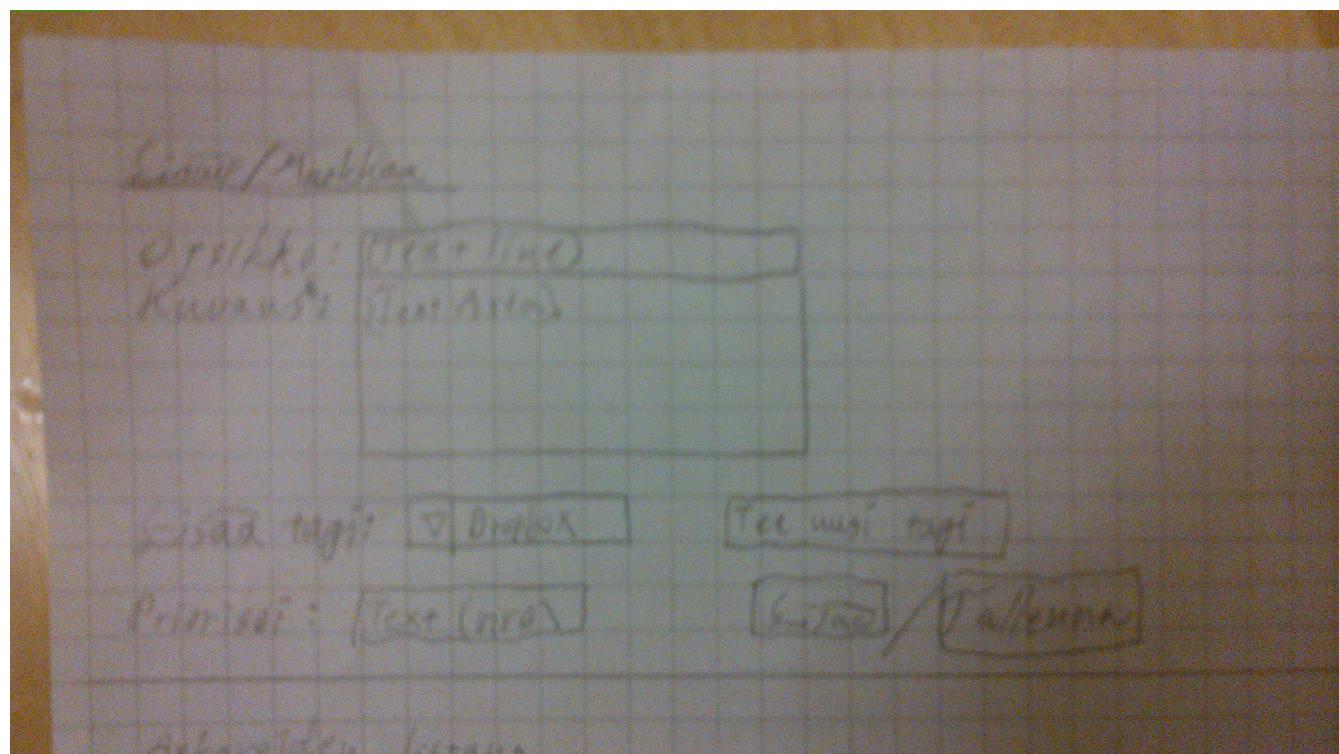
Käyttöliittymäkaaviot



