

Suunnitteludokumentti

Aija

Helsinki 1.9.2005

Ohjelmistotuotantoprojekti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

Sisältö

1	Uusia tarkennuksia käyttäjävaatimukseen	1
1.1	Yleistä	1
1.1.1	Järjestelmän nimen paikka etusivulla	1
1.1.2	Järjestelmän nimen paikka muilla sivuilla	1
1.1.3	Järjestelmän nimi valikkopalkissa	1
1.1.4	Käyttöliittymän kielivalinnasta ilmoittaminen	1
1.1.5	Käyttöliittymän kielen valitseminen	1
1.1.6	Kielivalinnan näkyminen	1
1.1.7	Kurki-järjestelmän tuomia vaatimuksia	1
1.1.8	Tapaamisaikapainikkeet	2
1.1.9	Ilmoitusten taustaväri	2
1.1.10	"Asetukset" ja "Kirjaudu ulos" paikka	2
1.1.11	Nimien asemointi	2
1.1.12	Värien vaihdettavuus	2
1.1.13	Tekstien värien vaihdettavuus	2
1.1.14	Tekstikenttien leveys	2
1.1.15	Tallenna-napit	2
1.1.16	Poista-nappi	2
1.2	Ylläpito	3
1.2.1	Ylläpitäjän sähköpostiosoite	3
1.2.2	Taustatoimintojen vakavat virhetilanteet	3
1.2.3	Taustatoimintojen muut virhetilanteet	3
1.2.4	Taustatoimintojen toistuvat virhetilanteet	3
1.2.5	Järjestelmää ylläpitää- tunnus	3
1.3	Sopivuudet	3
1.3.1	Sopivuuksien järjestys	3
1.3.2	Sopivuuksien Kesken-tekstit	3
1.3.3	"Ei-tiedossa- sopivuus	3
1.3.4	Viikonpäivälyhenteiden merkitseminen	4
1.3.5	Sopivuuksien värit	4
1.3.6	Sopivuuksien tekstit	4
1.3.7	Laskuri	4

		11
	1.3.8 Sopivuuksien viikkonäytön päivät	4
1.4	Järjestelynäyttö	4
	1.4.1 Valikkopalkin tekstin sanamuoto	4
	1.4.2 Järjestä- painike	4
	1.4.3 Varauksen siirtäminen	4
	1.4.4 Aikataulun järjestelytilan ilmoittaminen käyttäjälle	5
	1.4.5 Järjestelynäytön hyppytuntisarakkeiden koko	5
1.5	Kalenteri	5
	1.5.1 Yleisilmeen keventäminen	5
	1.5.2 Kalenterin selausnuolten sijainti	5
	1.5.3 Kalenterin selausnuolten selailukoko	5
	1.5.4 Kalenterin koko	5
	1.5.5 Kalenterin ja viikkonäytön synkronointi	5
1.6	Algoritmi	6
	1.6.1 Algoritmin toimintaan vaikuttavan rastin käyttö	6
	1.6.2 Järjestelynäytön muodostumiselle asetettuja vaatimuksia	6
	1.6.3 Algoritmin toiminta	6
1.7	Aikataulun luonti	6
	1.7.1 Tallenna- painikkeen käyttö	6
	1.7.2 Jakovälin valinta	6
1.8	Viestit	6
	1.8.1 Käyttäjälmoitus	6
	1.8.2 Sopivuussivun linkki	6
	1.8.3 Viestisivujen rakenne	7
	1.8.4 Kaikkien kutsunsaajien valinta	7
	1.8.5 Samanimisten erottaminen	7
	1.8.6 Viestinmuokkauskentän koko	7
	1.8.7 Tägien lisääminen	7
	1.8.8 Tapaamisajasta ilmoittaminen	7
	1.8.9 Uusien ryhmien viestien pohjat	7
	1.8.10 Järjestelmän kutsun/ilmoituksen oletuspohjan käyttö	7
2	Järjestelmän yleiskuvaus	8
	2.1 Sovellusalue	8

2.2	Sopimukset ja standardit	8
2.3	Laitteistoympäristö	9
2.4	Toteutuksen keskeiset reunaehdot	10
2.4.1	Tietoturva	10
2.4.2	Sekvenssin käyttö	10
2.4.3	Viestit	10
2.4.4	Käyttäjätiedot	10
3	Arkkitehtuuri	11
3.1	Osajärjestelmät	11
3.1.1	Kommunikointi	12
3.1.2	Yleiset palvelut	14
3.1.3	Syötteet	14
3.1.4	Virhekäsittely sovelluslogiikassa	15
3.1.5	Aikataulu - timetable.php	15
3.1.6	Aikataulun sovelluslogiikka - offer.php	22
3.1.7	Ajastettu toiminta - backrun.php	24
3.1.8	Asetukset - profile.php	28
3.1.9	Etusivu - index.php	30
3.1.10	Järjestelynäyttö	35
3.1.11	Osallistujan viikkonäyttö - match.php	38
3.1.12	Osallistujan sovelluslogiikka - prefer.php	41
3.1.13	Osallistuja - member.php	44
3.1.14	Ryhmä - group.php	46
3.1.15	Uloskirjautuminen - sign_out.php	48
3.1.16	Viesti - message.php	49
3.2	Tietokanta-arkkitehtuuri	53
3.3	Viite-eheys	53
3.4	Tietokantakuvaus	53
3.4.1	Calendars - taulu	54
3.4.2	Calendar_History-taulu	55
3.4.3	Codes -taulu	56
3.4.4	Groups -taulu	56
3.4.5	Locales -taulu	57

3.4.6	Matches -taulu	57
3.4.7	Messages-taulu	58
3.4.8	Offers-taulu	59
3.4.9	Users -taulu	59
3.4.10	Values -taulu	60
3.4.11	Weeks-taulu	61
3.5	Tiedostokuvaus	61
3.5.1	Ilmoitus-tiedosto	63
3.5.2	Kutsu-tiedosto	63
3.5.3	Rekisteröityminen-tiedosto	63
3.5.4	Salasana-tiedosto	64
3.5.5	Valmius-tiedosto	64
4	Komponenttien sijoitusjärjestys	64
5	Alustuskomponentti	65
5.1	Yleiskuvaus sovelluslogiikasta	65
5.2	initialize.php-komponentin käyttö	65
5.3	Vakiot	66
5.4	Alustustoimet	66
5.5	Palvelut	67
5.5.1	getCodeDefault	67
5.5.2	getCodeValue	67
5.5.3	getGUID	68
5.5.4	getParameterInteger	68
5.5.5	getParameterString	68
5.5.6	getPreviousPage	69
5.5.7	getQueryString	69
5.5.8	getUserRole	70
5.5.9	relocate	70
5.5.10	savePreviousPage	70
5.5.11	threeDots	71
6	Tietokantakomponentti	71

6.1	db.php-komponentin käyttö	71
6.2	Virhekäsittely	71
6.3	Tietokannan luonti	72
6.4	Vakiot	75
6.5	Palvelut	76
6.5.1	dbBegin	77
6.5.2	dbCommit	77
6.5.3	dbRollback	77
6.5.4	dbClose	77
6.5.5	dbConcatenate	77
6.5.6	dbConnect	78
6.5.7	dbErrCode	78
6.5.8	dbErrText	78
6.5.9	dbExecute	78
6.5.10	dbFree	79
6.5.11	dbGetID	79
6.5.12	dbGetRow	79
6.5.13	dbNumRows	79
6.5.14	dbOpen	80
6.5.15	dbQuery	80
7	Terminointikomponentti	80
7.1	terminate.php-komponentin käyttö	80
8	Monikielisyyden tuen komponentti	80
8.1	i18n.php-komponentin käyttö	81
8.2	Kielikoodi	81
8.3	Tekstin osoitus	81
8.4	Parametrointi	81
8.5	Tekstien ylläpito	82
8.6	Palvelut	82
8.6.1	defaultLanguage	82
8.6.2	languageDrop	83
8.6.3	languageList	83

8.6.4	localText	83
9	Kalenterikomponentti	84
9.1	calendar.php -komponentin käyttö	84
9.2	Palvelut	84
10	Reunalistakomponentti	86
10.1	sidepane.php-komponentin käyttö	86
10.2	Palvelut	86
10.2.1	activeLinkTo(\$userID)	86
10.2.2	sidePane(\$id, \$lang, \$itemID, \$parentID)	86
11	Järjestelmän esittely komponentti	87
12	Sähköpostikomponentti	88
12.1	email.php-komponentin käyttö	88
12.2	Palvelut	88
12.2.1	getFile	88
12.2.2	invalidEmail	88
12.2.3	sendMail	89
12.2.4	wrapAddress	90
13	Viestityskomponentti	90
13.1	message.php-komponentin käyttö	90
13.2	Palvelut	91
13.2.1	Viestit	91
13.2.2	Kehotteet	91
13.2.3	Poiston vahvistus	92
14	CSV jäsentäjä csv.php	92
14.1	Palvelut	92
14.2	Toiminnallisuus	93
15	Aikojenjakoalgoritmi matchingAlgorithm.php	93
15.1	Palvelut	94
15.2	Toiminnallisuus	94

15.3	Tietokantakyselyt	95
16	Unkarilaisen metodin toiminta	95
16.1	Unkarilainen metodin suoritus. Alkutoimet	96
16.2	Unkarilainen metodin suoritus. Toiminta pääpiirteittäin.	96
16.2.1	Unkarilaisen metodissa käytettävät tietorakenteet	98
16.2.2	hajonnan laskeminen	100
17	Käyttöliittymäkuvaus	100
17.1	Yleiskuvaus	100
17.1.1	Sivun metatiedot	100
17.1.2	HTML:n valmiiden elementtien käyttäminen	101
17.1.3	Komponenttien asemointi käyttöliittymässä	101
17.1.4	CSS-tyylimäärittelyjen käyttö	101
18	Testaussuunnitelma	103
18.1	Johdanto	103
18.2	Testauksen kohde ja tavoitteet	103
18.3	Testausympäristö	104
18.4	Testausstrategia	104
18.4.1	Ohjelmakoodissa havaitut puutteellisuudet ja virheellinen toiminnallisuus	105
18.5	Testauksen organisointi ja raportointi	105
18.6	Testausvaiheet	106
18.7	Yksikkötestaus	107
18.7.1	Integroititestausta	107
18.7.2	Komponenttien yhdistäminen	108
18.8	Testattavat rajapintafunktiot	108
18.8.1	Validointitestausta	108
18.9	Järjestelmätestaus	109
18.9.1	Testausmenetelmät	109
18.9.2	Testauksen resurssointi	111
18.9.3	Muut testattavat alueet	111

1 Uusia tarkennuksia käyttäjävaatimukseen

Nämä vaatimukset ovat tulleet esille määrittelyvaiheen jälkeen, käyttöliittymäprototyyppiä tehtäessä. Ne on kerätty asiakkaan lähettämistä sähköposteista, jotka ovat olleet hänen kommenttejaan hänelle esitellyistä ratkaisumalleista.

1.1 Yleistä

1.1.1 Järjestelmän nimen paikka etusivulla

Järjestelmän nimi, Aija - aikojenjakojärjestelmä, etusivulla isommalla. Etusivulla nimen ei tarvitse olla valikkopalkissa

1.1.2 Järjestelmän nimen paikka muilla sivuilla

Järjestelmän nimi yläpalkista valikkopalkin ensimmäiseksi kohdaksi.

1.1.3 Järjestelmän nimi valikkopalkissa

Valikkopalkissa ollessaan nimen tulisi näyttää erilaiselta kuin valikkopalkin varsinainen valikko, jotta nimen käsittäisi otsikon kaltaiseksi.

