

Muistilista-projekti

Johdanto

Muistilista-ohjelman tarkoituksena on sallia käyttäjien ylläpitää omaa muistilistaa, joka voi sisältää erilaisia askareita tai vastaavia tehtäviä ja antaa kullekin listalla olevalle askareelle prioriteetti, jonka perusteella käyttäjä voi priorisoida omaa työskentelyään. Askareet voi myös luokitella ”tehtäväluokkiin”, jotka sisältävät mahdollisesti useita eri askareita. Samalla yksi askare saattaa olla kuulua useisiin eri tehtäväluokkiin.

Järjestelmän toteutus tehdään users-palvelimelle käyttäen PHP:tä. Ottaen huomioon sovelluksen luonne, toteutuksen tulisi onnistua ilman erityisiä palvelinpuolen sovelluskehys- tai selainpuolen ohjelmointikielitukea. Toteutuksessa käytetään PostgreSQL-tietokantaa.

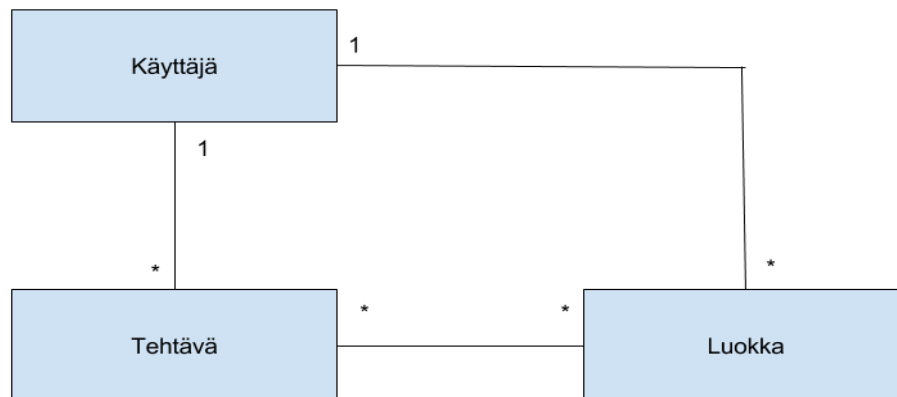
Käyttötapaukset

Järjestelmän käyttötapaukset keskittyvät muistilistan askareiden luomiseen ja näyttämiseen käyttäjäkohtaisesti. Käyttäjällä on pääsy vain omaan muistilistaansa ja hallinnoi sitä järjestelmän antamien toimintojen avulla. Näin ollen järjestelmä ei vaadi erillisiä käyttäjäryhmiä hallinnointia ja asiointia varten.

Esimerkkitapaus järjestelmän käytöstä

Käyttäjä X kirjautuu järjestelmään ja näkee omalla muistilistallaan olevat projektit ja niihin kuuluvat askareet järjestettynä prioriteetin perusteella suurimmasta prioriteetista pienimpään. Hän voi myös lisätä muistilistalle uuden askareen lomakkeella, jonka jälkeen kyseinen askare löytyy hänen muistilistaltaan sen oikeasta paikasta. Käyttäjä voi myös tarpeen mukaan poistaa ja muokata omia askareitaan ja projektejaan silloin kun askare on suoritettu tai sen luonne muuttuu, esimerkiksi jos askareen prioriteettia tai kuvausta tulee muuttaa. Tehtävät voi myös luokitella eri tehtäväluokkiin, ja päänäköymästä voi valita vain tiettyyn tehtäväluokkaan kuuluvat tehtävät näytettäväksi, jos käyttäjä haluaa hallita vain kyseisen luokan tehtäviä muistilistallaan.

Järjestelmän tietosisältö



Käyttäjä

Attribuutti	Arvojoukko	Kuvaus
käyttäjänimi	merkkijono (max 120 merkkiä)	Käyttäjän itselleen valitsema käyttäjänimi
salasana	merkkijono (max 120 merkkiä)	Käyttäjän itselleen valitsema salasana

Käyttäjä on järjestelmän käyttäjä, jolla on käyttäjänimi, salasana ja id, jonka perusteella järjestelmä voi käsitellä käyttäjälle kuuluvia muita tietokohteita.

