

Yhteinen kalenteri

Toteutusdokumentti

Helsingin yliopisto
Tietojenkäsittelytieteen laitos
Tietokantasovellusten harjoitustyö
Ohjaaja: Arttu Valo

JOHDANTO	2
OHJELMISTON YLEISRAKENNE.....	3
<i>Yleistä</i>	<i>3</i>
<i>Käyttäjät</i>	<i>4</i>
HTML-SIVUT	4
<i>index.html</i>	<i>4</i>
<i>tervetuloa.htm</i>	<i>4</i>
<i>luo_tunnus.htm.....</i>	<i>4</i>
SERVIEITIT	4
<i>Navi</i>	<i>4</i>
<i>LuoTunnus</i>	<i>5</i>
<i>OmatTiedot</i>	<i>6</i>
<i>LuoRyhma.....</i>	<i>6</i>
<i>OmatRyhmät</i>	<i>7</i>
<i>RyhmanAjat</i>	<i>7</i>
<i>Kalenteri.....</i>	<i>8</i>
<i>KalenteriYleinen</i>	<i>10</i>
<i>VuosiKalenteri</i>	<i>10</i>
<i>TeeVaraus.....</i>	<i>10</i>
<i>CheckDate</i>	<i>12</i>
ASENNUSTIEDOT	12
<i>Tiedostojen sijainti.....</i>	<i>12</i>
KÄYNNISTYSOHJE	12
<i>Tarvittavat osoitteet</i>	<i>12</i>
LIITTEET	
LIITE 1: <i>Tehtäväkuvaus</i>	
LIITE 2: <i>Ajankäyttöraportti</i>	
LIITE 3: <i>Oma arvio kurssista</i>	
LIITE 4: <i>Ohjelmakoodit</i>	
Navi.java	
LuoTunnus.java	
OmatTiedot.java	
LuoRyhma.java	
OmatRyhmät.java	
RyhmanAjat.java	
Kalenteri.java	
KalenteriYleinen.java	
VuosiKalenteri.java	
TeeVaraus.java	
CheckDate.java	
tarkista.js	

Johdanto

Yhteinen kalenteri on järjestelmä, jolla osaston työntekijät voivat varata kokousaikoja ja välittää osaston muille henkilöille tietoa tavoitettavuudestaan. Kalenteri toimii yrityksen intranetissä. Kalenterista näkyy kunkin henkilön julkinen kalenteri, eli mitkä ajat hänellä ovat vapaita ja mitkä on varattu, johonkin tarkoitukseen. Kalenterin aikoja voi katsoa joko viikkotasolla tai vuositasolla. Rekisteröitynyt käyttäjä voi kirjata järjestelmään tietoa erilaisista henkilöryhmistä (mm. työryhmistä tai kokouksen osanottajista). Ryhmien kohdalla saadaan näkyviin ryhmälle varatut kokousajat sekä sellaiset ajat jotka ovat vapaita kaikille ryhmän jäsenille. Rekisteröitynyt käyttäjä voi tehdä varauksen henkilön tai ryhmän kalenteriin. Varaukseen liitetään tieto varaajasta ja varausajasta.

Varauksen lyhin mahdollinen kesto on puoli tuntia. Varauksen alkamisajan voi myös valita puolen tunnin välein. Yhden varauksen kesto on korkeintaan 24h. Jos haluaa tehdä seuraavalle päivälle jatkuvan varauksen, järjestelmää huomaa sen ja tekee varauksen vain yhden päivän loppuun ja käyttäjän täytyy tehdä uusi varaus alkamaan seuraavan päivän alusta.

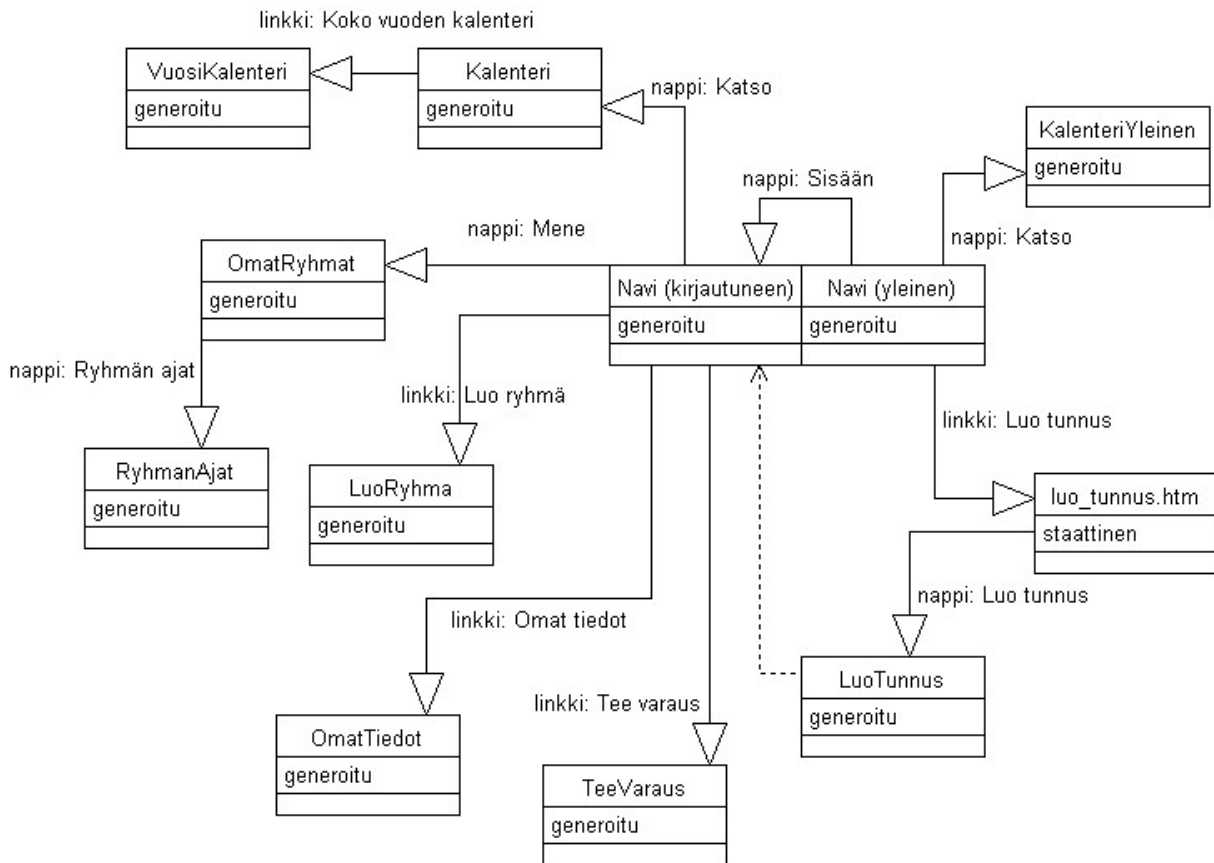
Ylläpitäjällä on oikeudet poistaa kaikkien tekemiä varauksia ja ylläpitää kaikkia ryhmiä. Toteutuksessa ei ole ylläpitäjälle käyttöliittymää henkilön poistamiseksi tietokannasta.