1.1.4 Käyttöliittymän kielivalinnasta ilmoittaminen

Käytetään muotoa "suomeksi". Suomen kielilinkki ensin.

1.1.5 Käyttöliittymän kielen valitseminen

"Käyttöliittymän kieli", jossa dropdown list. : suomi, english. Valittuna suomi. Sama valinta myös näyttöön, jolla näytetään osallistujasta tallennetut tiedot.

1.1.6 Kielivalinnan näkyminen

Kielivalinta ei ole valittavissa valikkopalkista.

1.1.7 Kurki-järjestelmän tuomia vaatimuksia

Kurki-järjestelmästä tulee pitkiä rivejä, niistä on saatava näkyviin suurin osa.

.

1.1.8 Tapaamisaikapainikkeet

Tapaamisaikapainikkeet samassa järjestyksessä kuin valikkopalkissakin.

1.1.9 Ilmoitusten taustaväri

Ilmoituksen taustaväri on muu kuin punainen.

1.1.10 "Asetukset" ja "Kirjaudu ulos" paikka

"Asetukset" ja "Kirjaudu ulos" ovat valikkopalkin viimeisiä kohtia, korostus kuten muissakin valikkopalkin kohdissa.

1.1.11 Nimien asemointi

Nimien tasaus vasemmalle.

1.1.12 Värien vaihdettavuus

Värit oltava helposti ylläpitäjän vaihdettavissa

1.1.13 Tekstien värien vaihdettavuus

On kuitenkin syytä määritellä kaikenlaisten tekstien väri tyylitiedostossa erikseen
20.6.5 Joni Salmi

1.1.14 Tekstikenttien leveys

Tekstikenttien leveys kaikkialla sellaiseksi, että se suurin piirtein vastaa valikkopalkkiin vastaavaan kohtaan yhdelle riville mahtuvan tekstin määrää.

1.1.15 Tallenna-napit

Jos mahdollista, Tallenna-napit kaikkialla siten, että ne ovat harmaannutettuja, kunnes tallennettavia muutoksia on tehty. 21.6.5 Joni Salmi

1.1.16 Poista-nappi

Vahvistusviestin Poista-painikkeen voisi nimetä Poista lopullisesti -painikkeeksi. 20.6.05
Joni Salmi

1.2 Ylläpito

1.2.1 Ylläpitäjän sähköpostiosoite

Ylläpitäjälle täytyy olla mahdollista määritellä sähköpostiosoite.

1.2.2 Taustatoimintojen vakavat virhetilanteet

Taustatoimintojen virhetilanteessa järjestelmä lähettää virheilmoituksen ylläpitäjän sähköpostiosoitteeseen.

1.2.3 Taustatoimintojen muut virhetilanteet

Ei ylläpitäjälle virheilmoituksia, jos virhetilanne on ohimenevä.

1.2.4 Taustatoimintojen toistuvat virhetilanteet

Kaikki samaan ajoon liittyvät virheet lähetetään yhdessä sähköpostissa. 17.6.5Joni Salmi

1.2.5 Järjestelmää ylläpitää- tunnus

Etusivulle "Järjestelmää ylläpitää "ohtu@cs.helsinki.fi". 20.06.05 Joni Salmi

1.3 Sopivuudet

1.3.1 Sopivuuksien järjestys

Järjestys sopivuuksille valikossa:"erinomaisesti", "hyvin", "välttävästi", "ei sovi", "ei tiedossa".

1.3.2 Sopivuuksien Kesken-tekstit

Sopivuuksien Kesken-tekstit (valikkopalkissa) tulisivat olla myös järjestäjän näky-
mässä, siis että järjestäjäkin näkisi yhdellä silmäyksellä valikkopalkista kenellä on
sopivuuksia syöttämättä.

20.6.5Joni Salmi

1.3.3 "Ei-tiedossa- sopivuus

"Ei-tiedossa-indikaattorina pelkkää valkoista väriä ilman kysymysmerkkejä.

1.3.4 Viikonpäivälyhenteiden merkitseminen

Viikonpäivälyhenteitä ei lihavoida missään.

1.3.5 Sopivuuksien värit

Sopivuuksien syöttönäytössä noudatetaan samaa väritystä kuin sopivuuksien järjestelynäytössä: valkoinen = ei tiedossa, tummanharmaa = ei sovi jne.

1.3.6 Sopivuuksien tekstit

Sopivuudet listakomponenttien asemesta tavallisella tekstillä. Kaikkien sopivuuksien tekstit ovat väriltään mustia. 21.6.5 Joni Salmi

1.3.7 Laskuri

Laskuri on muotoa "0 tarjottua aikaa, 0 osallistujaa".

22.6.5 Joni Salmi

1.3.8 Sopivuuksien viikkonäytön päivät

Sopivuuksien syöttönäytöissä ovat kaikki viikonpäivät. Tarkoitus leventää tyhjiäkin sarakkeita siten, että näyttöä voi verrata paperikalenteriin.

28.6.05 Joni Salmi

1.4 Järjestelynäyttö

1.4.1 Valikkopalkin tekstin sanamuoto

Valikkopalkissa käytetään termiä "Aikojen varaaminen".

1.4.2 Järjestä- painike

Järjestä-painiketta tarvitaan vain kun automaattisuusrasti ei ole päällä ja ajat eivät ole järjestyksessä. Muulloin Järjestä-painikkeen voi harmaantua. 20.6.05 Joni Salmi

1.4.3 Varauksen siirtäminen

Solun napsauttaminen siirtää varauksen aina. Taulukon yhteyteen ohjeteksti: "Voit siirtää varauksen napsauttamalla uutta ajankohtaa." 20.6.5 Joni Salmi

1.4.4 Aikataulun järjestelytilan ilmoittaminen käyttäjälle

Taulukon alla valintaruutu, jossa selite: "Sopivuudet muutettavissa, ajat varataan automaattisesti". Järjestelytilaa vastaa rastin puuttuminen. 21.6.5Joni Salmi

1.4.5 Järjestelynäytön hyppy tuntisarakkeiden koko

Järjestelynäytön aikojen hyppy tuntisarakkeen koko on suunnilleen yhtä leveä kuin välipäivien sarake. 22.6.5Joni Salmi

subsubsectionJärjestelynäytön viikkonäytön koko

La ja su ovat aina mukana, useimmiten tyhjinä kapeina harmaina sarakkeina. 27.6.05
Joni Salmi

1.5 Kalenteri

1.5.1 Yleisilmeen keventäminen

Jos ohuempia viivoja saa aikaan vain sisäkkäisillä taulukoilla, niin olkoot viivat nykyisellään. Vaakasuorat viivat pois kuukausinimi-vuosiluku-sarakkeesta. 22.6.5Joni Salmi

1.5.2 Kalenterin selausnuolten sijainti

Käsitteellisesti kalenterinauha "liikkuu" pystysuunnassa. Selausnuolten pitää olla ylös ja alas ja niiden pitää sijaita ylä- ja alareunassa.

1.5.3 Kalenterin selausnuolten selailukoko

Selausnuolet selatkoon neljä viikkoa kerrallaan.

1.5.4 Kalenterin koko

Kalenteria näytetään 22 viikkoa.

1.5.5 Kalenterin ja viikkonäytön synkronointi

Viikkonäytöllä on aina koko viikko näkyvissä, myös kalenterinauhalla viikot korostetaan aina kokonaan 28.6.5 Joni Salmi

1.6 Algoritmi

1.6.1 Algoritmin toimintaan vaikuttavan rastin käyttö

Oletusarvoisesti rasti on päällä. Varauksen siirtäminen manuaalisesti ottaa rastin pois päältä. Rastin kytkeminen takaisin päälle aiheuttaa varauksien palauttamisen algoritmin laskemaan tilaan. 20.6.5Joni Salmi

1.6.2 Järjestelynäytön muodostumiselle asetettuja vaatimuksia

Järjestä ei saa joutua odottamaan järjestelynäytön muodostumista kohtuuttoman pitkään.

1.6.3 Algoritmin toiminta

Algoritmin ei ehdottomasti tarvitse käynnistyä joka kerta kun kantaan tulee muutos. Algoritmi lähtisi käyntiin, kun järjestäjä kirjautuu sisään. 22.6.05 Joni Salmi

1.7 Aikataulun luonti

1.7.1 Tallenna- painikkeen käyttö

Tallenna-painikkeella tallennetaan vain aikataulun nimi. Tyypin ja jaon muutokset välittyvät heti muutosten jälkeen.

1.7.2 Jakovälin valinta

20 min on hyvä oletusarvo, viikkoruudukko on synkronissa jakovalinnan kanssa 20.6.05 Joni Salmi

1.8 Viestit

1.8.1 Käyttäjälmoitus

Ilmoitusteksti käyttäjälle: "Et voi muuttaa tarjottujen aikojen sopivuuksia, sillä tapaamisaikoja ollaan paraikaa varaamassa tai ne on jo varattu."

1.8.2 Sopivuussivun linkki

Elementeistä linkki sopivuussivulle on aina ensimmäinen painike.

1.8.3 Viestisivujen rakenne

Viestin vastaanottajat ja ei-vastaanottajat, käytetään kaksoislistaa: vastaanottajat ovat yhdessä listassa, ei-vastaanottajat toisessa listassa ja nämä listat ovat rinnakkain. Kummasta tahansa listasta voi valita yhden tai (vaihtonäppäintä hyödyntämällä) useampia henkilöitä ja painonapilla/napeilla siirtää ne toiseen listaan.

1.8.4 Kaikkien kutsunsaajien valinta

Siirrä kaikki -napeista voidaan kaikki kutsutu käsitellä kerralla. Sama toiminnallisuus saavutetaan vaihtonäppäintä hyödyntäen.

1.8.5 Samanimisten erottaminen

Nimien lisäksi on syytä ilmoittaa sähköpostiosoitteet (jos on) siltä varalta, että ryhmään sattuu kaksi samannimistä.

1.8.6 Viestinmuokkauskentän koko

Viestinmuokkauskentän olla riittävän suuri, jotta vierittämisen tarpeelta vältyttäisiin. Loppu tila joka jää komponenteilta sekä 13 osallistujalta annetaan viestinmuokkauskentälle (jonkin tyypillisen näyttökoon mukaan laskettuna).

1.8.7 Tägien lisääminen

Mahdollisuutta lisätä yksittäinen tägi kursorin kohdalle, vahingossa deletoitujen tilalle

1.8.8 Tapaamisajasta ilmoittaminen

Osallistuja saa tapaamisaikansa sähköpostiviestissä, mikäli hän on antanut sähköpostiosoitteensa..

1.8.9 Uusien ryhmien viestien pohjat

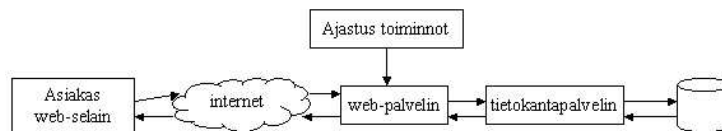
Uusien ryhmien kutsuviestin pohjaksi jokin käyttäjän muokkaamista kutsuviesteistä, jos sellainen on olemassa. Vastaavasti ilmoitusviestien kanssa.

