

scheduler

Janne Suomalainen

29. huhtikuuta 2017

Sisältö

1	Johdanto	2
2	Yleiskuva järjestelmästä	2
3	Järjestelmän tietosisältö	5
4	Relaatiotietokantakaavio	6
5	Järjestelmän yleisrakenne	6
6	Käyttöliittymä ja järjestelmän komponentit	6
7	Asennustiedot	7
8	Käynnistys- tai käyttöohje	7
9	Jatkokehitys	7

1 Johdanto

SCHEDULER on pieni www-alustainen tietokantajärjestelmä yrityksen työvuorolistojen laatimiseen, jakamiseen ja säilömiseen; ohjelmalla pystyy suunnitella ja tarkastella työvuoroja. Excel-pohjaiset työvuorolistat ovat – vaikka nerokkaita – taulukkolaskentaan tottumalle epäintuitiivisia käyttää, niiden jakaminen työntekijöille on varsin rajoittunutta ja niiden päivittäminen on työlästä. SCHEDULER-ohjelman tavoitteena on tarjota varteenotettava ratkaisu edellä mainittuihin ongelmiin.

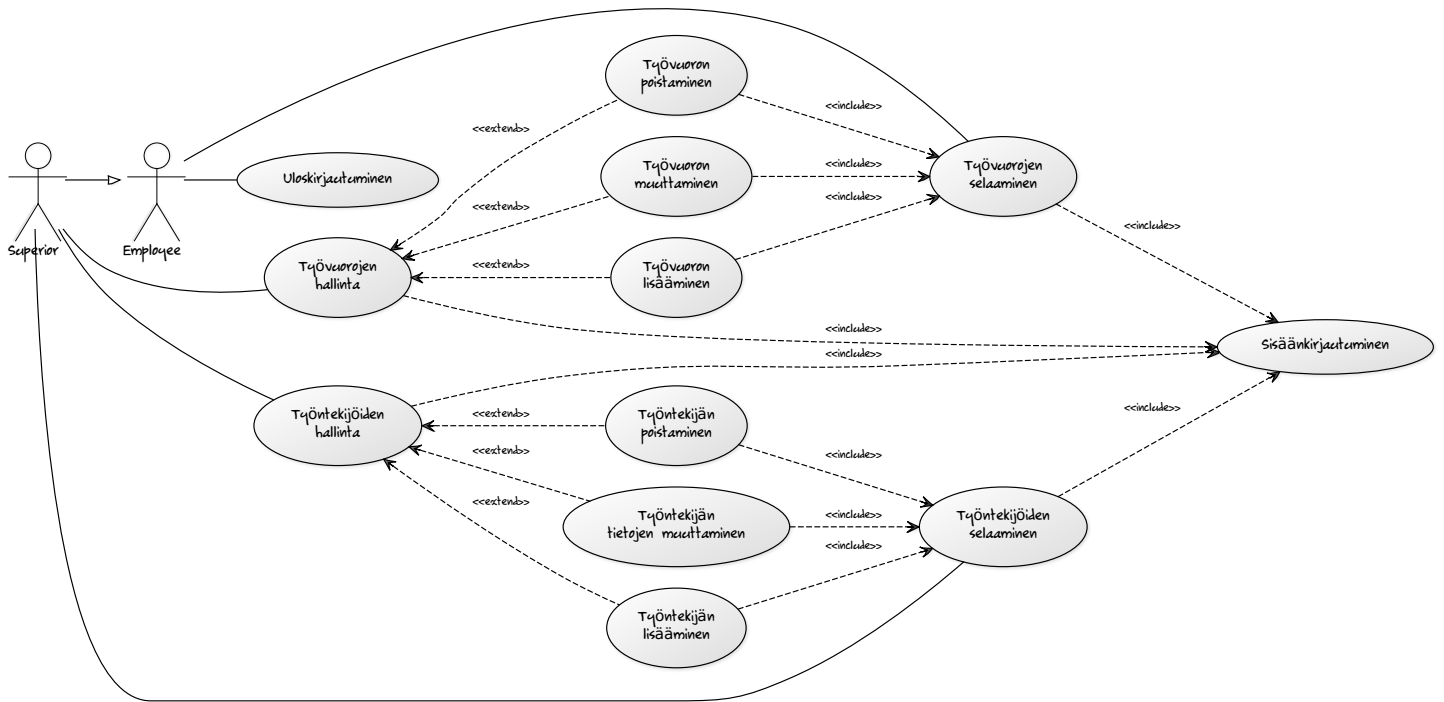
SCHEDULER toteutetaan Heroku-pilvipalveluun. Pääasiallinen ohjelmointikieli on Java 8; ohjelma käyttää Spark-nimistä web-sovelluskehystä, Javan tärkeimpiä kirjastoja sekä sql2o- ja Lombok-kirjastoa. Sovelluksen näkymät toteutetaan Thymeleaf-mallimoottorilla. Tallennustoiminnallisuudesta vastaa PostgreSQL-tietokanta.