Järjestelmää on mahdollista käyttää tavallisen www-selaimen kautta.

Järjestelmä on toteutettu Java Servlettejä ja Postgres relaatiotietokantaa hyväksikäyttäen. Toteutusympäristö on tietojenkäsittelytieteen laitoksen db-tietokantapalvelin ja siellä käytössä oleva jserv-ympäristö. Yhteys tietokannan ja Servlettien välillä hoidetaan jdbc-tietokantaajurin välityksellä.

Tässä dokumentissa on kerrottu kuinka yllämainitun järjestelmän toteutus hoidettiin, ja käydään läpi kaikki järjestelmään kuuluvat komponentit. Järjestelmä on testattu IE 6:lla, Mozilla 1.2:lla ja Netscape 6.2:lla. Ainakin näillä selaimilla järjestelmän ulkoasu on juuri sellainen kuin haluan. Netscapella saattaa puuttua yksi viiva varauksen ylä- tai alareunasta, mutta ei muuta. Servlettien toteutuksista kertovissa kappaleissa mainitaan monesti rivinumero lauseen jälkeen. Tämä tarkoittaa, että lauseen esittämän asian toteutus löytyy kyseisen servletin koodista suunnilleen rivinumeron paikkeilta.

Ohjelmiston yleisrakenne



Yleistä

Ohjelmisto koostuu muutamasta staattisesta HTML-sivusta sekä joukosta servlettejä, jotka toteuttavat kalenterin toiminnallisuuden. Staattisia HTML-sivuja on vain kolme. Koska toteutin ohjelmiston kehyksien avulla, on index.html tiedostoon määriteltä frameset-elementeillä selainikkunan jakautuminen kahteen osaan; vasemman reunan navigointipalkkiin ja oikean puolen pääikkunaan. Pääikkunaan avautuu navigointipalkista valitut linkit. Poikkeuksena sisäänkirjautuminen, jonka jälkeen selainikkunan vasempaan kehykseen ladataan rekisteröityneen käyttäjän navigointipalkki. Tosin sama servletti hoitaa molempien navigointipalkkien tuottamisen. Kuvia ei ole käytössä kuin kaksi. Kale_bg.gif ja transparent.gif ovat hyvin pieniä gif-tiedostoja, joita käytetään viikkokalenterin tuntiviivojen piirtämiseen ja näyttämiseen. Kuvan tekeminen oli hankalaa, koska ei ole selviä proseduureja, jotka hoitavat vain yhden määrätyn homman. Yritin jollain tasolla kuvata servlettien välisiä yhteyksiä. Kuvasta puuttuu esim. OmatTiedot servletin tietojen tallennusreitti, koska tiedot lähetetään servletille itselleen, jonka jälkeen servletti vuorostaan generoi kiitossivun.

Käyttäjät

Käyttäjät voivat luoda itselleen halutessaan tunnuksen, joka on heti voimassa. Oikeassa tuotantojärjestelmässä olisi varmempaa hyväksyttää käyttäjät ylläpitäjällä. Järjestelmään on luotu jo valmiiksi joitain testikäyttäjiä, joilla on joitain varauksiakin tietokannassa. Tavalliset käyttäjät, jotka näkyvät navigointipalkin henkilön valintalaatikossa, on rekisteröity järjestelmään salasanaan oma tunnus. Näin ollen voi käydä katsomassa kalenteria eri henkilöiden näkökulmasta. Järjestelmässä on myös admin –niminen käyttäjä. Tämä on ylläpitäjä, joka voi poistaa kaikkien henkilöiden tekemiä varauksia ja ylläpitää kaikkia olemassa olevia ryhmiä. Myös kalenterin tulostusasu on hieman erilainen testitarkoituksen vuoksi ylläpitäjällä. (kts. Kalenteri-servletti) Admin käyttäjän salasana on myös sama kuin tunnus eli admin.

