

Viivin Muistilista-sovellus

-dokumentaatio

Johdanto

Sovellukseni on eräänlainen muistilista, johon rekisteröitymisen ja kirjautumisen jälkeen voi tehdä omaa käyttäjää koskevan muistilistan. Listalle voi kirjoittaa askareita tai muita muistettavia asioita. Jokaista muistilistan kohtaa kohden voidaan valita sille tärkeysaste (1-5), jonka mukaan muistilista on järjestetty. Askareen lisäyksen jälkeen sen voi merkitä tehdyksi tai poistaa listasta lopullisesti. Askareita voi myös muokata. Jokaista askareta kohden voi lisätä yhden luokan ja jokaisessa luokassa voi luonnollisesti olla monta askareta.

Muistilistani tarkoitus on helpottaa asioiden muistamista; kun hoidettavat asiat tai askareet kirjoittaa johonkin, on helppo olla perillä siitä, mitä on tehty ja mitä pitää vielä tehdä. Parhaimmassa tapauksessa Muistilista-sovellukseni voi olla jokapäiväinen apuväline, joka kulkee aina mukana esim. puhelimen nettiyhteyden kautta. Muistilistaa voi käyttää kuka tahansa, joka kokee arkisten asioiden muistamisen olevan haastavaa.

Työ toteutetaan yliopiston users-palvelimella PHP-ohjelmointikieltä käyttäen. Sovelluksen muotoilussa ja muussa visualisoinnissa käytetään CSS:ää. Sovelluksessa käytetään Postgre-SQL -tietokantajärjestelmää.

Käyttötapaukset

Kirjautuminen

Sovellukseen voi rekisteröityä kuka tahansa ja kirjautua rekisteröitynyt käyttäjä.

Muistilistan selaaminen:

Sisäänkirjautumisen jälkeen käyttäjälle avautuu näkymä käyttäjän omasta Muistilistasta askareineen.

Askareen lisäys:

Selausnäkyvässä Lisää askare -nappulan painamisen jälkeen käyttäjä ohjataan lisäysnäkyvään, jossa käyttäjä voi lisätä muistilistaansa uuden askareen. Uutta askareta kohden voidaan valita askareen nimi, tärkeysaste, luokka ja suoritustilanne (kesken/valmis). Kun askare tallennetaan, se näkyy aloitusnäkyvän Muistilistassa.

Askareen muokkaaminen:

Jo luotuja askareita voi muokata muokkaus-näkyvässä, joka on käyttöasteeltaan samanlainen, kuin lisäysnäkyvä.

Askareen poistaminen:

Luotuja askareita voi poistaa muistilistasta pysyvästi jokaista luotua askareta koskevasta Poista-nappulasta askareen muokkaustilassa.

Uuden luokan lisääminen:

Jokainen askare voidaan liittää johonkin luokkaan. Näitä luokkia voidaan lisätä sovelluksen etusivulla olevan Luokat-linkin kautta, josta avautuu erillinen näkymä luokkien listaukseen ja niiden lisäämiseen.

Luokkien selaaminen:

Luokkia voidaan selata etusivun ylälaudassa olevan Luokat-linkin kautta. Tätä nappulaa painamalla saadaan auki sivu, jossa on luettelona kaikki jo luodut luokat. Myös jokaista yksittäistä luokkaa voi tarkastella painamalla luodun luokan nimestä aukeavaa linkkiä.

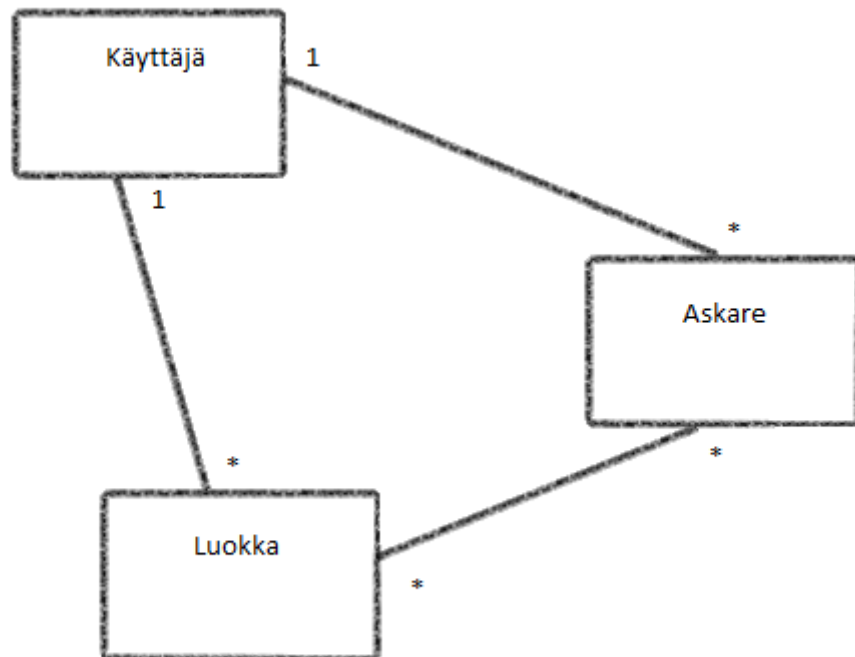
Luokan muokkaaminen:

Jo luotuja luokkia voidaan muokata jokaisen yksittäisen luokan selaustilassa olevasta napista.