2 Yleiskuva järjestelmästä

Yleiskuva SCHEDULER-järjestelmän toiminnallisuudesta on esitetty kuvassa 1. Kaavion tueksi annetaan lyhyet kuvaukset järjestelmän sidosryhminä toimivista käyttäjäryhmistä ja esitetään muutama käyttötapauksista tarkemmin.

Employee kuvaa tässä yrityksen työntekijää, joka on rekisteröity järjestelmään ja joka ei ole esimiesasemassa.

Superior on järjestelmään rekisteröity esimiesasemassa toimiva yrityksen työntekijä.



Kuva 1: Käyttötapauskaavio näyttää järjestelmän sidosryhmät ja miten ne liittyvät järjestelmään.

Employee selaa työvuoroja

Käyttötapausten käyttäjä on työntekijä, jonka tavoitteena on saada tietoa omista työvuoroistaan. Tapauksen esiehtona on, että työntekijä on kirjautunut sisään, jälkiehtona, että työntekijä on saanut tarvitsemansa tiedon työvuoroistaan. Käyttötapaus etenee seuraavasti:

1. Työntekijä aloittaa työvuorojen selaustoiminnon. Hän asettaa haetuille työvuoroille rajoitteita esimerkiksi ajankohdan tai toimipisteen mukaan.
2. Järjestelmä näyttää työntekijälle osoitetut työvuorot ottaen huomioon työntekijän asettamat rajoitteet.
 - a) Järjestelmä ilmoittaa, mikäli työntekijän asettamat rajoitteet ovat ristiriitaisia.
 - b) Järjestelmä ilmoittaa, mikäli työntekijän asettamien rajoitteiden mukaisia työvuoroja ei löydy.
3. Työntekijä tutkii työvuoroja.

Superior lisää työvuoron

Käyttötapauksen käyttäjä on esimies, jonka tavoitteena on asettaa työvuoro. Tapauksen esiehtona on, että esimies on kirjautunut sisään, jälkiehtona, että haluttu työvuoro on asetettu. Käyttötapaus etenee seuraavasti:

1. **Suoritetaan käyttötapaus Työvuorojen selaaminen.**
2. Esimies aloittaa työvuorojen lisäystoiminnon. Hän valitsee asianomaiset työntekijät, toimipisteen ja työvuoron alkamis- ja päättymisajat.
3. Järjestelmä vahvistaa työvuoron lisäyksen.
 - a) Järjestelmä ilmoittaa, mikäli esimiehen valinnat ovat ristiriitaisia.
 - b) Järjestelmä ilmoittaa, mikäli työvuoro menee päällekkäin valittujen työntekijöiden aiemmin asetettujen työvuorojen kanssa.

Superior muuttaa työvuoroa

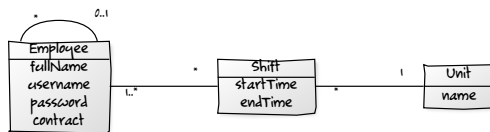
Käyttötapauksen käyttäjä on esimies, jonka tavoitteena on muuttaa jo asetettu työvuoro. Tapauksen esiehtona on, että esimies on kirjautunut sisään ja että työvuoroja on asetettu. Tapauksen jälkiehtona on, että haluttu työvuoro on muutettu. Käyttötapaus etenee seuraavasti:

1. **Suoritetaan käyttötapaus Työvuorojen selaaminen.**
2. Esimies valitsee muutettavan työvuoron ja käynnistää työvuoron muuttamistoiminnon. Hän valitsee uudet työvuoron alkamis- ja päättymisajat.
3. Järjestelmä vahvistaa työvuoron muuttamisen.
 - a) Järjestelmä ilmoittaa, mikäli esimiehen valinnat ovat ristiriitaisia.
 - b) Järjestelmä ilmoittaa, mikäli muutettu työvuoro menee päällekkäin sille asetettujen työntekijöiden muiden työvuorojen kanssa.