HTML-sivut

index.html

Staattinen HTML-sivu, jossa määritellään kahden selainikkunan jakavan kehyksen suhteet.

tervetuloa.htm

Staattinen HTML-sivu, joka näytetään aina kun sivulle tullaan ensimmäistä kertaa , tai kun rekisteröitynyt käyttäjä valitsee 'Kirjaudu ulos'-linkin navigointipalkista. Sivulla on tervetuloivotus ja sivulla kerrotaan lyhyesti mitä voi tehdä ilman käyttäjätunnusta ja käyttäjätunnuksen kanssa.

luo_tunnus.htm

Staattinen HTML-sivu, joka sisältää lomakkeen tunnuksen luontia varten. Sivulla on kentät käyttäjätunnusta, etunimeä, sukunimeä, sähköpostiosoitetta ja puhelinnumeroa varten. Lisäksi käyttäjän täytyy syöttää haluamansa salasana kaksi kertaa. Salasana-kentät ovat password-tyyppiä, joten käyttäjälle näytetään vain tähtiä salasanaa syötettäessä. Pakolliset kentät on merkitty punaisella tähdellä. 'Luo tunnus'-napin painallus lähettää kaikkien kenttien tiedot parametreina servletille LuoTunnus. Tietojen tarkistus tehdään servietissä.

Servletit

Navi

Navi servletti generoi sivun vasemmassa laidassa olevan navigointipalkin. Navigointipalkin sisältö riippuu siitä, onko käyttäjä rekisteröitynyt järjestelmään vai ei. Sekä

rekisteröityneen että tavallisen käyttäjän navigointipalkissa on valintalaatikot, joista voi valita katsottavaksi joko ryhmän tai yksittäisen henkilön kalenterit. Katso-napin painallus lähettää valintalaatikkojen tiedot parametreina Kalenteri-servletille, joka hoitaa kalenterin näyttämisen.

Ryhmän ja henkilön valintalaatikot ovat JavaScriptin pätjän kautta suhteessa toisiinsa siten, että aina kun käyttäjä painaa hiirellä toisessa valintalaatikossa, niin toisen valintalaatikon arvo nollaantuu (rivit 91, 124). Näin ollen Kalenteri-servletin ei tarvitse tarkistaa onko molemmat kentät valittu yhtä aikaa vai vain toinen niistä. Jos JavaScript on asetettu selaimesta pois päältä, niin ainoa ongelma tulee silloin kun joku henkilö on valittu ja yritetään katsoa ryhmän kalenteria. Tällöin oikeaan kehykseen tulee vain valitun henkilön kalenteri. Vasta kun poistaa henkilövalinnan ja jättää vain ryhmävalinnan, vaihtuu kalenteri ryhmän kalenteriksi.

Rekisteröimätön käyttäjä näkee palkissa kentät käyttäjätunnukselle ja salasanalle, sekä Sisään-napin. Napin painallus lähettää tunnus- ja salasana-kenttien sisällön parametreina Navi-servletille itselleen. Servletti tarkistaa löytyykö tietokannan salasana-taulusta yhtään parametreja vastaavaa riviä (rivi 27). Jos rivejä ei löydy, annetaan virheilmoitus tunnus ja salasana kenttien yläpuolelle. Muussa tapauksessa Navi-servletti generoi käyttäjälle rekisteröityneen käyttäjän navigointipalkin.

Käyttäjän on mahdollista luoda käyttäjätunnus itselleen 'Luo tunnus'-linkin kautta. Linkin valitseminen vaihtaa oikeaan kehykseen `luo_tunnus.htm` sivun.

Kun rekisteröitynyt käyttäjä kirjoittautuu sisään, tallennetaan käyttäjätunnus HttpSession-olioon. Näin se kulkee mukana kalenterin toimintoja käytettäessä. Kun käyttäjä valitsee 'Kirjaudu ulos'-linkin navigointipalkista, nollataan HttpSession-olion sessionTunnus-kenttä ja näytetään käyttäjälle rekisteröitymättömän käyttäjän navigointipalkki vasemmassa kehyksessä ja oikeaan kehykseen vaihdetaan tervetulotoivotus.

Parametrit:

submit – nähdään onko käyttäjä painanut 'Sisään'-nappia
ulos – saa arvon, jos rekisteröitynyt käyttäjä valitsee 'Kirjaudu ulos'-linkin
tunnus – käyttäjän antama käyttäjätunnus
salasana- käyttäjän antama salasana

Salasanan ja käyttäjätunnuksen tarkistuksen lisäksi tietokannasta haetaan henkilöt ja ryhmät valintalaatikoita varten (rivit 106,138). Lisäksi rekisteröityneen käyttäjän yhteydessä tarkistetaan tietokannasta omistaako henkilö yhtään ryhmää. Jos näin on, tulostetaan kaikki henkilön omistamat ryhmät valintalaatikkoon. Nyt käyttäjä voi valita oman ryhmänsä ja 'Mene'-napin painalluksella oikeaan kehykseen avautuu ryhmien hallintasivu OmatRyhmat-servletin toimesta..

LuoTunnus

Servletti luo käyttäjälle tunnuksen, jonka avulla hän pääsee käsiksi varausten tekoon yms. Jos tunnus, nimet tai salasanat puuttuvat tai ovat liian pitkiä generoidaan käyttäjälle virheilmoitus. Jos tiedot on täytetty oikein, tarkistetaan löytyykö tietokannan henkilö-taulusta ennestään riviä yritetyllä käyttäjätunnuksella. Jos yhtään riviä ei löydy, generoidaan 'Kiitos tunnuksen luonnista'-sivu

Parametrit:

tunnus – käyttäjän haluama tunnus
salasana – käyttäjän haluama salasana
varmistus – salasanan varmistus
sukunimi – käyttäjän sukunimi
etunimi – käyttäjän etunimi
sposti – käyttäjän sähköpostiosoite
puh – käyttäjän puhelinnumero

OmatTiedot

Servletti näyttää käyttäjälle hänen henkilötietonsa tekstikentissä ja antaa mahdollisuuden muuttaa niitä. Käyttäjän on mahdollista muuttaa tietoja suoraan tekstikenttiin. 'Tallenna tiedot'-nappi lähettää kenttien tiedot parametreina servletille itselleen, joka huomaa submit-parametrin saaneen arvon. Näin ollen servletti päivittää UPDATE-lauseella henkilön tiedot tietokantaan ja antaa generoi kiitossivun (rivit 128, 147).