1.8.10 Järjestelmän kutsun/ilmoituksen oletuspohjan käyttö

Järjestelmän oletuspohjaa kutsulle/ilmoitukselle käytetään vain, kun kutsua/ilmoitusta lähetettäessä ei ole olemassa yhtään muokattua kutsua/ilmoitusta, josta voisi kopioida. 16.6.05 Joni Salmi

2 Järjestelmän yleiskuvaus

Järjestelmän suunnittelussa on huomioitu ohjelman toteutus PHP-skripteillä. Valittu toteutustapa on proseduraalinen, jonka vuoksi dokumentoinnissa ei esitellä luokkamallia. Järjestelmä koostuu osajärjestelmistä sekä komponenteista, joita osajärjestelmät käyttävät. Kukin osajärjestelmä on toteutettu omana PHP-skriptisivunaan. Ohjelmakoodi on kommentoitu englanniksi ja sen toteutuksessa on käytetty englantia, jolloin järjestelmän mahdollinen myöhempi ylläpito ei ole sidottu suomen kieleen.



Kuva 1: Järjestelmän yleiskuvaus.

2.1 Sovellusalue

Aija-järjestelmä tarjoaa aikojenjakopalveluita eri tarkoituksiin. Tapaamisen järjestäjä laatii aikataulun, johon hän syöttää itselleen sopivat ajat tarjolle. Järjestäjä kutsuu osallistujat merkitsemään omat aikojen sopivuutensa tarjottuihin aikoihin. Osallistujat merkitsevät omat sopivuutensa, jonka jälkeen järjestäjä voi viimeistellä järjestelmän ehdottoman aikataulun.

2.2 Sopimukset ja standardit

Järjestelmän toteutuksessa käytetään seuraavia sopimuksia ja standardeja.

Nimi	Versio	Päiväys	Saatavuus	Kuvaus
ANSI SQL-92	ISO 9075:1992	30.7.1992	www.iso.org	SQL-relaatio-tietokantojen määrittelystandardi
CSS1	Level 1	11.1.1999	www.w3.org	XML-dokumenttien tyylisuositus
DTD	4.01 Transitional	24.12.1999	www.w3.org	HTML-dokumenttien tyyppimäärittely
HTML 4.01	4.01	24.12.1999	www.w3.org	HTML-sivun-kuvauskielen suositus
ISO 639	ISO 639:1988	18.7.2002	www.iso.org	Kansainväliset kielikoodit
ISO 3166	ISO 3166:1993	24.9.1997	www.iso.org	Kansainväliset maakoodit
ISO 8859-1	ISO 8859-1:1998	19.11.2003	www.iso.org	8-bitin merkistöstandardi latinalaisella aakkostolla
Java Language Specification	3. painos	6/2005	java.sun.com	Ohjelmakoodauksessa noudatetaan Java-ohjelmointikielen notatiota soveltuvin osin
Javadoc	1.4.2		java.sun.com	Ohjelmakoodi kommentoidaan noudattamalla Javadoc-työkalun ohjeita elementtien käytöstä
JavaScript	1.3		mozilla.org/js/	Osa käyttöliittymän toiminnallisuudesta toteutetaan JavaScriptillä
RFC 2821	2821	4/2001	www.faqs.org	Simple Mail Transfer Protocol, sähköpostien lähetysprotokolla
RFC 2822	2822	4/2001	www.faqs.org	Internet Message Format, käytetään lokitiedoston päivämäärämuodossa

2.3 Laitteistoympäristö

Ohjelma	Versio
Linux CSL	2.152
PHP	5.0.4
Apache	1.3
PostgreSQL	7.4

2.4 Toteutuksen keskeiset reunaehdot

2.4.1 Tietoturva

Järjestelmän tietoturva on määritellyllä tasolla. Käyttäjä voi kuitenkin rekisteröityä ilman autentikointia, sillä hän pääsee käyttämään järjestelmää heti rekisteröitymisen jälkeen. Tietoturvaa tulisi lisätä esimerkiksi siten, että rekisteröityneen käyttäjän on kirjauduttava sisään sähköpostiinsa lähetetyllä salasanaalla, jonka jälkeen hän pääsee järjestelmään. Näin varmistetaan, että käyttäjän rekisteröitymisessä antama sähköpostiosoite kuuluu hänelle.

Mikäli käyttäjä rekisteröityy sähköpostiosoitteella, joka ei kuulu hänelle, jää tämä osoite järjestelmään määritellyksi ajaksi. Jos edellä kuvattu tietoturvan tason nosto toteutetaan, voidaan rekisteröidyt osoitteet poistaa järjestelmästä nopeamminkin. Tällöin tutkitaan rekisteröintiäikää sekä vierailuaikaa. Jos vierailuakaleimaa ei vielä ole ja rekisteröinnistä on kulunut esimerkiksi 30 päivää, voidaan osoite poistaa järjestelmästä uutta rekisteröintiä varten.

2.4.2 Sekvenssin käyttö

Kukin tietue saa yksilöllisen avainarvon tietokannassa. Avainarvot saadaan sekvenssiltä, joka palauttaa järjestyksessä aina seuraavan vapaan kokonaisluvun. Mikäli tietueen tallennus epäonnistuu ja sekvenssiltä on jo saatu avainarvo, jää tämä avain käyttämättä. Seuraavalla yrityskerralla sekvenssi antaa uuden avainarvon.

Sekvenssillä on yläraja, joka riippuu käytettävästä tietokantajärjestelmästä. Periaatteessa sekvenssin yläraja on saavutettavissa, jonka jälkeen järjestelmään ei voi enää tallentaa uusia tietoja. Tietokannassa jo olevat tietueet tulisi järjestää uudelleen ja antaa uusi pienempi käyttämätön avainarvo. Tämän jälkeen sekvenssi voidaan alustaa alkamaan seuraavasta vapaasta avainarvosta.

2.4.3 Viestit

Järjestäjä voi muokata viestipohjia, jotka muokattuina tallentuvat tietokantaan. Viestipohjaan voidaan liittää elementtejä, jotka kertovat viestissä aikataulusta varatun ajan. Mikäli aikataulu poistetaan, jää muokattuun viestiin elementti viittaamaan poistettuun aikatauluun. Nämä elementit järjestäjä muuttaa itse käsin.

Viestipohjaan voi lisätä elementtejä merkiksi vaihtuvan tiedon sijainnista tekstissä. Elementit ovat suomenkielisiä, mikä voi vaikeuttaa järjestelmän käyttöä suomea taitamattomalle.

2.4.4 Käyttäjätiedot

Järjestelmässä on yksi tietokantataulu käyttäjätietoja varten. Tähän tauluun tallennetaan sekä järjestäjän että osallistujan tiedot. Järjestäjä voi muuttaa osallistujan

tietoja, kuten nimeä ja sähköpostiosoitetta. Jos osallistuja on osallistujana jo jossakin aikaisemmin luodussa ryhmässä, käyttää järjestelmä näitä tietoja. Järjestäjällä on kuitenkin mahdollisuus muuttaa osallistujan tietoja, jolloin muutokset näkyvät myös aiemmin luodussa ryhmässä. Järjestäjä ei voi kuitenkaan muuttaa sellaisen osallistujan tietoja, joka on järjestäjänä jossakin toisessa ryhmässä. Tämä mahdollisuus voidaan poistaa esimerkiksi luomalla oma taulu osallistujia varten.

3 Arkkitehtuuri

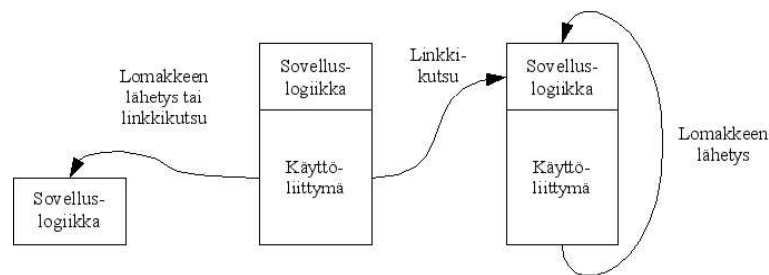
Järjestelmä on jaettu toiminnallisiin osajärjestelmiin. Tällaisia osajärjestelmiä ovat esimerkiksi aikataulujen laatiminen, osallistujien aikojen sopivuuksien syöttäminen ja omien asetusten käsittelyminen. Kukin osajärjestelmä on toteutettu omalla PHP-sivullaan, jotka puolestaan käyttävät järjestelmän komponenttien palveluja. Tietokantapalvelu ja monikielisyyden tuki ovat esimerkkejä järjestelmän komponenteista.

Selainpohjaisena toteutuksena järjestelmä tarjoaa palveluja Internet-verkossa. Hyödyntämällä HTML-sivunkuvauskielen mahdollisuuksia järjestelmän intuitiivinen ja graafinen käyttöliittymä tarjoaa käyttäjälle kaiken tarvittavan informaation eikä erillistä ohjeistusta ole suunniteltu järjestelmän käyttämiseksi. Toteutusvälineeksi valittua PHP-ohjelmointikieltä käytetään proseduraalisesti suoritustehokkuuden lisäämiseksi. Ohjelmoinnissa noudatetaan soveltuvin osin Java-ohjelmointikielen notatiota.

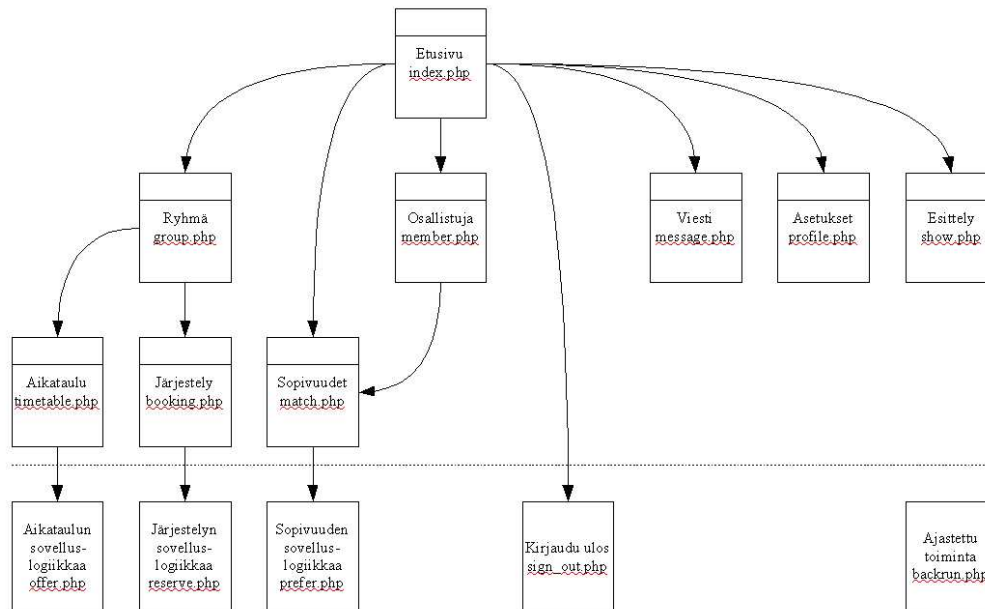
3.1 Osajärjestelmät

Sovelluslogiikka ja käyttöliittymä on erotettu toisistaan PHP-sivulla siten, että sivun alkuosassa on sovelluslogiikka ja tämän jälkeen alkaa käyttöliittymä ja siihen liittyvä logiikka. HTML-sivun lomakkeet lähetetään prosessoitavaksi pääsääntöisesti samalle sivulle, missä ne sijaitsevat. Tällä tavalla sovelluslogiikkaa ei tarvitse pitää erillään siihen loogisesti kuuluvasta käyttöliittymästä. Yhdistämällä sovelluslogiikka ja käyttöliittymä samalle PHP-sivulle nopeutetaan sivun prosessointia hieman, sillä PHP-tulkin ei tarvitse ladata erillistä sovelluslogiikkaosaa kiintolevyiltä. Myös ohjelmoijan ja ylläpitäjän työ helpottuu, kun osat ovat yhdessä. Sovelluslogiikka, jolla ei ole käyttöliittymää, on toteutettu komponentteina. Kuvassa 2 on kuvattu sivujen yleisluonteinen rakenne sekä osajärjestelmien eri kutsutavat.