Tehtävä

Attribuutti	Arvojoukko	Kuvaus
nimi	merkkijono (max 200 merkkiä)	Tehtävää kuvaava nimi (esim. "Siivoa huone.")
prioriteetti	kokonaisluku (1-1000)	Tehtävän prioriteetti, korkeampi luku tarkoittaa suurempaa prioriteettiä

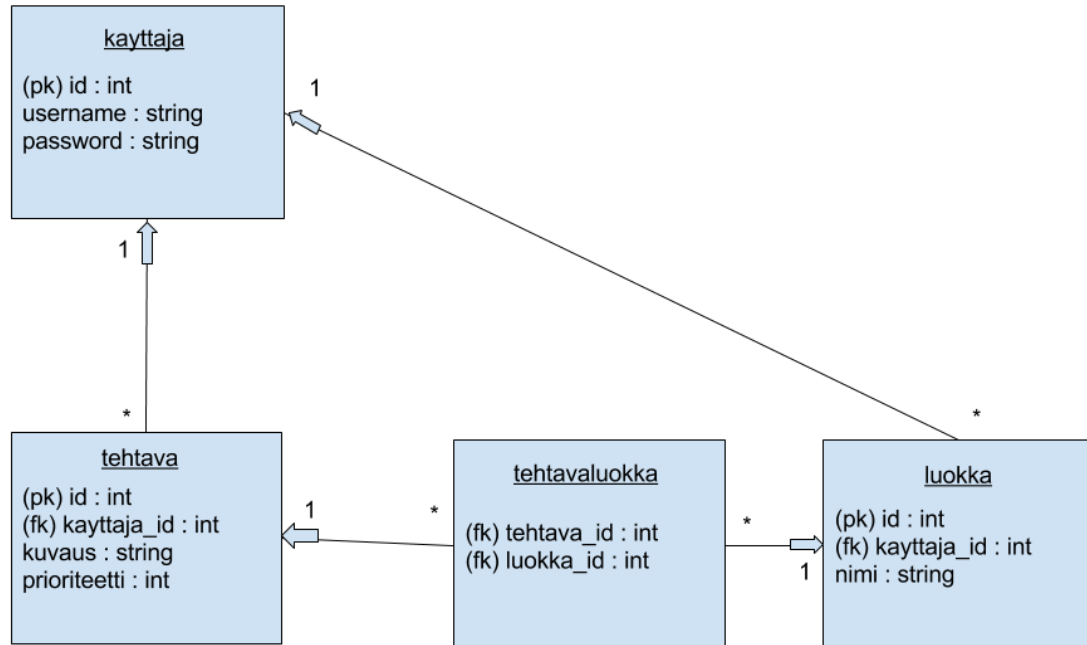
Tehtävä on yksittäinen käyttäjän järjestelmään tallentama askare, joka sisältää tehtävän kuvauksen sekä tiedon tehtävän tallentajasta, jotta jokainen käyttäjä näkee vain omat tehtävänsä. Lisäksi tehtävä sisältää tiedon tehtävän prioriteetista, jonka perusteella tehtäviä voidaan luokitella.

Luokka

Attribuutti	Arvojoukko	Kuvaus
nimi	merkkijono (max 200 merkkiä)	Tehtäviäluokkaa kuvaava nimi (esim. "Ohjelmointi")

Luokka on käyttäjän määrittelemä luokittelu, jonka alle eri tehtäviä voidaan tallentaa. Esimerkiksi opiskeluun liittyvät tehtävät voidaan tallentaa erilleen harrastuksiin liittyvistä tehtävistä. Koska luokittelut ovat myös käyttäjäkohtaisia, sisältää luokka myös tiedon tallentajastaan.

Relaatiotietokantakaavio



Käynnistys- ja käyttöohje

Sovellus on saatavilla osoitteessa <http://lsami.users.cs.helsinki.fi/muistilista/>. Sovelluksen käyttö vaatii käyttäjätunnuksen, jonka voi luoda osoitteessa <http://lsami.users.cs.helsinki.fi/muistilista/register>, mutta valmis testitili on saatavilla käyttäjätunnuksella "demo" ja salasanalla "demosalis".

Sovelluksen päänäkylässä voi luoda uusia tehtäviä muistilistalle "Lisää uusi tehtävä" painikkeella ja listalla jo olevien tehtävien kohdalta löytyvät muokkausta ja poistoa varten painikkeet. Klikkaamalla jotakin tehtävälistalla olevista luokista näkee vain tehtävät, jotka kuuluvat kyseiseen tehtäväluokkaan. Uusia tehtäväluokkia voi luoda "Hallitse tehtäväluokkia" painikkeen alta löytyvästä hallintänäkyelmästä, jossa voi myös muokata tai poistaa tehtäväluokkia.

Järjestelmän yleisrakenne

Järjestelmä on toteutettu MVC-mallin mukaisesti siten, että kontrollerit, näkymät ja mallit löytyvät app-kansion alla olevista kansioista (controllers, models, views).

Järjestelmän käyttämät apukirjastot on sijoitettu lib-kansioon, ja järjestelmän asetukset löytyvät config-kansion alta.

Järjestelmän komponentit ja käyttöliittymä