Parametrit:

submit – nähdään onko käyttäjä painanut 'Tallenna tiedot'-nappia
etunimi – käyttäjän etunimi
sukunimi – käyttäjän sukunimi
sposti – käyttäjän sähköpostiosoite
puh – käyttäjän puhelinnumero

LuoRyhma

Tämä servletti antaa käyttäjän perustaa ryhmän, johon voi kuulua järjestelmässä olevia henkilöitä. Servletti käynnistyy, kun rekisteröitynyt käyttäjä valitsee navigointipalkista 'Luo ryhmä'-linkin. Käyttäjälle näytetään HTML-lomake, jonka avulla hän voi antaa ryhmälle nimen ja valita ryhmän jäsenet kaikkien järjestelmään kuuluvien henkilöiden joukosta. Poikkeuksena admin-käyttäjä, joka tälläkään sivulla näy käyttäjille. Tietokannasta haetaan kaikki henkilöt lomaketta varten (rivi 76). Lomake lähetään samalle LuoRyhma-servletille, joka puuttuvien tietojen tapauksessa huomauttaa asiasta käyttäjälle (rivit 34, 54).

Kun käyttäjä on saanut tarpeelliset tiedot täytettyä, päästään luomaan ryhmää. Ensimmäiseksi tarkistetaan kannasta, löytyykö sieltä halutun nimistä ryhmää (rivi 132). Jos kaikki on kunnossa luodaan *insertRyhmä*-lauseella haluttu ryhmä tietokantaan ja lisätään sen jälkeen jäsenyys tauluun yhteys valittujen henkilöiden ja luodun ryhmän välille *insertJäsenyys*-lauseella (rivit 144, 154). Käyttäjälle annetaan viesti onnistuneesta ryhmän luonnista, mutta huomautetaan, että ryhmää ei vielä näy vasemman reunan navigointipalkissa. Navigointipalkkia on mielestäni mahdotonta päivittää toisesta kehyksestä käsin. Ongelmana on lähinnä POST-muuttujan sisältö. JavaScriptillä voisi kyllä käskä `top.navi.location.reload()`, mutta tällöin selain kysyy lähetetäänkö POSTDATA uudestaan, joka ei ole kovin käyttäjäystävällistä. Näin ollen käyttäjä näkee luomansa ryhmän vasta seuraavalla sisäänkirjautumisella.

Parametrit:

submit – nähdään onko painettu 'Luo ryhmä'-nappia
jäsenet – String-taulu, jossa kaikki valitut ryhmän jäsenet
ryhmä – haluttu ryhmän nimi

OmatRyhmat

Tämän servletin tuottaman sivun avulla rekisteröityneellä käyttäjällä on mahdollisuus hallita luomiaan ryhmiä. Käyttäjällä on mahdollista poistaa tai lisätä jäseniä ryhmäänsä tai poistaa koko ryhmä. Servlettiin päästään, jos käyttäjällä on luotuja ryhmiä. Tällöin navigointipalkissa on luetteloruutu, josta käyttäjä voi valita ryhmänsä ja painaa 'Mene'-nappia. Parametrina saadaan ryhmän nimi, ja sen perusteella haetaan tietokannasta kaikkien ryhmään kuuluvien henkilöiden tiedot (rivi 250). Samoin haetaan ryhmään kuulumattomien henkilöiden tiedot sisäkkäisillä SELECT-lauseilla (rivi 304).

Käyttöliittymä jäsenten poistoon ja lisäämiseen on hyvin intuitiivinen. Ryhmään kuulumattomien henkilöiden listasta voi valita yhden tai useamman henkilön ja painaa vasemmalle osoittavia nuolia esittävää '<<'-nappia. Tiedot valituista lähetetään samalle servletille, joka huomaa, että submitLisays-muuttuja on saanut arvon (rivi 47). If-haarassa tallennetaan *insertJäsenyys*-lauseella uusien jäsenyyksien tiedot kantaan, ja sen jälkeen generoidaan loput sivusta, eli haetaan jälleen ryhmän jäsenet ja ei-jäsenet tietokannasta ja näin ollen juuri lisätyt jäsenet näkyvät jäsenten joukossa kuten tarkoitus oli. Jäsenten poisto ryhmästä toimii täysin samalla logiikalla (rivi 87). Sekä jäsenten poistossa että lisäyksessä kuljetetaan piilokenttänä mukana ryhmän nimeä.

Koko ryhmän poisto onnistuu myös. 'Poista ryhmä'-napin painallus aiheuttaa submitPoistaRyhma-muuttujan arvon muuttumisen, joten servletti joutuu asianmukaiseen if-haaraan (rivi 127). Tällöin joudutaan tietokantataulujen viite-eheyden säilyttämiseksi poistamaan ensin jäsenyys-tilusta kaikki kyseiseen ryhmään kuuluvat rivi ja vasta sen jälkeen on mahdollista poistaa ryhmä-tilusta ryhmän rivi (rivit 146,155). Ryhmän poistamisen yhteydessä esiintyy sama ongelma, joka mainittiin jo LuoRyhma-servletin kuvauksessa. Käyttäjä joutuu kirjautumaan ulos ja takaisin sisään, jotta ryhmän poistuminen näkyy navigointipalkissa.