Sivu koostuu sivun alussa olevasta sovelluslogiikasta sekä sivun loppuosan käyttöliittymälogiikasta. Käyttöliittymä tarjoaa käyttäjälle linkit muihin osajärjestelmiin sekä mahdollisesti lomakkeen tietojen syöttämistä varten. Lomake lähetetään sivulle itselleen, jolloin sivun sovelluslogiikka prosessoi lomakkeen tiedot. Osajärjestelmä voi lähettää lomakkeen prosessoitavaksi myös pelkälle sovelluslogiikalle, jolloin käyttäjälle ei näytetä käyttöliittymää. Esimerkiksi uloskirjautuminen kutsuu erillistä uloskirjausosajärjestelmää, joka puolestaan siirtää käyttäjän etusivulle. Uloskirjausosajärjestelmällä ei ole omaa käyttöliittymää.



Kuva 2: Osajärjestelmien kutsutavat.



Kuva 3: Osajärjestelmät.

3.1.1 Kommunikointi

Osajärjestelmät välittävät tietoa toisilleen kahdella tavalla, joko parametrina tai evästeenä. Parametri välitetään osajärjestelmää kutsuvassa osoitteessa. Eväste välitetään istuntokohtaisissa muuttujissa. Osajärjestelmät välittävät tietoa komponenteille parametreilla, jotka on esitelty kunkin palvelun yhteydessä. PHP-ohjelmointikielessä istuntokohtaisia muuttujia käsitellään String-tyyppisellä taulukkomuuttujalla `$_SESSION`. Alla olevassa taulukossa on lueteltu elementtien nimet.

Nimi	Kuvaus
lang	Käyttäjän valitsema käyttöliittymän kielikoodi on tallennettu lang-elementtiin. Jos elementtiä ei ole tai se on tyhjä, käyttää järjestelmä oletuskielikoodia. Oletuskielikoodi on esitelty alustuskomponentissa globaalina vakiona DEFAULT_LANGUAGE.
reason	Mikäli osajärjestelmän kutsu epäonnistuu, välitetään kutsuneelle järjestelmälle tieto epäonnistumisen syystä reason-elementissä.
backto	Jos kutsuttu osajärjestelmä palauttaa suorituksen kutsuneelle järjestelmälle, voidaan kutsuttuun osajärjestelmään palata backto-elementissä olevaan osoitteeseen. Tätä osoitetta käytetään tyypillisesti silloin, kun sisäänkirjautumaton käyttäjä kutsuu sisäänkirjautumista edellyttävää osajärjestelmää. Osajärjestelmä tallentaa osoitteensa parametreineen backto-elementin arvoksi ja siirtää suorituksen sisäänkirjautumisosajärjestelmälle. Sisäänkirjautumisen jälkeen voidaan palata alkuperäiseen osajärjestelmään käyttämällä tätä osoitetta.
user	Sisäänkirjautuneen käyttäjän yksilöllinen avainarvo tietokannassa tallennetaan user-elementin arvoksi istunnon ajaksi. Jos elementissä on nollaa suurempi arvo, on käyttäjä kirjautunut järjestelmään. Mikäli elementtiä ei ole tai arvo on nolla tai sitä pienempi, ei käyttäjä ole sisäänkirjautuneena.

Osajärjestelmän kutsussa voidaan käyttää parametreja, jotka on lueteltu alla olevassa taulukossa. Nämä parametrit ovat yhteisiä kaikille osajärjestelmille.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
lang	String	Ei	Käyttöliittymän kielikoodi voidaan valita välittämällä se lang-parametrilla. Jos parametria ei ole tai sen arvo ei ole tuettu kielikoodi, käyttää järjestelmä oletuskielikoodia. Oletuskielikoodi on esitelty alustuskomponentissa globaalina vakiona DEFAULT_LANGUAGE.

Kukin osajärjestelmä tarkistaa parametrien oikeellisuuden. Numeeriset parametrit, jotka viittaavat tietokannan avaimiin, konvertoidaan positiivisiksi kokonaisluvuiksi. Jos parametri on tietueen avain, tarkistaa sovelluslogiikka, että lapsitietueet kuuluvat sisäänkirjautuneelle käyttäjälle. Järjestäjä voi esimerkiksi avata muokattavaksi sähköpostiviestin tekstipohjan antamalla mielivaltaisen parametrin osoitteeksi. Järjestelmä tarkistaa kuitenkin ensin, että kyseinen tietue kuuluu sisäänkirjautuneelle järjestäjälle.

3.1.2 Yleiset palvelut

Tietyt palvelut ovat yhteisiä kaikille osajärjestelmille. Nämä palvelut edellyttävät kuitenkin sisäänkirjautumista. Osajärjestelmien yleiset palvelut on lueteltu alla olevassa taulukossa.

Palvelu	Kuvaus
Kielen vaihto	Käyttäjä vaihtaa käyttöliittymän kielen. Jos sisäänkirjautuneen käyttäjän tietokannasta haetuissa henkilökohtaisissa asetuksissa on eri kielikoodi, käyttää järjestelmä ensisijaisesti käyttäjän tietokantaan tallentamaa koodia. Käyttäjä voi vaihtaa käyttöliittymän kielen etusivulla kirjautumatta järjestelmään.
Uloskirjautuminen	Käyttäjä kirjautuu ulos järjestelmästä ja suoritus siirtyy takaisin sisäänkirjautumisosajärjestelmälle.
Asetukset	Käyttäjä muuttaa tai katselee omia asetuksiaan, jotka on tallennettu tietokantaan. Käyttäjä voi myös vaihtaa salasansansa tällä palvelulla.
Käyttäjän rooli	Käyttäjä voi olla joko järjestäjä tai osallistuja. Käyttäjä on järjestäjä-roolissa, jos hänen sähköpostiosoitteensa kuuluu järjestäjille sallittujen toimialueiden piiriin. Muussa tapauksessa käyttäjä on osallistuja-roolissa. Järjestelmä tarkistaa jokaisen osajärjestelmä käynnistyksessä, mikä rooli käyttäjällä kulloinkin on. Rooli voi vaihtua myös kesken istunnon, mikäli toimialueissa tehdään muutoksia istunnon aikana.

3.1.3 Syötteet

Järjestelmä tarkistaa käyttäjän antamat syötteet. Syötteet siivotaan poistamalla teksteistä ylimääräiset kenoviivat, joita PHP lisää lomakkeiden lähetyksessä. Tekstit muokataan myös siten, että ne voidaan näyttää HTML-lomakkeella. Tämä tarkoittaa sitä, että tekstissä mahdollisesti olevat kulmasulkeet ja esimerkiksi skandinaaviset merkit muutetaan HTML-lomakkeella esitettävään muotoon. Pääsääntöisesti tekstisyötteen alusta ja lopusta poistetaan välilyönnit. Mikäli syötteen pituus on yli sallitun pituuden, katkaisee järjestelmä syötteen enintään sallittuun pituuteen.

Jos käyttäjän antama syöte on virheellinen, antaa järjestelmä tästä ilmoituksen käyttäjälle (ks. viestityskomponentti) ja samalla osoittaa lomakkeen kentän, jossa virhe on havaittu korostamalla kentän taustaväriä. Virheellisetkin syötteet jäävät lomakkeelle käyttäjän korjaustoimenpiteitä varten.

Järjestelmä ei hae tietokannasta tietoja ennen kuin käyttäjän antamien syötteen tallennus on täysin onnistunut. Tämän vuoksi sovelluslogiikassa käytetään Boolean-tyyppistä muuttujaa (`$tmpReload`), joka ilmaisee, voidaanko tiedot noutaa tietokannasta. Muuttujan arvo alustetaan arvolla `true`, joka sallii järjestelmän noutaa tiedot. Tätä käytetään, kun ensimmäisen kerran tullaan sivulle. Syötteen tarkis-

tuksessa muuttujan arvo asetetaan arvoksi false, joka estää alkuperäisten tietojen noutamisen tietokannasta ja näin ollen syötteen jäävät lomakkeelle. Jos tallennus onnistuu, palautetaan muuttujan arvoksi true, jolloin järjestelmä voi noutaa vastatallennetut tiedot ja päivittää lomakkeella olevat syötteen.

3.1.4 Virhekäsittely sovelluslogiikassa

Sovelluslogiikkaa suoritettaessa voi syntyä virhetilanteita. Virhetilanteet on jaettu fataaleiksi ja hallituiksi. Fataalille virheelle on tyypillistä suorituksen keskeytyminen esimerkiksi tietokantakyselyn epäonnistuesssa, jolloin sovelluslogiikka ei voi jatkaa. Hallitut virheet ovat tyypillisesti käyttäjän antamien syötteiden tarkistuksessa syntyneitä virhetilanteita. Fataalin virheen synnyttyä sovelluslogiikan suoritus keskeytyy ja sovelluksen suoritus siirtyy virheestä riippuen toiselle PHP-skriptille. Jos esimerkiksi käyttäjä ei ole kirjautunut sisään ja sovelluslogiikka edellyttää sisäänkirjautumista, keskeytyy sovelluslogiikan suoritus ja sovelluksen suoritusta jatketaan sisäänkirjautumisajärjestelmässä. Hallituista virhetilanteista käyttäjä saa virheilmoituksen käyttöliittymän viestityskomponentilla eikä sovelluslogiikan suoritus keskeydy.

3.1.5 Aikataulu - timetable.php

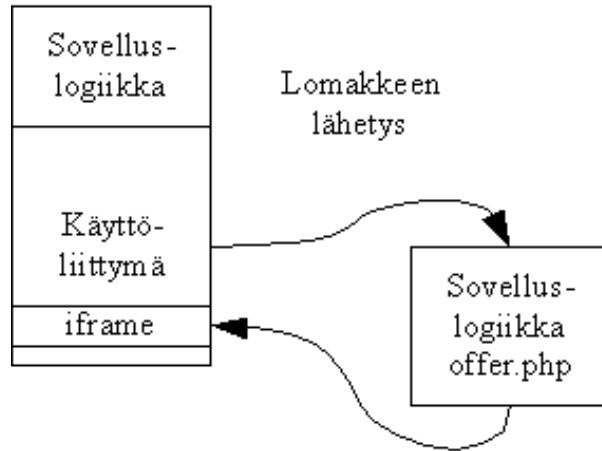
Järjestäjä luo aikatauluja ja merkitsee niihin tarjolla olevia aikoja tapaamisia varten. Järjestäjä ilmoittaa aikataulun keston viikkoina valitsemalla viikot kalenterista. Lisäksi järjestäjä valitsee aikataulun tyyppin, joko henkilökohtaisille tapaamisille tai ryhmätapaamisille ja määrittää tapaamisen keston eli tuntijaon.

Kun aikataulu on tallennettu ensimmäisen kerran, näyttää reunalistakomponentti tallennetun aikataulun nimen sekä sen alla "Tarjotut ajat" ja "Aikojen järjestelylinkit, joista ensin mainittu on korostettuna.

Järjestäjä merkitsee tarjolla olevat ajat viikonäytöllä. Jokainen merkintä tallennetaan tietokantaan välittömästi merkinnän jälkeen. Jotta järjestelmä ei suorittaisi sivun uudelleenlatausta, kutsutaan lomakkeen lähetyksessä erillistä aikataulun sovelluslogiikkaa (offer.php). Erillisellä sovelluslogiikalla ei ole omaa käyttöliittymää, ja sen tehtävänä on tallentaa tietokantaan parametrina saatu tarjottu aika tai poistaa aika tietokannasta. Lomakkeen lähetys kutsuu sovelluslogiikkaa ja ohjaa mahdollisen tulosteen sivullaan olevalle iframe-kehykselle. Tämä kehys ei ole näkyvissä, joten myöskään sovelluslogiikan tulosteet eivät tule näkyviin. Kuva 4 esittelee toiminnallisuuden.