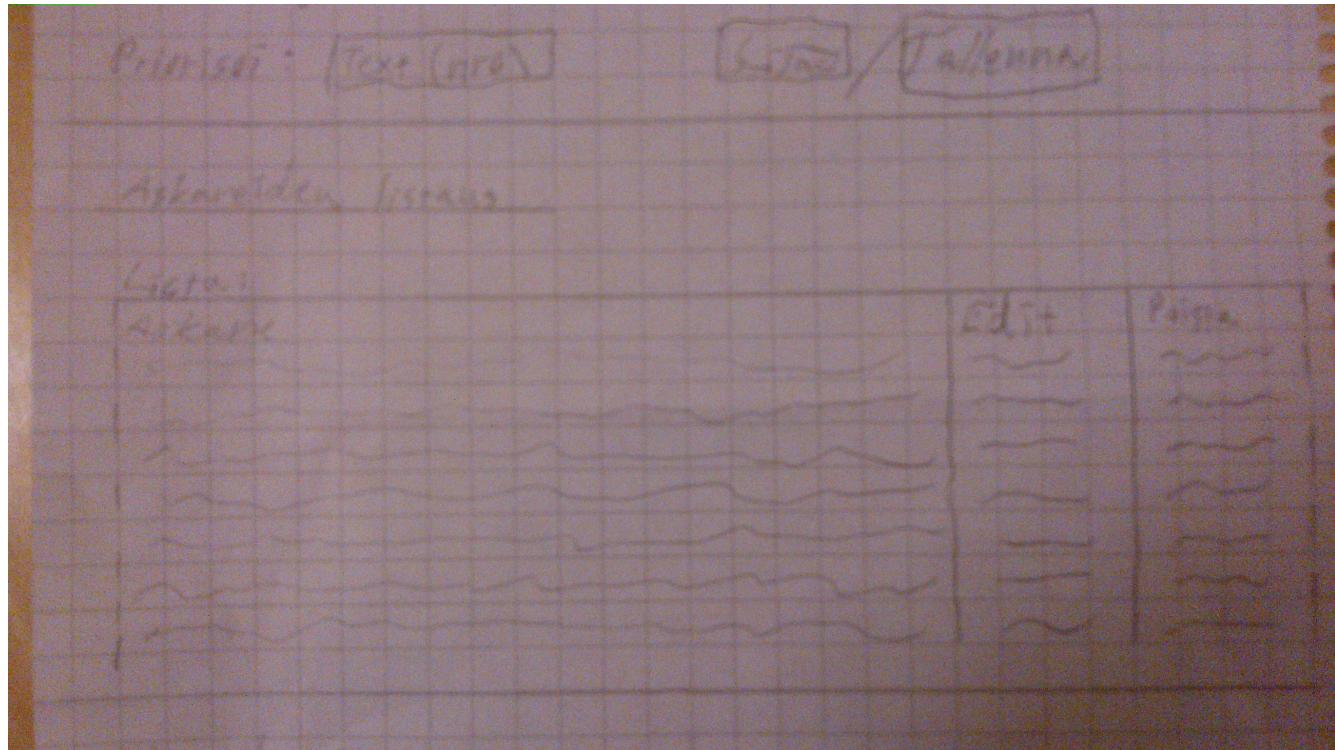
Kirjautuminen ja rekisteröinti:



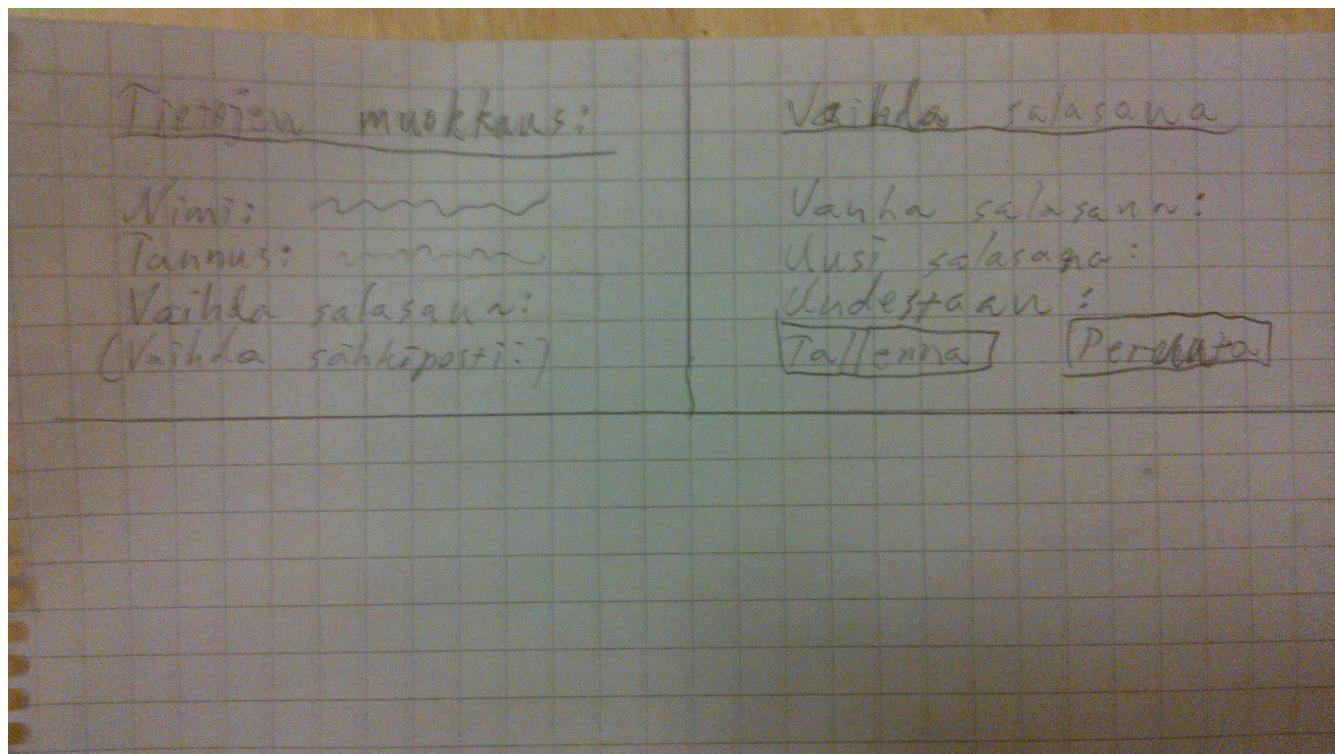
Lisäys ja muokkaus:



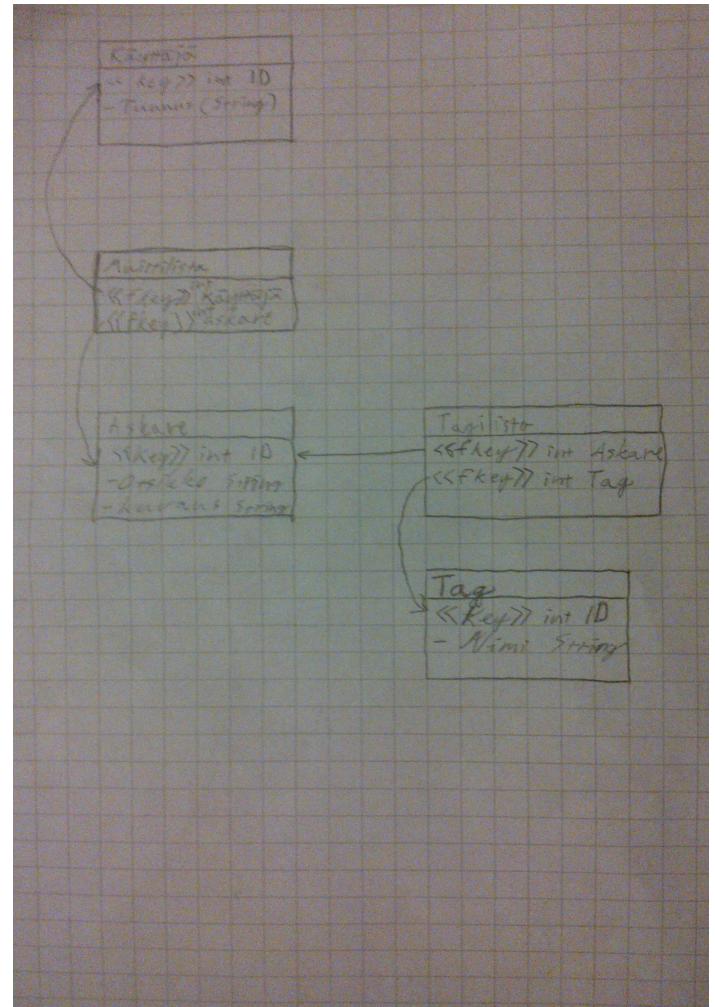
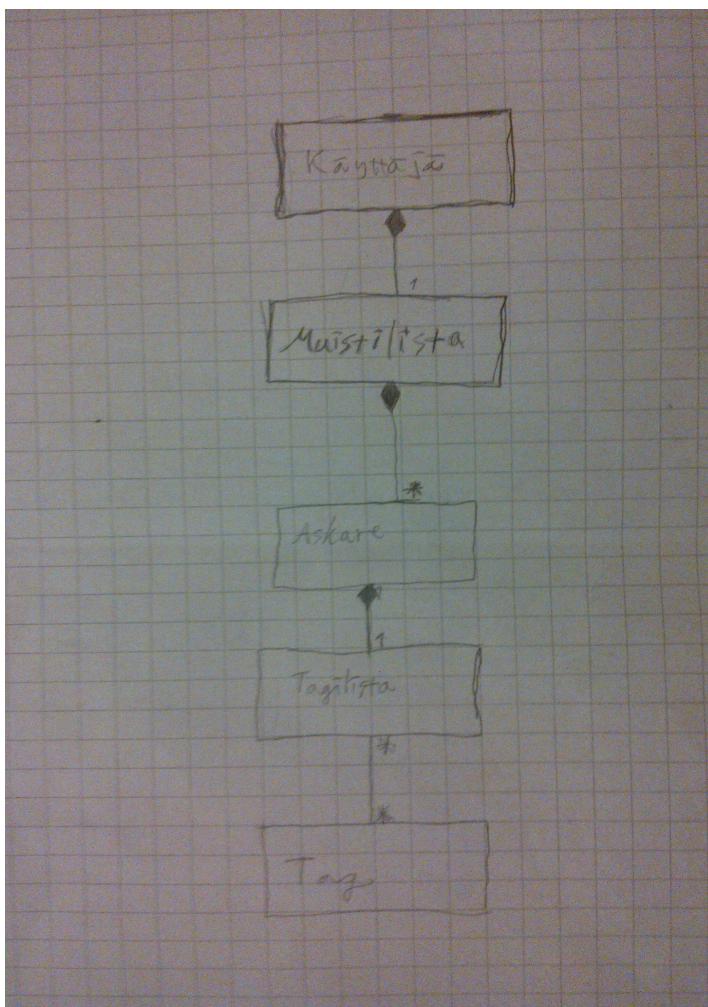
Listaus:



Tietojen muokkaus:



Luokka- ja relaatiotietokantakaaviot:



Tietokantataulut ja niiden attribuuttien kuvaukset:

Käyttäjä:

Attribuutti	Arvojoukko	Kuvailu
etunimi	Merkkijono (max 30)	Käyttäjän oikea etunimi.
sukunimi	Merkkijono (max 30)	Käyttäjän oikea sukunimi.
tunnus	Merkkijono (max 30)	Käyttäjän valitsema tunnus.
salasana	Merkkijono (max 30)	Käyttäjän valitsema salasana.
yllapitaja	Boolean	Tosi jos käyttäjä on ylläpitäjä.

Askare.

Attribuutti	Arvojoukko	Kuvailu
Otsikko	Merkkijono (max50)	Askareen nimi. Pakollinen
Kuvaus	Merkkijono (max150)	Askareen kuvaus. Vapaaehtoinen

Päivämäärä	Päivämäärä ja kellonaika	Askareen tapahtuma-aika. Vapaaehtoinen.
------------	--------------------------	--

Tagi.

Attribuutti	Arvojoukko	Kuvailu
Nimi	Merkkijono (max 20)	Tagin nimi. Käyttäjän määrittelemä. Esim. "luento" tai "bileet".

Näiden lisäksi konstruktoidaan myös viitetaulukot.

Muistilista.

Attribuutti	Arvojoukko	Kuvailu
KayttajaID	Integer(ref)	Viite käyttäjään.
AskareID	Integer(ref)	Viite askareeseen.

Tagilista

Attribuutti	Arvojoukko	Kuvailu
TagiID	Integer(ref)	Viite tagiin.
AskareID	Integer(ref)	Viite askareeseen.