Parametrit:

submitLisays – nähdään, onko lisäysnappia painettu
submitPoisto – nähdään, onko poistonappia painettu
submitPoistaRyhma – nähdään onko 'Poista ryhmä'-nappia painettu
jasenet – ryhmän jäsenet String-tauluna
henkilot – kaikki muut henkilöt, jotka eivät kuulu ryhmään String-tauluna
ryhmä – käsittelyssä oleva ryhmä
omatryhmät – parametri, joka saadaan Navi-servletiltä. tarkoittaa käsittelyssä olevaa ryhmää

RyhmanAjat

RyhmanAjat-servletti toteuttaa yhden järjestelmän keskeisistä toiminnoista, ryhmän kaikkien jäsenten yhteisen vapaan ajan laskemisen ja tulostuksen. Servletti käynnistyy, kun OmatRyhmat-servletin generoimalta ryhmien hallinta -sivulta painetaan 'Ryhmän ajat'-nappia. Napin painalluksen mukana lähetetään parametrina ryhmän nimi.

Servletin toteutuksessa etenkin kalenterin päivien laskemisen osalta on käytetty osittain samaa koodia kuin Kalenteri-servletissä. Ensin alustetaan kalenteri osoittamaan kuluva viikon

maanantaihin. Sitten luodaan kalenterin navigointinuolet. Sen jälkeen tulostetaan käyttäjälle solujen värit selityksineen. Musta on varattu vähintään yhdellä ryhmän jäsenellä ja vaalean harmahtavat ovat vapaita kaikilla jäsenillä. Seuraavaksi päästäänkin koko vapaiden aikojen laskemisen toteutuksen oleellisimpaan osaan. Luodaan 48 x 7 kokoinen boolean-*taulukko* (rivi 163), johon tallennetaan ryhmän jäsenten varaukset. Haetaan tietokannasta ryhmän jäsenet, käydään jokainen jäsen läpi. Jokaista jäsentä kohti käydään läpi viikon seitsemän päivää ja jokaista päivää kohti käydään läpi käsittelyssä olevan henkilön varaukset sille päivälle. Taas joudutaan muuntelemaan varausten aikoja laskettavaan muotoon. Puolituntia on yksi rivi taulukossa. Näin olleen taulukon indekset saadaan suoraan kertomalla tunnit kahdella ja jos ajassa on puolta tuntia merkitsevä osa, niin lisätään lukuun yksi. Nyt voidaankin käydä suoraan taulukon indeksejä läpi *for*-lauseella. Muutetaan *for*-lauseessa taulukon alkio *true*ksi varauksen kattamilla indekseillä.

Kun kaikki viikon varaukset kaikille ryhmän jäsenille on saatu laskettua, tulostetaan tiedot kalenterimaisessa muodossa (rivit 300-390). Erotuksena tavalliseen kalenteriin, on vapaiden aikojen kalenterin viivojen väli vain puoli tuntia. Tämä johtuu siitä, että taulukko on helppo käydä läpi rivi riviltä ja tulostaa aina musta solu kalenteriin, kun taulukon alkio on *true*. *True* ei siis tässä tapauksessa tarkoita, että vapaa aika on *true*, vaan varaus tietyllä indeksillä on *true*.

Parametrit:

omaryhmat – käsiteltävän ryhmän nimi

Kalenteri

Kalenteri-servleti hoitaa valitun henkilön tai ryhmän viikkokalenterin näyttämisen. Käyttäjällä on myös mahdollisuus poistaa tekemiään varauksia tämän servletin avulla. Viikkokalenteria voi siirtää päivän eteen- tai taaksepäin '>' ja '<' -napeilla ja viikon eteen- tai taaksepäin '>>' ja '<<' -napeilla. Kalenterin ensimmäisen päivän määrittelee *alkupvm*-parametri, jos sitä ei ole, näytetään kalenteri kuluvaan viikon maanantaista eteenpäin.

Valittuaan navigointipalkista ryhmän tai henkilön ja painettuaan 'Katso'-nappia, lähetetään Kalenteri-servletille joko ryhmän tai henkilön nimi parametrina. Valinnasta riippuen tulostetaan sivun alkuun joko henkilön tiedot tai ryhmän jäsenet. Servleti laskee sen jälkeen *alkupvm*:stä riippuvat *viikonAlku* ja *viikonLoppu* -muuttujat ja päivämäärät kalenterin liikuttamiseen tarvittavien nuolien linkkejä varten (rivi 220). Päivien otsikoiden tulostuksen jälkeen alkaa varsinainen varausten käsittely.

Servleti hakee tietokannasta henkilön tai ryhmän koko käsiteltävän viikon aikaisimman varauksen alkuajan (rivi 370). Tätä käytetään hyväksi, jos halutaan trimmata turhat tyhjät pois aamupäivistä. Tällä hetkellä ominaisuus ei ole käytössä tavallisella käyttäjällä, koska mielestäni kalenterin pituuden hyppiminen edestakaisin kalenteria selattaessa häiritsee käytettävyyttä. Kuitenkin *admin*-käyttäjällä trimmaus on asetettu päälle testitarkoituksessa. Varauksia tulostettaessa servleti käy läpi viikon seitsemän päivää ja hakee jokaisen päivän kohdalla halutun henkilön tai ryhmän varaukset. Varausten alkuaikoja ja kestoja joutui hieman muuntelemaan muodosta toiseen, koska varauksen alkuaika tietokannassa on *SQL*:n *time*-muotoa ja kesto *numeric(3,1)*. Lisäksi olen servleteissani hakenut kaikki kentät tietokannasta *String*-muotoisina. Tästä johtuen joutui pyörittelemään *substring*-metodia ja *wrapperi*-luokkia, jotta tiedot saatiin aritmeettisten operaatioiden vaatimaan muotoon. Laskutoimituksia tarvitaan kun lasketaan kuinka paljon tyhjää tilaa jätetään ennen päivän ensimmäistä varausta ja saman päivän varausten väliin (rivi 570). Tyhjä tila päivän ensimmäiselle varaukselle on suoraan varauksen alkuaika kerrottuna tunnin korkeudella pikseleissä. Näin saadaan pikseliluku, jonka avulla voidaan luoda oikean korkuinen taulukon solu. Saman päivän muille varauksille lasketaan tyhjä tila siten, että on laskettu