Aikataulujen tyyppit on tallennettu tietokannan Values-tauluun, koska käyttöliittymän lomakkeella näytetään kielikoodista riippuvaiset vaihtoehdot.

Aikataulu
timetable.php



Kuva 4: Aikataulun logiikka.

Kokoelma	Nimi	Kielikoodi	Arvo	Lajittelujärjestys
TIMETABLE	PERSONAL	fi	Henkilökohtainen	
TIMETABLE	PERSONAL	en-US	Personal appointment	
TIMETABLE	GROUP	fi	Ryhmätapaaminen	
TIMETABLE	GROUP	en-US	Group appointment	

Oletusarvo on Codes-taulussa nimellä TIMETABLE_TYPE kokoelmassa DEFAULT, ja tällä arvolla valitaan listalta aikataulun oletustyyppi. Aikataulun aloitus- ja loppusajat on ilmoitettava tunnit ja minuutit erotettuina kaksoispisteellä, esimerkiksi 8:00. Tässä kokoelmassa ovat myös muut kalenteriin liittyvät oletusarvot.

Kokoelma	Nimi	Arvo	Lajittelujärjestys
DEFAULT	TIMETABLE_TYPE	PERSONAL	
DEFAULT	TIMETABLE_START	8:00	
DEFAULT	TIMETABLE_FINISH	16:00	

Valmiit tuntijaot on tallennettu tietokannan Codes-tauluun. Oletusarvo on nimellä SLOT kokoelmassa DEFAULT, ja tällä arvolla valitaan listalta oletustuntijako. Oletusarvo on annettu minuutteina. Koodien arvot näytetään käyttöliittymän tuntijakolistalla, kun taas nimet ilmaisevat tuntijakojen keston minuutteina.

Kokoelma	Nimi	Arvo	Lajittelujärjestys
SLOT	10	10 min	1
SLOT	15	15 min	2
SLOT	20	20 min	3
SLOT	30	30 min	4
SLOT	60	1 h	5
SLOT	120	2 h	6
DEFAULT	SLOT	20	

Osajärjestelmä tarjoaa järjestäjälle seuraavat palvelut:

Palvelu	Kuvaus
Aikataulun nimeäminen	Järjestäjä nimeää aikataulun tai muuttaa nimeä.
Aikataulun poisto	Järjestäjä poistaa aikataulun. Samalla poistuvat aikatauluun viittaavat tietueet.
Tyyppin valinta	Järjestäjä valitsee aikataulun tyyppin (henkilökohtainen tapaaminen tai ryhmätapaaminen).
Tuntijaon valinta	Järjestäjä valitsee tuntijaon.
Aloitus- ja lopetusaika	Järjestäjä valitsee, mistä kellonajasta viikkoa aloitetaan esittää ja mihin kellonaikaa esitys päätetään.
Aikataulun kesto	Järjestäjä valitsee kalenterista aikataulun keston viikkoina eli ne viikot, jolloin aikataulu on voimassa.
Tilannetiedot	Järjestäjälle näytetään, montako aikaa hän on tarjonnut aikataulusta sekä miten monta osallistujaa ryhmässä on.
Ajan tarjoaminen	Järjestäjä merkitsee tarjottavan ajan viikkonäytöllä tai poistaa tarjotun ajan.

Osajärjestelmän kutsussa voidaan käyttää seuraavia parametreja yhteisten parametrien lisäksi.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
id	int	Kyllä	Aikataulun yksilöllinen avain tietokannassa, jos tällä avaimella ei löydy aikataulua tietokannasta tai se ei ole sisäänkirjautuneen järjestäjän ryhmään kuuluva osallistuja, siirtyy järjestelmä uuden aikataulun lisäystilaan ja näyttää järjestäjälle virheilmoituksen.
p	int	Kyllä	Viiteavain Groups-aulun tietueeseen, joka viittaa ryhmään, johon aikataulu kuuluu. Jos avain ei kuulu sisäänkirjautuneen järjestäjän ryhmälle, siirtää järjestelmä järjestäjän uuden ryhmän luontitilaan Ryhmä-sivulle ja näyttää virheilmoituksen.

Yleiskuvaus sovelluslogiikasta

Osajärjestelmän sovelluslogiikka on kuvattu alla olevalla pseudokoodilla. Suluissa olevat numerot viittaavat SQL-lauseeseen, joka on kuvattu koodin lopussa.

Alusta istunto

Lataa komponentit

Jäsennä parametrit

Tarkista kielikoodi

Tarkista istunnon voimassaolo

Jos istunto ei ole voimassa

Valmistaudu siirtymään sisäänkirjausosajärjestelmään

Muutoin jos sisäänkirjautunut ei ole järjestäjä

Valmistaudu siirtymään sisäänkirjausosajärjestelmään

Muutoin jos parametrina annettu ryhmä ei ole järjestäjän (1)

Valmistaudu siirtymään ryhmän osajärjestelmään annetulla parametrilla

Muutoin

Alusta muuttujat

Jos järjestäjä on valinnut aikataulun poiston

Poista aikataulu (2)

Muutoin jos lomake on lähetetty ja lomakkeen taulukkomuuttujassa (\$_POST) on arvoja

Tarkista syötteet

Tarkista aikataulun tyyppi (3)

Tarkista tuntijako (4)

Jos syötteet kelpaavat

Jos aikataulun avain on nolla

Tallenna aikataulu (5)

Muutoin

 Päivitä aikataulu (6)

 Jos tallennus onnistui

 Jos järjestäjä on valinnut tarjotun ajan poiston

 Poista tarjottu aika (7)

 Muutoin

 Tallenna tarjottu aika (8)

 Jos aikataulun avain ei ole nolla

 Lue aikataulun tiedot tietokannasta (9)

 Jos aikataulu löytyi

 Pura tietue muuttujiin

 Laske tarjottujen aikojen määrä (10)

 Laske osallistujien määrä (11)

 Jos ollaan siirtymässä toiseen osajärjestelmään

 Sulje tietokanta

 Siirry osajärjestelmään

 Tallenna sivuhistoria

 Siivoa muuttujien arvot

 Jos muuttujista puuttuu arvoja

 Aseta oletusarvot

 Esittele paikalliset funktiot

 Näytä käyttöliittymä

 Laadi luettelo aikataulun tyypeistä (12)

 Laadi luettelo aikataulun tuntijaoista (13)

 Jokaiselle ajalle viikkonäytöllä

 Tarkista, onko tietokannassa tarjottu aika (14)

 Terminoi osajärjestelmä

SQL-lauseet

Numero	Lause
1	SELECT * FROM Groups WHERE ID = (ryhmän avain) AND UserID = (järjestäjän avain)
2	DELETE FROM Calendars WHERE GroupID = (ryhmän avain) AND ID = (aikataulun avain)
3	SELECT * FROM Values WHERE Name = '(aikataulun tyyppi)' AND Collection='TIMETABLE'
4	SELECT * FROM Codes WHERE Name = '(aikataulun tuntijako)' AND Collection = 'SLOT'
5	INSERT INTO Calendars (ID,GroupID,Name,Type,Start,Finish,Slot) VALUES (arvolista)
6	UPDATE Calendars SET Name,Type,Start,Finish,Slot WHERE GroupID = (ryhmän avain) AND ID = (aikataulun avain)
7	DELETE FROM Offers WHERE CalendarID = (aikataulun avain) AND Period = '(aika)' AND Weekday = (viikonpäivän numero)
8	INSERT INTO Offers (ID,CalendarID,Weekday,Period) VALUES (arvolista)
9	SELECT * FROM Calendars WHERE GroupID = (ryhmän avain) AND ID = (aikataulun avain)
10	SELECT COUNT(*) CNT FROM Offers WHERE CalendarID = (aikataulun avain)
11	SELECT COUNT(*) CNT FROM Members WHERE GroupID = (ryhmän avain)
12	SELECT Name,Value FROM Values WHERE Collection = 'TIMETABLE' AND Lang='(valittu kielikoodi)' ORDER BY Sorting
13	SELECT Name,Value FROM Codes WHERE Collection = 'SLOT' ORDER BY Sorting
14	SELECT ID FROM Offers WHERE CalendarID = (aikataulun avain) AND Period = '(aika)' AND Weekday = (viikonpäivän numero)

Toiminnallisuus

- Istunnon voimassaolo tarkistetaan ja tarvittaessa käyttäjä siirretään etusivulle sisäänkirjautumista varten.
- Varmistetaan, että parametrina annettu viiteavain kuuluu sisäänkirjautuneen järjestäjän ryhmälle.
- Jos parametrina annettu viiteavain ei viittaa järjestäjän ryhmään, siirretään järjestäjä uuden ryhmän luontitilaan ja hänelle näytetään virheilmoitus.
- Jos parametrina annetulla avaimella ei löydy aikataulua, näytetään järjestäjälle virheilmoitus ja uuden aikataulun syöttö alkaa.
- Syötteiden alusta ja lopusta poistetaan välilyönnit.
- Aikataulun nimi ei saa puuttua.

- Aikataulun nimen enimmäispituus on 255 merkkiä.
- Aikataulun tyyppin on oltava jokin määritellyistä.
- Aikataulun tuntijaon on oltava jokin määritellyistä.
- Oletuksena on valittu oletusarvoissa määritelty tuntijako.
- Tuntijakoa ei voi muuttaa, jos aikatauluun on tehty merkintöjä.
- Tuntijaon vaihtaminen päivittää viikkonäyttöä vastaavasti.
- Järjestäjälle näytetään, miten monta aikaa hän on merkinnyt tarjolle.
- Järjestäjälle näytetään, miten monta osallistujaa ryhmässä on.
- Viikkonäytön aloitus- ja lopetusaika on väliltä 00:00 - 24:00.
- Aloitus- ja lopetusaika on jaettu tuntijaon mukaisesti.
- Oletuksena on valittu oletusarvoissa määritelty aloitus- ja lopetusaika.
- Viikkonäyttö jaetaan tuntijaon mukaisesti.
- Viikkonäytössä näytetään arkipäivien lisäksi viikonloppu.
- Viikkonäyttö alkaa maanantaista.
- Tarjotun ajan merkintä tallentuu tietokantaan ilman erillistä lomakkeen lähetystä.
- Tarjotun ajan merkintä tallentaa myös aikataulun nimen ja muut valitut tiedot, jos niitä ei ole vielä tallennettu.
- Jos järjestäjä merkitsee uudelleen jo ennestään merkityn ajan, poistuu merkitty aika.
- Jos tarjottu aika poistetaan, poistuvat samalla osallistujien siihen merkitsemät aikojen sopivuudet.
- Tilastot päivittyvät jokaisen tarjotun ajan merkinnän jälkeen.
- Käyttöliittymän kielikoodi vaihdetaan joko parametrina annettuun tai istuntauuuttujassa olevaan.
- Kielikoodin on oltava tuettu, oletusarvo on `DEFAULT_LANGUAGE`.
- Poistopainike on näkyvässä vain, jos ollaan muokkaamassa olemassa olevaa aikataulua.
- Järjestäjä voi poistaa aikataulun.

- Järjestäjää pyydetään vahvistamaan poisto.
- Jos järjestäjä vahvistaa poiston, aikataulu poistetaan tietokannasta ja kaikki viittaukset aikatauluun poistetaan samalla.
- Poiston jälkeen alkaa uuden aikataulun syöttö.
- Aikataulun tiedot tallennetaan Calendars-tilaan.