Asennustiedot

Sovellus pyörii php-palvelimella sekä käyttää postgresql-tietokantaa.

Käyttöohje

Sovelluksen URL:

<http://krha.users.cs.helsinki.fi/muistilista/kirjautuminen.php>

Sivulle voi kirjautua tunnuksella "Tyhjis" ja salasanalla "testi". Voit myös luoda oman käyttäjätunnuksen. Kirjautumisen jälkeen käyttäjä ohjataan etusivulle, jossa listataan käyttäjän lisäämät askareet. Etusivu sisältää linkit uloskirjautumiseen ja askareen lisäämiseen. Jokaisen listatun askareen alapuolella on "Muokkaa tai poista" -nappi, joka ohjaa käyttäjän askareen muokkaamissivulle.

Järjestemän yleisrakenne

Sovelluksen toiminnot on järjestetty projektin juuren alapuolella sijaitseviin alihakemistoihin. Jokaiselle tietokantataululle on toteutettu omat php-tiedostot, jotka sisältävät luokat funktoineen. Erona on luokka "Tagi", joka ei sisällä kuin kaksi julkista muuttujaa, eikä yhtään funktiota. Tageja manipuloivat funktiot sisältyvät täysin "Askare" -luokkaan.

Kaikki sovelluksen sisältämät kontrollerit ovat projektin juureessa kansiossa htdocs/muistilista.

Hakemistorakenne users-palvelimella: muistilista/ (sisältää kontrollerit)
views/ (sisältää näkymätiedostot html)
libs/ (common.php -tiedoston)
models/ (mallitiedostot eli luokat)

Järjestelmän komponentit

kirjautuminen.php

Ensimmäinen sivu. Sisältää toiminnot sisäänkirjautumiseen ja ohjauksen rekisteröintiin. Käyttää näkymää "login.php".

rekisteroi.php

Sisältää toiminnot uuden käyttäjän lisäämiseen tietokantaan.

etusivu.php

Sovelluksen pääsivu. Listaa käyttäjän askareet, sekä sisältää ohjaukset askareen lisäämiseen, jakamiseen muille käyttäjille, poistamiseen ja muokkaamiseen.

lisays.php

Sisältää toiminnot uuden askareen lisäämiseen tietokantaan.

poista.php

Askareen poistaminen. Ei käytä yhtään näkymätiedostoja. Ohjaa suoraan takaisin etusivulle.

jaa.php

Askareen jakaminen muille käyttäjille. Esittää käyttäjälle kaikki ne käyttäjät, joille askaretta ei ole vielä jaettu. Käyttää näkymätiedoston share.php:tä. Käyttäjä voi jakaa kyseisen askareen niin monelle kuin haluaa tai palata takaisin etusivulle.

muokkaus.php

Toiminnot askareen tietojen muokkaamiseen. Käyttäjä voi muokata askareen otsikkoa, kuvausta tai ajankohtaa. Askareelle voi myös lisätä tageja.

logout.php

Uloskirjautuminen. Käyttäjä ohjataan takaisin sisäänkirjautumiseen.

Views/template.php

Jokaisen sivun käyttämä pohjanäkymätiedosto, joka sisältää <head>-tagin sisältämät asiat. Sivun aloittava <html>-tagi löytyy täältä.

Views/lowertemplate.php

Pohjanäkymätiedoston alapuoli. Sivun lopettava </html>-tagi löytyy täältä.

Views/frontpage.php

Etuksen näkymätiedosto. Askareiden listaus.

Views/login.php

Sisäänkirjautumisen näkymätiedosto. Sisältää linkin rekisteröitymiseen.

Views/register.php

Rekisteröitymisen näkymätiedosto. Sisältää lomakkeen uuden käyttäjän tietojen keräämiseen.

Views/add.php

Askareen lisäämisen lomakkeet sisältävä näkymätiedosto. Käyttäjä voi määrittää otsikon, kuvauksen ja ajankohdan tai palata takaisin etusivulle lisäämättä mitään.