ja otettu talteen edellisen varauksen loppuaika, jolloin seuraavien varausten tyhjän tilan määrä saadaan vähentämällä varauksen alkuaikasta edellisen varauksen loppuaika. Se voi olla myös nolla, jolloin varaukset tulevat kiinni toisiinsa. Varausta tehdessä Teevaraus-servietti tarkistaa ettei kahta varausta voi laittaa päällekkäin, joten sitä ei tarvitse ottaa huomioon varauksia tulostettaessa. Varausta esittävää punertavaa solua ennen tulostetaan yhden pikselin korkuinen mustalla taustavärillä varustettu solu, jos varaus ei ala kello 00:00, koska tällöin koko taulukon omat reunat aiheuttaisivat tuplapaksuisen viivan. Sama tehdään varauksen jälkeen, jos varaus ei lopu kello 24:00. Itse varauksen tietoja varten pyöritellään taas aikastringejä, jotta saadaan esitys varauksen alku- ja loppuajoille. Jos varaus kestää vähemmän kuin kaksi tuntia, niin varauksen tekijää ei tulosteta tilan puutteen vuoksi (rivi 658). Tälläkin rajoituksella tunnin tai puolen tunnin varaukset näkyvät kalenterissa liian korkeina, mutta varausaika lukee kuitenkin varauksessa, joten sekaannusta ei tule. Tosin jos koko päivän täyttää esim. puolen tunnin varauksilla, niin kalenteri venyy alas päin ja tällöin vasemmassa laidassa olevat ajat eivät enää täsmää varauksien kanssa.

Varauksien sisään tulee pieni valintalaatikko, jos sisäänkirjautunut käyttäjä on varauksen tekijä (rivi 617). Tällöin käyttäjä voi valita näitä varauksia ja halutessaan poistaa ne painamalla kalenterin yläpuolella olevaa 'Poista valitut'-nappia. Tällöin valittujen varausten id-numerot lähetetään parametreina samalle Kalenteri-servletille, joka huomaa poista-parametrin saaneen arvon ja käy läpi haaran, jossa varaukset poistetaan (rivit 157,415). Yksittäisen henkilön kohdalla varausten poisto tapahtuu yksinkertaisella DELETE-lauseella, joka poistaa kaikki rivit varaus-taulusta parametrina saamiensa id-numeroiden perusteella. Tässä ei ole mitään virhemahdollisuutta. Ryhmän varausten kohdalla joudutaan toimimaan eri tavoin. Varausten tallentaminen tietokantaan on kuvattu tarkemmin TeeVaraus-servletin yhteydessä. Poistossa saadaan samalla tavalla kuin henkilöidenkin kohdalla parametrina poistettavien varausten id-numerot. Jos varauksia useita, käydään läpi yksi kerrallaan. Jokaisen varauksen kohdalla haetaan tietokannasta varauksen kaikki tiedot. Tämän jälkeen poistetaan varaus-taulusta kaikki varaukset, joiden tiedot ovat täysin samat kuin juuri noudetun ryhmän varauksen ja kuuluvat ryhmän jäsenille (rivi 435). Teoriassa tietokannassa voisi olla kaksi eri varausta täysin samoilla tiedoilla, mutta käytännössä TeeVaraus-servletti huolehtii siitä, että näin ei voi käydä.

Kalenteri-servletti käyttää hyväksi muutamissa kohdissa tekemääni apuluokkaa CheckDate, joka tekee pieniä tarkistuksia päivämääristä.

Viikkokalenterin näkymästä on myös linkki kyseisen henkilön koko vuoden kalenteriin. Linkkiä painettaessa lähetetään tieto henkilöstä ja viikon ensimmäisen päivän mukaisesta vuodesta VuosiKalenteri-servletille, joka generoi henkilön koko vuoden kalenterin.

Kuvat:

kale_bg.gif – 1 x 26 taustakuva esittää kalenterin tuntiviivoja jokaisen päivän solussa
transparent.gif – 1 x 1 läpinäkyvä taustakuva tyhjää tilaa esittävälle taulukon soluille, jotta tuntiviivat näkyvät läpi

Parametrit:

valitutTaulu – tähän tallennettu valitut varaukset, jotka poistetaan
poista – nähdään onko painettu 'Poista valitut'-nappia
henkilo – henkilö, jonka kalenteria ollaan katsomassa
ryhma – ryhmä, jonka kalenteria ollaan katsomassa
alkupvm – päivämäärä, josta kalenterin näkymä alkaa, tarvitaan kalenterin ohjausnuolissa
viikkoAlusta – näyttää kalenterin alkupvm:n sisältämän viikon alusta, tarvitaan vuosikalenterin kautta tultaessa

KalenteriYleinen

Tämä servletti on suureksi osaksi kopio Kalenteri-servletistä. KalenteriYleinen generoi viikkokalenterin rekisteröitymättömille käyttäjille. Kalenteri on muuten samanlainen kuin edellisessä servletissä esitelty, mutta varauksista ei näy minkäänlaisia tietoja (rivi 395). Rekisteröitymätön käyttäjä näkee ainoastaan, että tietyt ajat ovat varattuja. Vuosikalenteriin ei ole pääsyä tältä sivulta. Varausten poisto ei ole myöskään mahdollista.