3.1.6 Aikataulun sovelluslogiikka - offer.php

Kun järjestäjä merkitsee ajan tarjolle, kutsuu aikataulu-osajärjestelmä erillistä sovelluslogiikkaa (offer.php). Tämä sovelluslogiikka tallentaa tarjotun ajan tietokantaan tai poistaa sen, jos järjestäjä valitsee jo tarjotun ajan uudelleen. Käyttämällä erillistä sovelluslogiikkaa ei varsinaista käyttöliittymää tarvitse muodostaa uudelleen, sillä viikkonäytön lomake lähetetään tälle erilliselle sovelluslogiikalle.

Sovelluslogiikka tarjoaa aikataulu-osajärjestelmälle seuraavat palvelut:

Palvelu	Kuvaus
Tarjotun ajan tallennus	Järjestäjän valitsema aika tallennetaan tarjotuksi ajaksi Offers-tilaan, jos valittu aika ei ole vielä tarjolla.
Tarjotun ajan poisto	Jos järjestäjä valitsee jo tarjotun ajan uudelleen, poistaa järjestelmä sen tietokannasta.

Sovelluslogiikan kutsussa käytetään seuraavia parametreja.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
calendarID	int	Kyllä	Aikataulun yksilöllinen avain tietokannassa. Avain ilmoittaa, mille aikataululle tarjottu aika tallennetaan tai poistetaan.
weekday	int	Kyllä	Viikonpäivän numero, missä 1 = maanantai, ..., 7 = sunnuntai. Viikonpäivän numero ilmoittaa, minkä viikonpäivän tarjotusta ajasta on kyse.
period	Time	Kyllä	Viikkonäytöltä valittu aika, joka ilmoittaa, milloin aika tulee tarjolle. Tätä arvoa käytetään myös haettaessa tietokannasta poistettava aika.
action	String	Kyllä	Käskey, joka ilmoittaa, minkä operaation sovelluslogiikka tekee. Sallitut arvot ovat INSERT, DELETE; missä INSERT = tarjottu aika lisätään tietokantaan ja DELETE = aiemmin tarjottu aika poistetaan tietokannasta.

Yleiskuvaus sovelluslogiikasta

Aikataulun sovelluslogiikka on kuvattu alla olevalla pseudokoodilla. Suluissa olevat numerot viittaavat SQL-lauseeseen, joka on kuvattu koodin lopussa.

Alustaistunto

Lataa komponentit

Tarkista istunnon voimassaolo

Jos istunto on voimassa

 Muodosta yhteys tietokantapalvelimelle

 Jos yhteys muodostui

 Avaa tietokanta

 Jos tietokannan avaaminen onnistui

 Jäsennä parametrit

 Tarkista parametrit

 Jos parametrit kelpaavat

 Jos tarjottu aika pitää poistaa

 Poista tarjottu aika (1)

 Muutoin

 Tallenna tarjottu aika (2)

 Sulje tietokanta ja tietokantayhteys

SQL-lauseet

Numero	Lause
1	DELETE FROM Offers WHERE CalendarID = (aikataulun avain) AND Period = '(aika)' AND Weekday = (viikonpäivän numero)
2	INSERT INTO Offers (ID,CalendarID,Weekday,Period) VALUES (arvolista)

Toiminnallisuus

- Istunnon voimassaolo tarkistetaan ja jos istunto ei ole voimassa, sovelluslogiikan suoritus keskeytyy ja suoritusta aikataulu-osajärjestelmässä jatketaan.
- Jos parametrina annettu viiteavain aikatauluun on pienempi kuin yksi, suoritus keskeytyy.
- Jos parametrina annettu viikonpäivän numero ei ole joukossa 1, 2, ..., 7, suoritus keskeytyy.
- Jos parametrina annettua aikaa ei ole, suoritus keskeytyy.
- Parametrien arvot tallennetaan tarjotuksi ajaksi tietokantaan, jos kutsunut osajärjestelmä on pyytänyt tallennusta.
- Tarjottu aika poistetaan tietokannasta parametrien ilmoittamilla arvoilla, jos kutsunut osajärjestelmä on pyytänyt poistoa.

3.1.7 Ajastettu toiminta - backrun.php

Järjestelmä suorittaa tiettyjä toimintoja ajastetusti. Käyttöjärjestelmä kutsuu backrun.php -skriptiä ajastustaulussa (cron job) olevan aikataulun mukaisesti. Aikataulu asetetaan suoritettavaksi päivittäin. Backrun.php suorittaa seuraavat tehtävät:

Tehtävä	Kuvaus
Vanhentuneen aikataulun poisto	Järjestelmä poistaa aikataulun ja siihen viittaavat tietueet, kun aikataulun päättymisestä on kulunut yli 30 päivää. Aikataulun voimassaolo päättyy viimeisen aikatauluun merkityn viikon lopussa.
Vanhentuneen ryhmän poisto	Järjestelmä poistaa ryhmän ja siihen viittaavat tietueet, kun ryhmään ei ole luotu uusia aikatauluja 14 kuukauteen. Kun järjestäjä luo uuden aikataulun, päivittää järjestelmä ryhmätietueen Update-kentän aikaleiman, josta ryhmän vanhentuminen laskeaan. Kentän aikaleima alustetaan ryhmää luotaessa, jolloin se ilmaisee ryhmän luontiajan, kunnes järjestäjä luo ensimmäisen aikataulun.
Vanhentuneen käyttätilin poisto	Järjestelmä poistaa käyttäjän ja siihen viittaavat tietueet, kun viimeisimmästä sisäänkirjautumisesta on kulunut yli 14 kuukautta. Jos käyttäjä ei ole kirjautunut kertaakaan sisään, katsotaan käyttäjä vanhentuneeksi, kun rekisteröitymisestä on kulunut yli 14 kuukautta. Käyttäjätietoinen vanhentuminen alkaa, kun käyttäjä ei ole enää yhdesäkään voimassa olevassa ryhmässä järjestäjänä. Käyttäjän viimeisin sisäänkirjautumisaika päivitetään Visited-kenttään. Rekisteröitymisaika päivitetään Created-kenttään.

Tarvittavat SQL -lauseet aikataulun poistossa:

1. `SELECT calendarid, year, MAX(week) AS week FROM weeks WHERE (calendarid,year) IN (SELECT DISTINCT calendarid, MAX(year) AS year FROM weeks GROUP BY calendarid) GROUP BY calendarid, year`
2. `DELETE FROM Calendars WHERE id IN (Poistettavien kalentereiden joukko)`

Tarvittavat SQL-lauseet ryhmän poistossa. Huomioi että hakasulkeet eivät kuulu lauseeseen.

1. `DELETE FROM groups WHERE updated < TIMESTAMP [14kk kuluvasta päivästä]`

Tarvittavat SQL-lauseet Käyttäjän -poistossa. Huomaa että hakasulkeet eivät kuulu lauseeseen.

```
1. DELETE FROM users
   WHERE (visited = NULL AND created < [aikaleima 14 kk kuluva päivästä])
   OR (id NOT IN (
   SELECT U.id FROM users AS U, Groups AS G
   WHERE G.userid = U.id) OR ( visited < TIMESTAMP [aikaleima 14kk kuluva päivästä]))
```

Järjestelmä aloittaa skriptin suorituksen tutkimalla, onko /log-kansiossa lokitiedosto. Jos kansiossa on lokitiedosto, järjestelmä lukee sen ja selvittää sen sisällöstä, montako kertaa skriptin suoritusta on yritetty aikaisemmilla kerroilla. Jos skriptiä on suoritettu kolme kertaa tai useammin, laatii järjestelmä lokitiedoston sisällöstä sähköpostiviestin ja lähettää sen ylläpitäjän sähköpostiosoitteeseen. Järjestelmä merkitsee lokitiedostoon lähetyspäivämäärän, tallentaa lokitiedoston ja keskeyttää skriptin suorituksen.

Ylläpitäjän sähköpostiosoite on tallennettu Codes-taulun ADMINISTRATION-koelmaan ADMINISTRATOR-nimellä. Jos ylläpitäjän sähköpostiosoitetta ei ole saatavana, ei sähköpostiviestiä lähetetä.

Aikataulut on sijoitettu tietokantaan Calendars -tauluun. Aikataulun viimeinen voimassa olo viikko määritellään Weeks -taulun week -kentästä. Aikataulun poiston yhteydessä poistetaan myös Offers -taulusta ne monikot, jotka viittaavat kyseiseen aikatauluun Calendars -taulussa. Tämän johdosta myös Matches-taulusta poistetaan ne monikot jotka viittaavat Offers-taulun poistettaviin monikoihin.

Ryhmän on sijoitettu tietokantaan Groups -tauluun. Ryhmän vanhentuminen lasketaan Groups -taulun updated -kentästä. Ryhmän poiston yhteydessä poistetaan siihen viittavaat aikataulut Calendars -taulusta, osallistuja Members -taulusta ja viestit Message -taulusta.

Jos jonkin tehtävän suorituksen aikana syntyy virhetilanne, järjestelmä kirjoittaa lokitiedostoon tiedot syntyneestä virheestä, mikäli se katsotaan fataaliksi. Tiedostoon kirjoitetaan tieto siitä, missä kohtaa skriptiä virhe syntyi sekä mahdollinen virheen numero ja selväkielinen virheilmoitus. Järjestelmä jatkaa muiden tehtävien suorittamista.

Fataaliksi virheiksi luokitellaan seuraavat tilanteet:

- Tietokantaan ei saada yhteyttä.
- Taulua ei ole olemassa.
- Taulun kenttää ei ole olemassa.

Palvelu ei palauta mitään.

Lokitiedosto

Taustatoiminnon aikana voi syntyä virhetilanteita. Nämä virhetilanteet tallennetaan lokitiedostoon backrun.xml, joka sijaitsee log-kansiossa. Administraattorin on poistettava tiedosto ilmoituksen saatuaan tai aika ajoin tarkistettava, ettei lokitiedosto kasva liian suureksi. Lokia voi katsella selaimella, sillä siihen liittyy backrun.xsl-tyylitiedosto. Lokitiedoston rakenne:

```
<?xml version="1.0"encoding="ISO-8859-1"?>
<!DOCTYPE entries SYSTEM "backrun.dtd»
<?xml-stylesheet type="text/xsl"href="backrun.xsl"?>
<entries>
<count>yrittyskertojen määrä</count>
<sendDate>RFC 2822 mukaan formatoitu aika</sendDate>
<entry>
<date>RFC 2822 mukaan formatoitu aika</date>
<phase>vaiheen numero, missä virhe syntyi</phase>
<message>virheilmoitus</message>
<code>virheilmoituksen numero</code>
<sql>viimeksi yritetty SQL-lause</sql>
</entry>
</entries>
```

Esimerkki:

```
<?xml version="1.0"encoding="ISO-8859-1"?>
<!DOCTYPE entries SYSTEM "backrun.dtd»
<?xml-stylesheet type="text/xsl"href="backrun.xsl"?>
<entries>
<count>1</count>
<sendDate></sendDate>
<entry>
<date>Thu, 21 Dec 2000 16:01:07 +0200</date>
<phase>3</phase>
<message>Table Calendars does not exists.</message>
<code>14670</code>
<sql>DELETE FROM Calendars WHERE ID=19</sql>
</entry>
</entries>
```

backrun.dtd:

```
<?xml version="1.0"encoding="UTF-8"?>

<!ELEMENT entries (count,sendDate,entry+)>
```

```

<!ELEMENT count (#PCDATA)>
<!ELEMENT sendDate (#PCDATA)>

<!ELEMENT entry (date,phase,code?,message?,sql?)>
<!ELEMENT date (#PCDATA)>
<!ELEMENT phase (#PCDATA)>
<!ELEMENT code (#PCDATA)>
<!ELEMENT message (#PCDATA)>
<!ELEMENT sql (#PCDATA)>

```

Administraattorille lähetetään ilmoitus, kun yrityskertojen määrä on kolme tai suurempi. Viestipohja on log.txt tiedostossa res-kansiossa. Jos järjestelmä ei kykene lukemaan viestipohjaa, muodostetaan oheinen geneerinen ilmoitus. Taulukossa olevaa viestin otsikkoa käytetään myös viestipohjaa käytettäessä.

