Drinkkiarkisto web-sivusto suunnitelma

Käyttötapauskaavio:

Drinkkiarkisto:

Super user

user

admin

Login

Rekisteröidy

Luo drinkki Login

Ehdota drinkki

Muokkaa drinkkiä

Poista drinkki

Hyväksy /hylkää ehdotus

Poista käyttäjä

Ylennä käyttäjä super useriksi

Katso drinkin tietoja

Katso käyttäjän tietoja

Hae drinkkejä

User: Login, Rekisteröidy, Katsele drinkkejä, hae drinkkejä, ehdtota drinkkekä

Super user: Login, Rekisteröidy, Katsele drinkkejä, hae drinkkejä, ehdtota drinkkekä, luo drinkkejä

Admin: Login, Rekisteröidy, Katsele drinkkejä, hae drinkkejä, ehdtota drinkkekä, luo drinkkejä, poista drinkkejä, muuta drinkkejä, hyväksy/hylkää ehdotus, ylennä käyttäjä super useriksi, poista käyttäjä, katso käyttäjän tietoja.

Käyttötapaukset:

Userin käyttötapaukset:

Login: jos user on jo rekisteröitynyt voi hän kirjautua sisään järjestelmään omalla käyttäjätunnuksellaan.

Rekisteröidy: jos userilla ei ole käyttäjätunnusta hän voi luoda uuden antamalla käyttäjätunnuksen (min 5 max 15 merkkiä), salasanan (min 5 max 15 merkkiä) ja emailin (pakollinen). Käyttäjätunnusta ei saa olla jo olemassa.

Katsele drinkkejä: Haun jälkeen näytöllä on linkkejä eri drinkkeihin. Linkkiä klikkaamalla drinkin tiedot aukeavat.

Hae drinkkiä: drinkkejä voi hakea sanan perusteella, aakkosjärjestyksessä tai tyypin mukaan (esim. cocktail). Tulokset näkyvät linkkeinä sivulla.

Ehdota drinkkiä: lomake jonne user voi syöttää drinkkiehdotuksia (nimi, pääainesosa, määrä, muut ainesosat ja määrät max 4 kpl). Ehdotetun drinkin tyypiksi tulee ”ehdotus” tietokantaan. Admin hyväksyessään sitten muuttaa drinkin tyyppiä.

Super user:

Samat toiminnallisuudet kuin userilla. Lisäksi super user voi luoda drinkkejä.

Luo drinkki: Super user täyttää lomakkeen joka on samanlainen kuin ”ehdota drinkkiä” paitsi että on pakollisena drinkin tyyppi kenttä. Drinkin tiedot tallennetaan kantaan. Syötteillä on tarkistuksia.

Admin:

Kaikki samat toiminnallisuudet kuin super userilla ja lisäksi:

Muokkaa drinkkiä: Admin haun jälkeen drinkin tietoja voi muokata ja tallentaa kantaan.

Poista drinkki: Admin haun jälkeen drinkin voi poistaa kannasta.

Poista käyttäjä: Käyttäjä haun jälkeen käyttäjän voi poistaa järjestelmästä.

Ylennä käyttäjä super useriksi: Admin haun jälkeen käyttäjän tilan voi ylentää super useriksi.

Hyväksy/hylkää ehdotus: Ehdotuksia voi hakea ja sen jälkeen avata yksitellen ja hyväksyä jolloin ne tallennetaan kantaa tai hylätä jolloin ne poistetaan kannasta.

Katso käyttäjän tietoja: Käyttäjää voi hakea ja sen jälkeen klikata linkistä ja katsoa käyttäjän tietoja.