Parametrit:

henkilo – henkilö, jonka kalenteria katsellaan
ryhmä – ryhmä, jonka kalenteria katsellaan
alkupvm – päivä, josta lähtien viikkokalenteri näytetään

VuosiKalenteri

Vuosikalenteri on ominaisuus, jonka keksin toteuttaa vasta viimeisenä päivänä. VuosiKalenteri-servlettiin pääsee klikkaamalla Kalenteri-servletin luoman kalenterin yläpuolella olevaa linkkiä 'Koko vuoden kalenteri'. Linkissä on mukana kolme lähetettävää parametria. Joko henkilö tai ryhmä, jolle vuoden kalenteri generoidaan ja generoitava vuosi.

Aluksi alustetaan Calendar-olio osoittamaan valutun vuoden ensimmäiseen päivään. Sen jälkeen lähdetään tulostamaan vuoden kalenteria. Käydään for-lauseessa läpi kaksitoista kuukautta yksi kerrallaan. Ensin tulostetaan kuukauden nimi ja rivi viikonpäivien lyhennyksille. Sen jälkeen tarkistetaan osuuko kuukauden ensimmäinen päivä maanantaille. Jos ei, niin tulostetaan tyhjiä soluja, kunnes on ensimmäisen päivän vuoro. Tulostellaan kuukauden päiviä ja aina sunnuntain kohdalla vaihdetaan riviä taulukossa. Kuukauden lopussa tarkistetaan vuorostaan osuuko kuukauden viimeinen päivä sunnuntaille. Jos ei, tulostetaan viimeinen viikko loppuun tyhjillä soluilla. Näin on yksi kuukausi tulostettu.

Jokaisen tulostettavan päivän kohdalla tarkistetaan löytyykö tietokannasta yhtään varausta kyseiselle henkilölle tai ryhmälle sinä päivänä (rivi 165-190). Jos löytyy yksikin varaus päivälle, tulostetaan taustaväriiltään punertava solu ja laitetaan päivän numeron ympärille linkki, joka osoittaa takaisin Kalenteri-servlettiin. Linkin parametreilla kerrotaan Kalenteri-servletille, että tarkoitus olisi näyttää parametrina annetun päivämäärän sisältävä viikko alusta lähtien.

Kuukausien tulostus allekkain ei ole paras mahdollinen, mutta tämän servletin ollessa viimeisimpiä toteutettavia toimintoja, voimat eivät enää riittäneet esim. 4 x 3 tapaisen tulostuksen tekemiseen.

Parametrit:

henkilo – henkilö, jonka vuosikalenteria generoidaan
ryhmä – ryhmä, jonka vuosikalenteria generoidaan
vuosi – vuosi, jonka perusteella vuosikalenteri generoidaan

TeeVaraus

Servletti hoitaa varausten teon. Rekisteröityneen käyttäjän valitessa 'Tee varaus'-linkin navigointipalkista, avataan oikeaan kehykseen varauksen tekoon tarkoitettu lomake. Lomakkeessa on kentät varauksen aiheelle, päivämäärälle, alkuajalle, kestolle, näkyvyydelle ja kenelle varaus tehdään. Varauksen voi tehdä joko henkilölle tai ryhmälle.

Sivulla käytetään JavaScript-tarkistusta, joka on määritelty tiedostossa tarkista.js. Kun käyttäjä lähettää lomakkeen, napataan tämä tapahtuma onSubmit-tapahtumakuuntelijalla. Kaikki lomakkeen kentät tarkistetaan, ja jos löytyy tyhjiä kenttiä, tästä mainitaan käyttäjälle. Samoin tarkistetaan, että käyttäjä ei voi valita yhtä aikaa ryhmää ja henkilöä. Ajattelin, että tämä JavaScript-tarkistus on tässä tilanteessa paras vaihtoehto. Parametrit kyllä tarkistetaan myös servletin toimesta, jos esim. JavaScript ei ole päällä. JavaScriptin etu tulee esille, jos käyttäjä on jo osin täyttänyt lomakkeen ja lähettää sen. Tällöin jo täytetyt kentät jäävät paikoilleen ja annetaan vain alert-laatikkoon virheilmoitus. Näin olleen käyttäjän ei tarvitse aina virheen yhteydessä täyttää koko lomaketta uudestaan.

Tietojen täyttöön tarkoitetun HTML-lomakkeen generoinnissa tehdään kolme tietokantahakua. Yhdellä haetaan kaikki henkilöt tietokannasta, toisella kaikki ryhmät ja kolmannella näkyvyyden eri vaihtoehdot. Kuten muissakin henkilövalintalaatikoissa, tässäkin ei tulosteta admin-käyttäjää listaan mukaan.

Kun lomake lähetetään oikein täytettynä, välitetään kaikki tiedot parametreina samalle TeeVaraus-servletille. Tällöin servletti huomaa submit-muuttujan saaneen arvon ja siirtyy sen mukaisesti haaraan, jossa käsitellään varauksen tarkistus ja tallentaminen tietokantaan (rivi 302).