Superior poistaa työntekijän

Käyttötapauksen käyttäjä on esimies, jonka tavoitteena on poistaa työntekijä järjestelmästä. Tapauksen esiehtona on, että esimies on kirjautunut sisään, jälkiehtona, että poistettua työntekijää ei enää löydy järjestelmästä. Käyttötapaus etenee seuraavasti:

1. **Suoritetaan käyttötapaus Työntekijöiden selaaminen.**
2. Esimies valitsee poistettavan työntekijän ja aloittaa työntekijän poistamistoiminnon.
3. Järjestelmä vahvistaa työntekijän poistamisen.



Kuva 2: Käsitekaavio on järjestelmään säilöttävälle tiedolle käsitetason malli, jonka perusteella johdetaan toteutustason relaatiotietokantakaavio.

3 Järjestelmän tietosisältö

Järjestelmän tietosisältö kuvataan käsitekaavion (Kuva 2) avulla. Määritellään kaavion tietokohteet yksitellen.

Tietokohde: Employee

Attribuutti	Arvojoukko	Kuvailu
fullName	Merkkijono	Työntekijän koko nimi muodossa <i>Etunimi Sukunimi</i>
username	Merkkijono, uniikki	Työntekijän käyttäjätunnus järjestelmään
password	Merkkijono	Työntekijän salasana järjestelmään
contract	Desimaaliluku, vähintään nolla, oletusarvoisesti 37,5	Työntekijän työsopimuksessa lukeva viikkotuntimäärä

Työntekijällä on enintään yksi esimies, joka on myös työntekijä. Edelleen esimiehellä voi olla alaisia. Työntekijällä voi olla useita työvuoroja.

Tietokohde: Shift

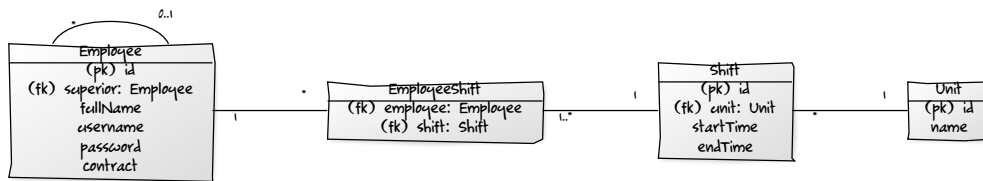
Attribuutti	Arvojoukko	Kuvailu
startTime	Aikaleima	Työvuoron alkamisajankohta, joka sisältää sekä päivämäärän että kellonajan
endTime	Aikaleima	Työvuoron päättymisajankohta, joka sisältää sekä päivämäärän että kellonajan

Työvuoroon liittyy aina vähintään yksi työntekijä. Lisäksi työvuoroon liittyy täsmälleen yksi toimipiste.

Tietokohde: Unit

Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono	Toimipisteen nimi

Toimipisteeseen voi liittyä useita työvuoroja.



Kuva 3: Relaatiotietokantakaaviossa tiedon säilömiseen käytettävän tietokannan rakenne esitetään kaaviokuvana.

4 Relaatiotietokantakaavio

SCHEDULER-järjestelmän tietokantakaavio on esitetty kuvassa 3.

5 Järjestelmän yleisrakenne

Tietokantasovellusta tehdessä on noudatettu MVC-mallia. Tavanomaisesta kerrokseen perustuvasta hakemistorakenteesta (controllers, views, models) poiketen sovellus on pakattu ominaisuuksittain (employee, index, login, shift, unit); esimerkiksi työntekijää kuvaava malli Employee, sen DAO-malli EmployeeDao ja siihen liittyvien näkymien kontrolleri EmployeeController löytyvät kaikki hakemistosta employee. Yleishyödyllisiä apuluokkia löytyy hakemistosta util. Kaikki sovelluksen näkymät löytyvät vuorostaan hakemistosta templates.