Views/edit.php

Askareen muokkaukseen käytetty näkymätiedosto. Nämä muuten sama kuin add.php:ssä, mutta käyttäjällä on mahdollisuus lisätä tageja askareelle.

Views/share.php

Näkymätiedosto askareen jakamiseen. Esittää käyttäjälistan.

Views/css/

Tämä kansio sisältää Twitter-bootstrap-kirjaston css-tiedostot. Näistä tiedostoista on käytössä vain bootstrap.css ja bootstrap-theme.css.

Libs/common.php

Sisältää yleisessä käytössä olevat funknot, kuten superglobaalien muuttujien alustaminen ja tuhoaminen sekä näkymätiedostojen näyttäminen. Jokainen kontrolleri käyttää tätä tiedostoa.

Libs/models/yhteys.php

Sisältää funktion tietokantayhteyden muodostamiseen. Jokainen mallitiedosto käyttää tätä tiedostoa.

Libs/models/askare.php

Sisältää "Askare"-luokan, joka sisältää funknot askareiden hakemiseen, lisäämiseen, muokkaamiseen ja poistoon. Sisältää myös "Tagi"-luokan vastaavat funknot.

Libs/models/tagi.php

"Tagi"-luokka. Sisältää ainoastaan kaksi julkista muuttuja.

Libs/kayttaja.php

"Käyttäjä"-luokan määrittely. Sisältää funknot sovelluksen käyttäjien hakemiseen ja lisäämiseen.

Testaus, tunnetut bugit ja puutteet sekä jatkokehitysideoat

Sovellusta on testattu suoraan käytännössä. Automaattisia yksikkötestejä ei tehty. Viimeisimmässä testeissä ei ohjelmaa käytettäessä ilmennyt vikoja. Aiempissa testauksissa on kuitenkin ilmennyt bugeja, joissa tietokantaan on kirjoitettu sama teksti moneen kertaan, vaikka käyttäjä ei ole tätä tehnyt. Tämä bugi saattaa toistua vielä, sillä vikaa ei löydetty.

Suurin vika sovelluksessa on hieman sekava käyttöliittymä. Ohjelma ei välitämättä kaikissa tilanteissa ole niin ohjeistava. Jatkokehitysideana voisi nimenomaan olla käyttöliittymän selkeyttäminen.

Joidenkin toimintojen lisääminen toisi myös lisää ulottuvuuksia sovellukseen, kuten askareiden kommentointi eri käyttäjien toimesta.

Omat kokemukset

Tietokantasovelluksen rakentamisessa oli helpointa itse tietokannan suunnittelu, sillä olen aikaisemmin tehnyt yhden suuremman ja monimutkaiseman tietokannan. Itse sovelluksen toteuttaminen oli paljon hankalampaa, sillä web-sovellusten ohjelmointia en ole koskaan aikaisemmin tehnyt ja jopa HTML-kieltä olen viimeksi kunnolla opettellut vuosituhannen vaihteen aikoihin. HTML 5 ei toisaalta kovin hankalalta loppujen lopuksi vaikuttanut. Suurimpana kynnyksenä toimi php-kieli, jonka opettelu aiheutti alussa muutaman harmaan hiukseen. Tyypittömistä ja tulkatuista ohjelmointikielistä minulla oli projektin alussa hyvin minimaalinen kokemus. Kun lopulta php:n salat alkoivat hieman paljastua, olin jo kirjoittanut sillä sen verran purukumikoodia, jonka korjaaminen olisi vienyt aikaa enemmän kuin sitä oli saatavilla. Tämän takia sovelluksen koodi sisältää mielestäni hieman liikaa rivejä. Projektin loppua kohden php alkoi kuitenkin tuntua kotoisalta ja jos joskus aion web-ohjelmointia harrastaa, php olisi kieli jonka valitsisin tällä hetkellä, sillä haluaisin oppia siitä enemmän.

Kaikin puolin pidin tätä harjoitustyötä erittäin hyödyllisenä. Uskon, että opin käytännössä paljon enemmän uusia asioita kuin muilla harjoitustyökursseilla.

Kristian Hansson