Luokan poistaminen:

Kuten askareitakin, myös luokkia voidaan poistaa. Tämä tapahtuu yksittäisen luokan sivulla poista luokka -painikkeen kautta.

Järjestelmän tietosisältö



Käyttäjä:

- nimi: merkkijono max. 20 merkkiä
- salasana: merkkijono max 50 merkkiä

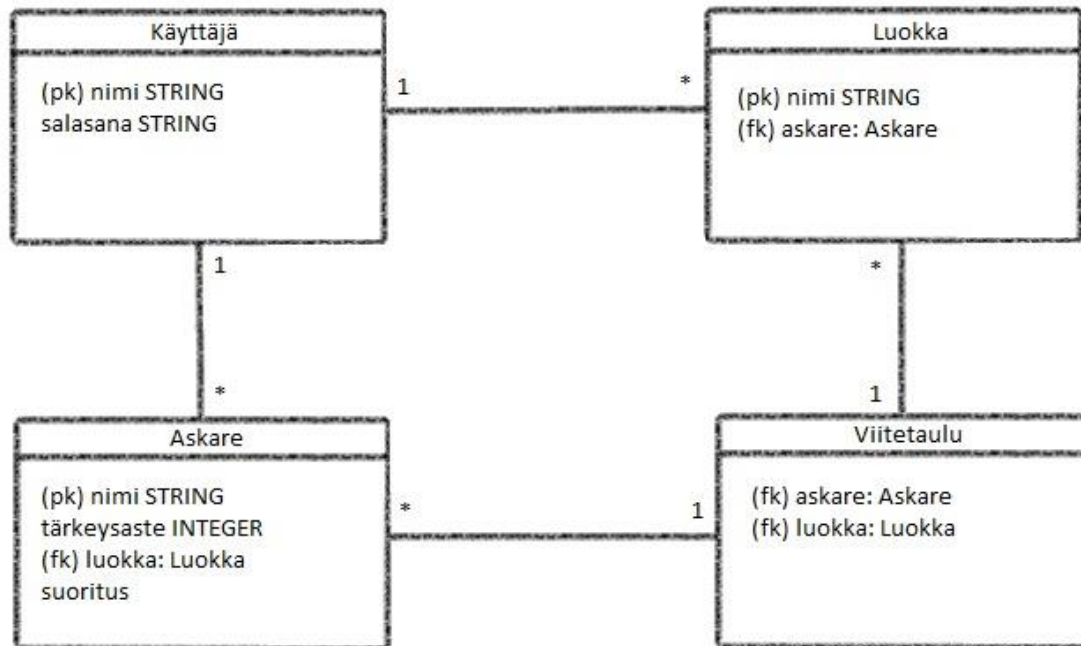
Askare:

- nimi: merkkijono max 20 merkkiä
- tärkeysaste: kokonaisluku väliltä 1-5, kuinka tärkeänä askareta pidetään
- luokka: merkkijono max 20 merkkiä, mihin ryhmään/ryhmiin askare voidaan luokitella
- suoritus: merkkijono kesken/valmis

Luokka:

- nimi: merkkijono max 20 merkkiä
- askareet: merkkijono max 20 merkkiä, mitä askareita luokkaan on liitetty