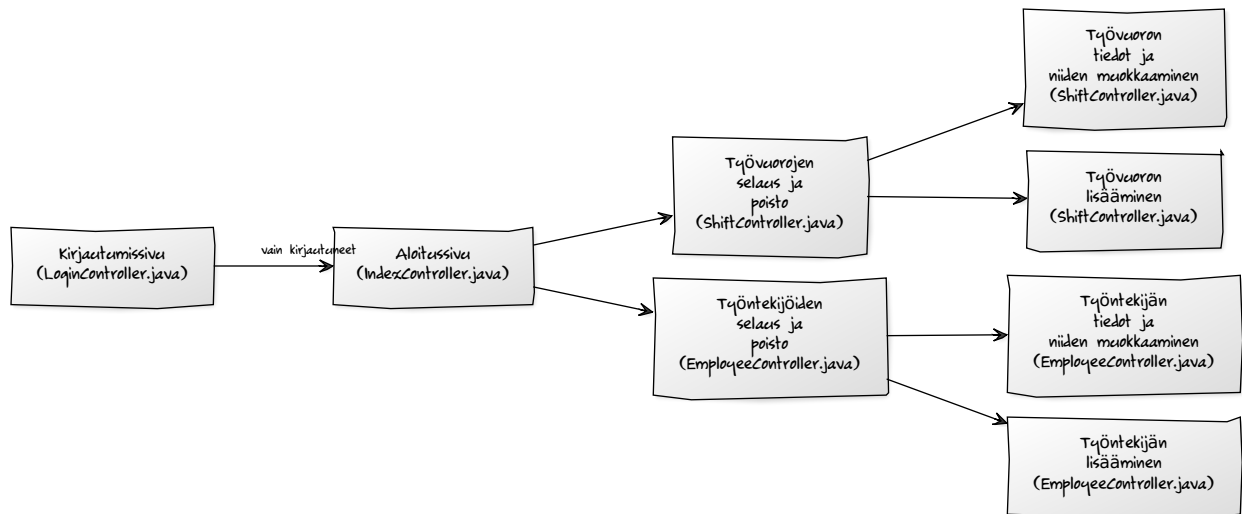
Sovellus käyttää kahta istuntoattribuuttia: loggedOut ja currentUser. Ensimmäisen arvo on tosi, kun käyttäjä uloskirjautuu onnistuneesti. Jälkimmäisen arvo on kirjautuneen käyttäjän käyttäjätunnus.

6 Käyttöliittymä ja järjestelmän komponentit

SCHEDULERin käyttöliittymän näkymien välisiä yhteyksiä on esitetty kuvassa 4. Näiden lisäksi sivustolla on navigaatiopalkki, jonka johdosta kirjautunut käyttäjä pääsee mistä tahansa näkymästä aloitussivulle sekä työvuorojen ja työntekijöiden selaukseen ja poistoon. Myös uloskirjautuminen tapahtuu navigaatiopalkin kautta.

7 Asennustiedot

Sovellus on asennettu Heroku-pilvipalveluun. Aktiivinen, kuunneltava portti asetetaan ohjelman ConnectionUtil-luokan getHerokuAssignedPort()-metodissa. Oletusarvoisesti palvelu kuuntelee porttia 4567. Lisäksi sovellus hyödyntää Herokun



Kuva 4: Käyttöliittymän näkymien välisiä yhteyksiä on luontevaa esittää kaavioina.

tarjoamaa tietokantapalvelua (PostgreSQL), johon luodaan JDBC-yhteys samaisen luokan getConnection-luokan avulla. Muutokset kyseisissä metodeissa on riittävä asennusympäristön vaihtamiseen.

8 Käynnistys- tai käyttöohje

Sovellus löytyy osoitteesta <https://onlinescheduler.herokuapp.com/>. Testikäyttäjätunnuksen *esimes1* salasana on *password*.

9 Jatkokehitys

Käyttäjien roolit. Ei-esimiesten oikeuksia työntekijöiden ja työvuorojen muokkaukseen tulisi rajoittaa.

Työsopimuksen täyttyminen. Toiminnallisuus tarkistaa työntekijän keskimääräinen viikotuntimäärä ja verrata sitä työsopimukseen.

Älykäs työvuorottaja. Osaisiko sovellus itse tarkistaa työvuorot tai luoda tasapainoisia työvuorolistoja napin painalluksella?