Attribuutti	Arvo
Otsikko	Problems running Aija background tasks
Viesti	An error has occurred while trying to run background tasks of Aija System. There is back-run.xml log file available in the log folder.

3.1.8 Asetukset - profile.php

Käyttäjä voi muuttaa omia järjestelmään tallentamia tietoja sekä vaihtaa salasansa. Osajärjestelmä tarjoaa seuraavat palvelut:

Palvelu	Kuvaus
Salasanan vaihtaminen	Käyttäjä vaihtaa salasansa antamalla sekä vanhan että uuden salasanan. Uusi salasanan on kirjoitettava kahteen kertaan varmistettavaksi.
Tietojen muutos	Käyttäjä voi halutessaan vaihtaa nimen ja sähköpostiosoitteen sekä käyttöliittymän kielen.

Osajärjestelmä ei käytä omia parametreja.

Salasanan voi vaihtaa muuttamatta muita tietojaan. Vastaavasti nimen ja/tai sähköpostiosoitteen voi muuttaa vaihtamatta salasanaansa. Samalla kertaa voi toisaalta sekä muuttaa tietojaan että vaihtaa salasansa. Salasanaa vaihdettaessa on käyttäjän annettava uusi salasana sekä kirjoitettava se toistamiseen varmistusta varten.

Yleiskuvaus sovelluslogiikasta

Osajärjestelmän sovelluslogiikka on kuvattu alla olevalla pseudokoodilla. Suluissa olevat numerot viittaavat SQL-lauseeseen, joka on kuvattu koodin lopussa.

Alustaistunto

Lataa komponentit

Jäsennä parametrit

Tarkista kielikoodi

Tarkista istunnon voimassaolo

Jos istunto ei ole voimassa

Valmistaudu siirtymään sisäänkirjausosajärjestelmään

Muutoin

Alusta muuttujat

Jos järjestäjä tai osallistuja on valinnut asetusten tallennuksen

Tarkista syötteet

Jos syötteet kelpaavat

Jos salasana on annettu

Tarkista annettu salasana (1)

Jos salasanaa ei ole annettu tai salasana kelpaa

Päivitä käyttäjän tiedot (2)

Nollaa salasananmuuttujat

Lue järjestäjän tai osallistujan asetukset tietokannasta (3)

Jos asetukset löytyivät

Pura tietue muuttujiin

Muutoin

Valmistaudu siirtymään sisäänkirjausosajärjestelmään

Jos ollaan siirtymässä toiseen osajärjestelmään

Sulje tietokanta

Siirry osajärjestelmään

Tallenna sivuhistoria

Siivoa muuttujien arvot

Näytä käyttöliittymä

Terminoi osajärjestelmä

SQL-lauseet

Numero	Lause
1	SELECT Password FROM Users WHERE Password = '(salasana)' AND ID = (käyttäjän avain)
2	UPDATE Users SET Lastname,Firstname,Email,Password,Lang WHERE ID = (käyttäjän avain)
3	SELECT * FROM Users WHERE ID = (käyttäjän avain)

Toiminnallisuus

- Syötteiden alusta ja lopusta poistetaan välilyönnit.

- Sukunimi ei saa puuttua.
- Sukunimen enimmäispituus on 64 merkkiä.
- Etunimi ei saa puuttua.
- Etunimen enimmäispituus on 64 merkkiä.
- Sähköpostiosoitteen enimmäispituus on 320 merkkiä.
- Sähköpostiosoitteen on oltava validi (ks. sähköpostikomponentin `invalidEmail`-palvelu).
- Uusi sähköpostiosoite ei saa olla jonkun toisen käyttäjän rekisteröimä.
- Käyttöliittymän kielikoodi vaihdetaan aluksi joko parametrina annettuun tai istuntomuuttujassa olevaan.
- Kielikoodin on oltava tuettu, oletusarvo on `DEFAULT_LANGUAGE`.
- Kielivalinta astuu heti voimaan tallennuksen jälkeen.
- Salasanojen enimmäispituus on 32 merkkiä.
- Salasanoja ei näytetä sellaisenaan käyttöliittymässä, vaan niiden tilalla näytetään tähtimerkkejä.
- Jos vanha salasana on annettu, pitää myös uusi ja sen varmistus antaa.
- Uuden salasanan on oltava sama kuin sen varmistus.
- Jos uusi salasana tai sen varmistus on annettu, pitää myös vanha salasana antaa.
- Vanhan salasanan on vastattava järjestelmässä olevaa.
- Salasanan vaihdon jälkeen salasanakentät tyhjennetään.
- Käyttäjä ei voi itse poistaa käyttäjätietoja.
- Käyttäjän tiedot tallennetaan `Users`-tauluun.

3.1.9 Etusivu - `index.php`

Pääsääntöisesti järjestelmää ryhdytään käyttämään kirjautumalla ensin sisään etusivulla. Käyttäjä voi kirjautua järjestelmään myös sisäänkirjautumislinkillä, jolloin linkki vie hänet suoraan `Sopivuudet (match.php)` -sivulle. Osajärjestelmä tarjoaa seuraavat palvelut:

Palvelu	Kuvaus
Järjestelmän esittely Rekisteröityminen	Käyttäjä siirtyy järjestelmän esittelyyn. Käyttäjä rekisteröityy järjestelmän uudeksi käyttäjäksi nimellään ja sähköpostiosoitteellaan. Järjestelmä lähettää generoidun salasanan annettu sähköpostiosoitteeseen.
Salasanana lähetys	Käyttäjä pyytää järjestelmää lähettämään salasanansa järjestelmään rekisteröityy sähköpostiosoitteeseen.
Sisäänkirjautuminen	Aiemmin rekisteröitynyt käyttäjä kirjautuu sisään järjestelmään sähköpostiosoitteellaan ja salasanallaan.

Osajärjestelmän kutsussa voidaan käyttää seuraavia parametreja yhteisten parametrien lisäksi.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
lang	String	Ei	Käyttöliittymän kielikoodi voidaan valita välittämällä se lang-parametrilla. Jos parametria ei ole tai sen arvo ei ole tietty kielikoodi, käyttää järjestelmä oletuskielikoodia. Oletuskielikoodi on esitelty alustuskomponentissa globaalina vakiona DEFAULT_LANGUAGE.
linkto	String	Ei	Osajärjestelmälle voidaan välittää linkto-parametrissa tieto osoitteesta, johon järjestelmä siirtää käyttäjän suorituksen päätyttyä. Jos parametria ei ole, järjestelmä siirtää käyttäjän osajärjestelmään, joka riippuu käyttäjän roolista (joko järjestäjä tai osallistuja).

Yleiskuvaus sovelluslogiikasta

Osajärjestelmän sovelluslogiikka on kuvattu alla olevalla pseudokoodilla. Suluissa olevat numerot viittaavat SQL-lauseeseen, joka on kuvattu koodin lopussa.

Alustaistunto

Lataa komponentit

Jäsennä parametrit

Tarkista kielikoodi

Jos osajärjestelmään tultiin toisesta osajärjestelmästä sisäänkirjautumista varten

Valmistaudu näyttämään mahdollinen virheilmoitus

Jos paluosoite kutsuneeseen osajärjestelmään on välitetty

Valmistaudu palaamaan kutsuneeseen osajärjestelmään

Alusta muuttajat

Jos käyttäjä on valinnut sisäänkirjautumisen

- Tarkista syötteet
- Jos syötteet kelpaavat
 - Tarkista annettu salasana (1)
 - Jos salasana kelpaa
 - Aseta käyttäjän tiedot muuttujiin
 - Jos käyttäjä on järjestäjä
 - Valmistaudu siirtymään edellisen istunnon sivulle tai ryhmän osajärjestelmään
 - Muutoin jos osallistujan edellisen istunnon aikataulu on tiedossa
 - Valmistaudu siirtymään aikataulun osajärjestelmään
 - Muutoin jos osallistujan vanhin keskeneräinen aikataulu on tiedossa
 - Valmistaudu siirtymään aikataulun osajärjestelmään
 - Jos ollaan siirtymässä toiseen osajärjestelmään
 - Päivitä käyttäjän vierailuaika (2)
 - Sulje tietokanta
 - Siirry osajärjestelmään

Muutoin jos käyttäjä on valinnut rekisteröitymisen

- Tarkista syötteet
- Tarkista sähköpostiosoite (3)
- Jos syötteet kelpaavat
 - Generoi salasana
 - Tallenna käyttäjä (4)
 - Jos tallennus onnistui
 - Lähetä salasana sähköpostilla
 - Jos käyttäjä on järjestäjä
 - Valmistaudu siirtymään ryhmän osajärjestelmään
 - Muutoin jos osallistujan edellisen istunnon aikataulu on tiedossa
 - Valmistaudu siirtymään aikataulun osajärjestelmään
 - Muutoin jos osallistujan vanhin keskeneräinen aikataulu on tiedossa
 - Valmistaudu siirtymään aikataulun osajärjestelmään
 - Jos ollaan siirtymässä toiseen osajärjestelmään
 - Sulje tietokanta
 - Siirry osajärjestelmään

Muutoin jos käyttäjä on valinnut salasanan lähetyksen

- Tarkista sähköpostiosoite (5)
- Jos sähköpostiosoite löytyy
 - Lähetä salasana sähköpostilla

Näytä käyttöliittymä

- Laadi luettelo järjestäjälle sallituista toimialueista (6)

Terminoi osajärjestelmä

SQL-lauseet

Numero	Lause
1	SELECT ID,Password,Lang FROM Users WHERE Email='(sähköpostiosoite)'
2	UPDATE Users SET Visited WHERE ID = (käyttäjän avain)
3	SELECT Email FROM Users WHERE Email='(sähköpostiosoite)'
4	INSERT INTO Users (ID, Lastname, Firstname, Email, Password, GUID, Lang, Visited, Created) Values (arvolista)
5	SELECT Lastname,Firstname,Password FROM Users WHERE Email='(sähköpostiosoite)'
6	SELECT Value FROM Codes WHERE Collection='DOMAIN'

Toiminnallisuus

- Näyttää virheilmoituksen reason-istuntomuuttujan perusteella, jos muuttujassa on jokin arvo.
- Ottaa muistiin ensimmäiset 2048 merkkiä linkto-parametrissa tai -istuntomuuttujasta ja tuhoaa istuntomuuttujan
- Vaihtaa linkto-parametrissa saadun arvon backto-istuntomuuttujan arvoksi, jos sellainen on saatavilla ja tuhoaa istuntomuuttujan
- Syötteiden alusta ja lopusta poistetaan välilyönnit.
- Sukunimi ei saa puuttua.
- Sukunimen enimmäispituus on 64 merkkiä.
- Etunimi ei saa puuttua.
- Etunimen enimmäispituus on 64 merkkiä.
- Sähköpostiosoitteen enimmäispituus on 320 merkkiä.
- Sähköpostiosoitteen on oltava validi (ks. sähköpostikomponentin invalidEmail-palvelu).
- Sähköpostiosoite ei saa puuttua (ks. myös sähköpostiosoitteen validointi).
- Rekisteröitävä sähköpostiosoite ei saa olla jo rekisteröity.
- Rekisteröityvän käyttäjän nimi selvitetään sähköpostiosoitteesta.
- Rekisteröityneelle käyttäjälle lähetetään salasana hänen antamaansa sähköpostiosoitteeseen.
- Rekisteröitynyt käyttäjä kirjataan sisään.