Relaatiotietokantakaavio



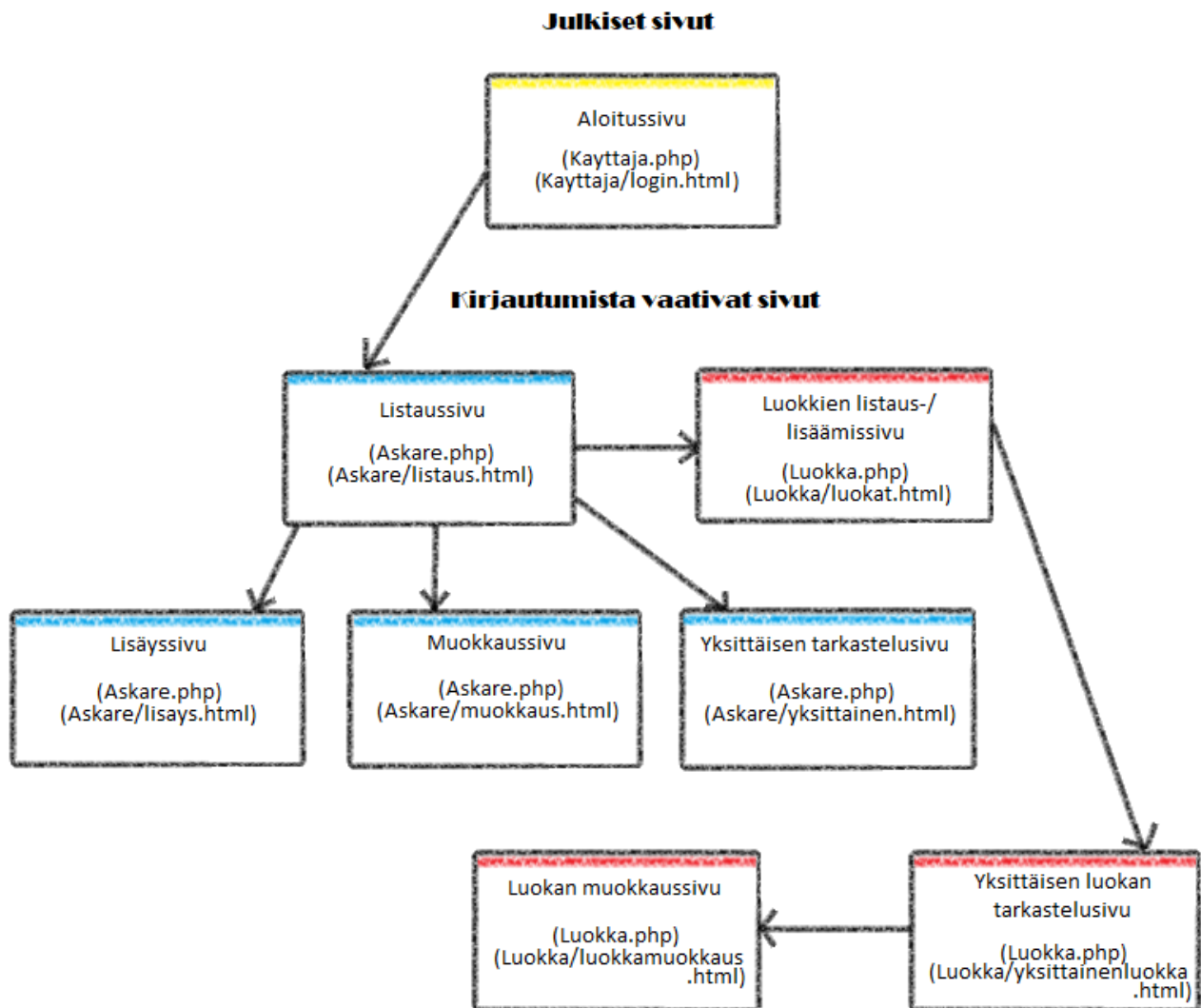
Järjestelmän yleisrakenne

Sovellukseni on toteutettu MVC-mallia (Model, View, Controller) mukaillen. Juuri mainitut löytyvät projektin kansioista app; mallit kansioista models, näkymät kansioista views ja kontrollerit loogisesti kansioista controllers. Views-kansioista löytyy erikseen jokaista luokkaa koskevat näkymät kansioittain.

Tietokantojen luomiseen, dropaamiseen ja testidatan lisäämiseen tarvittavat SQL-kyselyt löytyvät kansioista sql. Kansio doc sisältää pdf-muotoisen dokumentaatiotiedoston (jota parhaillaan tarkastelet) ja config-kansion alta taas löytyy mm. routes-tiedosto, joka sisältää kaikki sovelluksen polut ja tarvittavat ohjaukset eri metodeille.

Sovellusta toteutettaessa on käytetty perusohjelmointikielenä PHP:tä, muotoilussa CSS:ää ja näkymien toteuttamisessa HTML:ää. Perusrakenteessaan sovelluksessa on kaksi CRUD (create, read, update, delete) -nelikon toteuttavaa tietokohdetta; askare ja luokka. Sovelluksen taustalla loogisena tietokohteena on toteutettuna myös käyttäjä, joka hallinnoi sitä, mihin kaikkiin näkymiin sovelluksessa voi päästä. Kirjautumaton (ja rekisteröitymätön) käyttäjä voi tarkastella ainoastaan kirjautumissivua, kun taas kirjautuneelle käyttäjälle aukeaa sovelluksen loputkin näkymät ja tärkeimmät toiminnallisuudet.

Käyttöliittymä ja järjestelmän komponentit



Kaikissa sovelluksen näkymissä on sivun yläreunassa navigointipalkki, josta löytyy polut muistilistan sekä luokkien listaussivulle. Palkista löytyy myös kirjaudu sisään/kirjaudu ulos -nappi.

Käynnistys-/käyttöohje

Harjoitustyö on asennettuna osoitteessa <http://tviivi.users.cs.helsinki.fi/tsoha/login>. Tietokantatauluja pääsee tarkastelemaan selaimen kautta osoitteessa <http://tviivi.users.cs.helsinki.fi/tsoha/tietokantayhteys>.

Sovellukseen voi kirjautua esimerkiksi tunnuksilla:

- käyttäjätunnus: testikäyttäjä
- salasana: testisalasana