Ensimmäiseksi varauksen päivämäärä lähetetään tekemäni apuluokan CheckDate metodin checkDate läpi. Metodi korjaa kuukauden maksimipäivän oikeaksi, eli varausta ei voi tehdä esim. päivälle 31.2.2002. Metodissani on otettu huomioon myös karkausvuodet. Tietokantaan olisi mahdollista tallettaa varaus päivämäärälle 31.2.2002, mutta tällöin varaus merkittäisiin tosiasiasa päivämäärälle 3.3.2002. Ajattelin että raaka muuttaminen alaspäin on paras vaihtoehto, koska henkilön halutessa tehdä varaus vaikka huhtikuun 31. päivälle (huhtikuussa on 30 päivää), hän luultavasti ajattelee vain tekevänsä varauksen kuukauden viimeiselle päivälle. Näin ollen metodi korjaa päivämäärän käytännössä oikeaksi.

Seuraavaksi tarkistetaan meneekö yritetty varaus jo kalenterissa olevan varauksen päälle (rivit 334-584). Menetelmä on käytännössä sama niin henkilön varaukselle kuin ryhmän varauksellekin. Ryhmän varausta käsiteltäessä haetaan ainoastaan ensin kaikki ryhmän jäsenet talteen, jotta voidaan hakea yritetyn varauksen päivämäärän mukaiset vanhat varaukset kaikille ryhmän jäsenille tietokannasta (rivi 492). Jos kyseessä on yhdelle henkilölle tehtävä varaus, niin tietokannasta haetaan kaikki henkilön varaukset yritetyn varauksen päivälle. Sitten taas pyritellään saatuja tietoja sen verran, että voidaan tehdä laskuoperaatioita varausten ajoilla. Käytännössä kaikki ajat muutetaan doubleiksi, koska kestot ja aloitusajat saattavat olla myös puolikkaita tunteja. Tarkistetaan myös ylittääkö haluttu varaus vuorokausirajan. Jos näin on tallennetaan virheilmoitus myöhempää käyttöä varten. Virheilmoituksessa kerrotaan, että ylityksestä huolimatta varaus on tehty varauspäivän loppuun asti, mutta kehoitetaan käyttäjää tekemään uusi varaus seuraavasta aamusta lähtien, jos hän haluaa varauksen jatkuvan. Jokaisen kalenterista varauspäivälle löytyvän varauksen kohdalla tarkistetaan menevätkö yritetty varaus ja kalenterissa jo oleva varaus päällekkäin. Päällekkäisyyksiä on kolmea lajia. Yritetty varaus voi alkaa ennen kalenterissa olevaa varausta, mutta jatkuu tämän päälle. Varaukset voivat alkaa yhtä aikaa. Yritetty varaus alkaa keskellä vanhaa varausta. Kaikille kolmelle tapaukselle on omat virheilmoituksensa. Jos yritetty varaus osuu johonkin näistä kolmesta vaihtoehdosta, generoidaan käyttäjälle sivu, jossa kerrotaan virheestä, ja näytetään visuaalisesti näiden kahden varauksen suhde.

Jos päällekkäisyyksiä ei löydy, tallennetaan lomakkeesta saadut tiedot INSERT-lauseella tietokantaan (rivi 387,599). Ryhmän varauksen kohdalla tallennetaan varaus jokaisen ryhmän jäsenen sekä ryhmän itsensä kalenteriin (rivi 605). Sen jälkeen generoidaan käyttäjälle viesti varauksen onnistumisesta. Varauksentekosivulla ei ole linkkejä, ainoastaan napit 'Varaa' ja 'Tyhjennä'.

Parametrit:

submit – nähdään onko painettu 'Varaa'-nappia
paiva – varauksen päivä
kuukausi – varauksen kuukausi
vuosi – varauksen kuukausi
aihe – varauksen aihe
henkilo – henkilö, jolle varausta ollaan tekemässä
ryhma – ryhmä, jolle varausta ollaan tekemässä
alkuaika – varauksen alkuaika
kesto – varauksen kesto
nakyyvyys – varauksen näkyvyys

CheckDate

CheckDate on luokka, jossa on muutama apumetodi päivämäärien tarkistamiseen. Metodi `c(int)` palauttaa annetun kokonaisluvun Stringinä ja varustettuna etunollalla, jos luku on pienempi kuin kymmenen. Metodi `checkDate(String)` palauttaa muodossa DD-MM-YYYY saaneensa päivämäärämerkkijonon tarkistettuna takaisin. Käytännössä tarkistetaan, ettei päivien määrä ylitä kuukauden maksimipäivämäärää. Jos näin tapahtuu, vähennetään päivien määrää vastaamaan oikeaa tilannetta. Karkausvuodet otettu myös huomioon metodissa.

Asennustiedot

Tiedostojen sijainti

Servlettien .java tiedostot sijaitsevat kotihakemistoni `jserv/servlets` alihakemistossa. HTML-, JavaScript- ja CSS-tiedostot sijaitsevat `public_html`-hakemistoni alla osoitteessa `public_html/tsoha`. Kaikki osoitteet ovat tietenkin suhteellisia kotihakemistooni nähden `db.cs.helsinki.fi-koneella`.

Käynnistysohje

Tarvittavat osoitteet

Yhteisen kalenterijärjestelmään pääsee osoitteesta <http://db.cs.helsinki.fi/~avtanska/tsoha/>

Servletit ovat ohjeiden mukaan päällä kymmenen vuorokautta palautusajankohdasta lähtien. Pohjalla olevan Postgres-tietokannan automaattinen pysäytys näyttää olevan pois päältä, joten jätän sen päälle myös samaisiksi kymmeneksi päiväksi. Jos tietokanta ei jostain syystä kuitenkaan ole päällä, voi sen käynnistää täältä: <http://db.cs.helsinki.fi/~avtanska/dmt/start.php>