

Tietokantasovellus - FlexDo

17. maaliskuuta 2014

1 Johdanto

Tavoitteena on kehittää tehokas järjestelmä tehtävien hallintaan. Tehtävät tulisi voida priorisoida, ja lajitella omiin luokkiinsa.

1.1 Järjestelmän tarkoitus

Järjestelmän perusta on priorisoitu tehtävälista. Järjestelmän tulisi sallia käyttäjän lisätä tehtäviä listalle, asettaa tehtäville prioriteetti, ja selata niitä eri järjestyksissä (mm. prioriteetin mukaan). Lisäksi tehtävät tulisi voida luokitella erillisiin kategorioihin, mielellään hierarkisesti.

1.2 Toteutus

Järjestelmä toteutetaan Java/Tomcat verkkosovelluksena, tietokantajärjestelmänä PostgreSQL. Järjestelmän tulisi olla käytettävissä puhtaasti html muotoisena, mutta käyttöä helpottavia lisätoimintoja voidaan tarvittaessa toteuttaa javascriptillä.

2 Yleiskuva järjestelmästä

2.1 Käyttäjäryhmät

Käyttäjä

Rekisteröitynyt ja järjestelmään sisäänkirjautunut käyttäjä

2.2 Käyttötapaukset

Kirjautuminen

Rekisteröityminen

Askareen lisäys

Käyttäjä voi lisätä uuden askareen. Askareelle annetaan lisätessä nimi, mahdollinen kuvaus ja prioriteetti.

Askareen muokkaus

Käyttäjä voi muokata järjestelmässä olevan askareen prioriteettia, nimeä sekä kuvausta.

Askareen poisto

Käyttäjä voi poistaa askareen järjestelmässä. Poiston yhteydessä voidaan joko merkitä askare suoritetuksi, jolloin se jää järjestelmään erilliseen kategoriaan kirjanpitoa varten, taikka poistaa askare kokonaan järjestelmästä.

Askareiden selaus prioriteetin mukaan

Käyttäjä voi selata järjestelmässä olevia askareita prioriteettijärjestyksessä.

Luokkien lisäys ja poisto

Käyttäjä voi lisätä ja poistaa järjestelmässä olevia askareiden luokkia.

Askareiden luokittelu

Käyttäjä voi liittää askareen luokkaan, taikka poistaa sen luokasta. Yksi askare voi kuulua useaan luokkaan.

Askareiden selaus luokittain

Käyttäjä voi selata järjestelmässä olevia askareita luokan mukaan.

Suoritettujen askareiden selaus

Käyttäjä voi selata suoritettuja askareita suoritusajankohdan mukaan järjestettynä.

Luokkien hierarkkinen järjestely

Käyttäjä voi asettaa luokkia toistensa alaluokiksi, muodostaen hierarkkisen luokkarakenteen.

3 Jatkokoehitysideoita

- Askareiden aikataulut
- Askareiden välisiä linkkejä (wiki-tyyli)
- Muistioita ("askare", joka toimii muistiona eikä suoritettavana tehtävänä) (vaatii askareiden väiset linkit)
- Askareiden välisiä vaatimuksia (Tehtävä B ei ole suoritettavissa ennenkuin tehtävä A on suoritettu)
- Projektinäkymä (kartoitettu näkymä aikataulusta ja askareiden väisistä riippuvuuksista)