Käsitekaavio:

Login

Drinkki

Drinkki\_tyyppi

Tyyppi

DrinkkiAinesosa

Ainesosa

Taulukuvaukset:

Taulu:Login

String id => javalla generoitava yksilöivä id

String name => käyttäjän nimi

String password => käyttäjän salasana

String authority => admin, user, superuser

String email => käyttäjän email

Taulu Drinkki:

String drinkki\_id => javalla generoitava yksilöivä id

String drinkki\_name => drinkin nimi

Taulu DrinkkiAinesosa:

String drinkki\_id => minkä drinkin id

String ainesosa\_id => minkä ainesosan määrä

Int maara = > paljonko tähän drinkkin tätä ainesosaa laitetaan

Taulu Ainesosa:

String ainesosa\_id => Javalla luotu yksilöivä id

String ainesosa\_name => ainesosan nimi

Taulu DrinkkiTyyppi:

String drinkki\_id => minkä drinkin id

String tyyppi\_id => minkä tyypin id

Taulu Tyyppi:

String tyyppi\_id => Javalla luotu yksilöivä id

String tyyppi\_name => tyypin nimi

Relaatiotietokantakuvaus:

Login

Drinkki

<<key>>drinkki\_id:String

Drinkki\_name:String

DrinkkiTyyppi

<<key>> id:String

<<fkey>>drinkki\_id;String

<<fkey>> tyyppi\_id;String

Tyyppi

<<key>>tyyppi\_id:String

Tyyppi\_name:String

DrinkkiAinesosa

<<key>>id:String

<<fkey>>drinkki\_id;String

<<fkey>>ainesosa\_id;String

Maara:Int

<<key>>id;String

Username;String

Password;String

Authority;String

Ainesosa

<<key>>ainesosa\_id;String

Ainesosa\_name;String

Järjestelmän rakenne:

Spring sovelluskehys, pyynnöt ohjataan controllerille paketissa werkko.controller. Controlleri validoi pyynnöt ja ohjaa oikeaan palveluun tarvittaessa. Palvelut löytyvät werkko.services paketista. Palvelut käyttävät repositoryjä tietokantaan tallettamiseen. Repositoryt löytyvät werkko.repositories kansiosta. Lopulta controlleri ohjaa pyynnön oikealle jsp sivulle. Jsp sivut löytyvät WEB-INF kansiosta.

Käyttöliittymä:

Login

Rekisteröidy

Haku

Admin

Luo Drinkki

Admin, superuser

admin

Drinkin tiedot

Ehdota drinkkiä

Hyväksy ehdotus

Käyttäjä

Admin drinkin tiedot

Asennustiedot:

[git@github.com:miklehti/tietokantasovellus2.git](mailto:git@github.com:miklehti/tietokantasovellus2.git)

Sovellus tehty Netbeans 7.3.1 (Glassfish 4.0)

Pom.xml pitäisi olla kaikki riippuvuudet jotta sovelluksen voi ajaa Netbeansissä, pyörii localhostisssa

Käyttöohje:

Admin user:

Username: mikko

Password: secret

User:

Username: lase

Password: secret

Superuser

Username:Jouni

Password:secret

Testaus, tunnetut bugit jatkokehitys, omat kokemukset:

Jonkun verran yksikkötestejä, integraatiotestejä ja systeemitestejä Junitilla ja Seleniumilla, ei tarpeeksi kattavasti testejä tällä hetkellä. Loput testattu suoraan selaimesta.

Virheitä sessiossa, kun tarpeeksi kauan luo drinkkejä tms. niin drinkin nimi kenttään tulee jotain ihmeellistä syötettä valmiiksi tyyliin {drink\_name=…

Melko tyytyväinen data, service ja repository pakettien luokkiin. Controller luokasta tuli aika sotkuinen.

Security konffausta ei ehtinyt => nyt tarkistetaan joka kutsussa onko istunto voimassa. => selkeyttäisi controlleri luokkaa.

Localhost rumasti kovakoodattu, voisi luoda yhden paikan missä ympäristö mainittu.

Käyttöliittymä on ruma

Kanta tukee että yhdellä drinkillä on monta tyyppiä, käyttöliittymä ei tue vielä tätä=)

Drinkin luonti Springillä oli vaivatonta. Spring vyörytti drinkkiolion tiedot oikein kannan tauluihin. Jostain syystä delete ja update eivät toimineet samalla tavalla fiksusti. En ehtinyt tutkia tätä mutta veikkaan että jollain asetuksella nuokin olisivat toimineet.

Springin konfiguraatiot hankalia.