- Rekisteröityvälle käyttäjälle näytetään järjestäjille sallitut sähköpostiosoitteiden toimialueet.
- Salasana ei saa puuttua sisäänkirjauksessa.
- Sisäänkirjauksessa salasanan on vastattava järjestelmässä olevaa.
- Unohtunut salasana lähetetään järjestelmään rekisteröityyn sähköpostiosoitteeseen.
- Käyttöliittymän kielikoodi vaihdetaan joko parametrina annettuun tai istuntomuuttujassa olevaan.
- Kielikoodin on oltava tuettu, oletusarvo on `DEFAULT_LANGUAGE`.
- Järjestelmän kielikoodi vaihtuu käyttäjän asetusten mukaisesti.
- Käyttäjän sisäänkirjautumisaika päivitetään tietokantaan.
- Sisäänkirjautuneen käyttäjän tietokannassa oleva yksilöllinen avainarvo tallennetaan istuntomuuttujaan.
- Epäonnistuneesta sisäänkirjautumisesta annetaan virheilmoitus.
- Sisäänkirjautunut käyttäjä on järjestäjä, jos hänen roolinsa nimi on `ORGANIZER`, muussa tapauksessa hän on osallistuja.
- Sisäänkirjautumisen jälkeen käyttäjä siirretään sivulle, jonka hän yritti avata, jos tämä tieto on saatavilla.
- Sisäänkirjautumisen jälkeen järjestäjä siirretään sille sivulle, joka hänellä oli avoinna edellisen istuntokerran lopussa.
- Jos edellisen istuntokerran tietoa ole saatavana, siirretään järjestäjä uuden ryhmän luontitilaan.
- Järjestelmä siirtää osallistujan vanhimpaan keskeneräiseen aikatauluun.
- Jos keskeneräistä aikataulua ei kyetä valitsemaan, järjestelmä siirtää osallistujan aikataulusivulle, joka hänellä oli avoinna edellisen istuntokerran lopussa.
- Jos edellisen istuntokerran tietoa ole saatavana, siirretään osallistuja uusimman aikataulun sivulle.
- Jos osallistujalla ei ole yhtään aikataulua valittavissa, siirtää järjestelmä hänet aikataulusivulle, joka näyttää muuten tyhjällä aikataulusivullaan varoituksen tilanteesta.
- Rekisteröityneen käyttäjän tiedot tallennetaan `Users`-tauluun.

3.1.10 Järjestelynäyttö

Järjestäjä voi seurata, mitä tapaamisaikoja järjestelmä ehdottaa. Hän voi muokata aikataulun tapaamisaikoja, jolloin tapaamisaikojen muokkaus on vain hänen hallinnassa. Tämän jälkeen järjestäjä voi vapauttaa aikataulun kaikkien muokattavaksi. Tapaamisajat näytetään järjestelynäytöllä taulukossa. Jokaisella osallistujalla voi olla vain yksi tapaamisaika ehdotettuna. Aikataulun kesto viikkoina näytetään taulukon vieressä olevassa kalenterissa. Kyseinen kalenteri ei ole muutettavissa. Järjestelynäytön komponenttien selitykset ovat erikseen nähtävillä näytöllä.

Näytettävä aikataulu haetaan id -parametrin avulla tietokannan Calendars -taulusta. Tieto siitä onko aikataulu vain järjestäjän muokattavissa saadaan saadaan Calendars -taulun locked -kentästä. Tarjotut ajat haetaan aikataulun tunnuksen perusteella tietokannan Offers -taulusta ja osallistujan ilmoittama sopivuus Matches -taulusta. Osallistujien nimet löytyvät p -parametrin avulla Users -taulun monikoista, joihin viitataan p -parametrin osoittamien members -taulun monikoista.

Osajärjestelmä tarjoaa järjestäjälle seuraavat palvelut.

Palvelu	Kuvaus
Varausehdotuksen tarjoaminen	Järjestelmä antaa ehdotuksen tapaamisaikojen varauksille.
Varausten uudelleen järjestely	Järjestäjä voi aloittaa varattujen tapaamisaikojen manuaalisen järjestämisen. Tällöin aikataulu lukittuu vain hänen käyttöönsä.
Aikataulun vapautus	Järjestäjä voi vapauttaa aikataulun osallistujien ja järjestäjän käyttöön, mikäli se on hänelle lukittu.
Osallistujien järjestely	Manuaalisesti muokatun aikataulun varatut ajat järjestetään diagonaaliin siten että ensimmäinen varaus on vasemmassa yläkulmassa ja viimeinen varaus on oikeassa alakulmassa.

Osajärjestelmän kutsussa voidaan käyttää seuraavia parametreja yhteisten parametrien lisäksi.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
id	int	Kyllä	Aikataulun yksilöllinen avain tietokannassa, jos tällä avaimella ei löydy aikataulua tietokannasta tai se ei ole sisäänkirjautuneen järjestäjän ryhmään kuuluva osallistuja, siirtyy järjestelmä uuden aikataulun lisäystilaan ja näyttää järjestäjälle virheilmoituksen.
p	int	Kyllä	Viiteavain Groups-taulun tietueeseen, joka viittaa ryhmään, johon aikataulu kuuluu. Jos avain ei kuulu sisäänkirjautuneen järjestäjän ryhmälle, siirtää järjestelmä järjestäjän uuden ryhmän luontitilaan Ryhmä-sivulle ja näyttää virheilmoituksen.
matchId	Int	Ei	Viiteavain Matches -taulun tietueeseen, joka viittaa sopivuuteen, josta tehdään uusi varattu tapaamisaika. Parametrin oletusarvo on NULL.
lang	String	Kyllä	Käyttöliitymän kielikoodi, käyttää järjestelmän oletusarvona <code>DEFAULT_LANGUAGE</code> .

Osajärjestelmän toiminnallisuus.

- Istunnon voimassaolo tarkistetaan ja tarvittaessa käyttäjä siirretään etusivulle (`index.php`) sisäänkirjautumista varten.
- Varmistetaan, että parametrina annettu viiteavain kuuluu sisäänkirjautuneen järjestäjän ryhmälle.
- Jos parametrina annettu viiteavain ei viittaa järjestäjän ryhmään siirretään järjestäjä uuden ryhmän luontitilaan ja hänelle näytetään virheilmoitus.
- Jos parametrina annetulla avaimella ei löydy aikataulua, näytetään järjestäjälle virheilmoitus ja uuden aikataulun syöttö alkaa.
- Jos `matchId` -parametri on annettu, poistetaan sopivuuden omistaman osallistujan nykyinen varattu aika ja asetetaan parametrina annettu sopivuus hänen uudeksi varatuksi tapaamisajaksi. Mikäli parametri on NULL tai sitä ei ole annettu tämä kohta ohitetaan ja siirrytään sovelluslogiikassa eteenpäin.
- Tässä vaiheessa kutsutaan aikojenjakoa algoritmia.

- Järjestelynäytöllä esitetään pystysarakeissa päivät ja tarjotut kellonajat. Vaakariveillä näytetään osallistujat.
- Osallistujat näytetään aakkosjärjestyksessä, muodossa Sukunimi, Etunimi vasemmalle tasattuna.
- Näytöllä näytetään kaikki viikonpäivät. Ne viikonpäivät joilla ei ole aikataulua merkitään kapeammilla soluilla kuin aikataululliset päivät, siten että viikonpäivän nimi kahdella kirjaimella mahtuu soluun. Viikonpäivasolut määritellään CSS-tyylin `bgday` -luokalla.
- Mikäli tapaamisaikojen välillä on hyppytunteja, ne näytetään kapeammilla soluilla kuin tapaamisajalliset solut. Hyppytunneille ei merkitä kellonaikoja. Tapaamisajoille merkitään kellonajat. Tapaamisaika solut määritellään CSS-tyylin `bgtime` -luokalla.
- Osallistujan ilmoittama tapaamisajan sopivuus näytetään osallistujan ja tarjotun ajan risteyskohdassa. Sopivuus ilmaistaan CSS-tyylin `bgprefX` -luokilla, jossa X on sopivuuden nimi.
- Järjestelmän ehdottama osallistujan tapaamisaika esitetään CSS-tyylin `borReserved` -luokalla.
- Päällekkäin varatut tapaamisajat esitetään CSS-tyylin `borBooked` -luokalla.
- Järjestelmä lajittelee varatut tapaamisajat taulukkoon automaattisesti siten, että opiskelijoiden sopivuudet ja varaukset muodostavat diagonaalin. Diagonaali alkaa vasemmasta yläkulmasta, jossa on viikon ensimmäinen varattu tapaamisaika. Diagonaali päättyy oikeaan alakulmaan, jossa on viikon viimeinen varattu tapaamisaika.
- Tapaamisaika -taulukon alapuolella on "Järjestä-painike ja valintaruutu jonka selitteenä on "Sopivuudet muutettavissa, ajat varataan automaattisesti".
- "Järjestä-painike järjestää manuaalisesti vaihdetut tapaamisajat aikaisemmin kuvattuun diagonaaliin. Se on toiminnallisessa tilassa vain silloin kun sen viereinen valintaruutu on tyhjä.
- Valintaruutu on oletusarvoisesti rastitettu. Rastin pois kytkeminen tai taulukon tapaamisaikojen painaminen aloittaa tapaamisaikojen manuaalisen muokkaamisen.
- Järjestäjä aloittaa tapaamisaikojen manuaalisen muokkaamisen painamalla haluamaansa sopivuutta taulukosta tai painamalla taulukon alla olevaa valintaruutua. Tällöin aikataulu on vain hänen muokattavissa. Kun manuaalinen järjestely aloitetaan valintaruutu tyhjenee ja tietokannan Calendars -tauluun asetetaan päiväys milloin aikataulu lukkiutui.

- Järjestäjä voi vaihdella aikojenjakoalgoritmin ehdottamia tai itse valitsemiaan aikoja valitsemalla osallistujan riviltä tämän ilmoittaman sopivuuden. Uusi varaus tallennetaan välittömästi tietokantaan, kutsumalla sivun omaa sovel-luslogiikkaa ja antamalla osajärjestelmälle parametrina sopivuuden yksilöivä tunnus.
- Manuaalinen muokkaaminen asettaa "Järjestä-painikkeen toiminnalliseen ti-laan.
- Järjestäjä voi lopettaa aikataulun muokkaamisen samasta valintaruudusta kuin sen aloittamisenkin tai poistamalla järjestelynäytöltä. Kun valintaruutuun kytketään takaisin rasti palautuu varaukset aikojenjakoalgoritmin laskemaan tilaan. Muuten järjestäjän valinnat jäävät voimaan. Muokkaamisen lopettami-nen vapauttaa aikataulun osallistujien ja järjestelmän muokattavaksi. Tällöin myös tietokannasta poistetaan lukituksen päiväys. Mikäli järjestäjä poistuu järjestelynäytöltä jää lukitus voimaan.
- Järjestelynäytön komponenttien selitykset ovat erikseen nähtävillä näytöllä taulukon alla.

3.1.11 Osallistujan viikkonäyttö - match.php

Osallistuja pääsee järjestelmään sähköpostilla saamansa linkin kautta. Kun järjes-telmä on tarkistanut sisäänkirjautumisen, osallistuja pääsee merkitsemään sopivuu-tensa järjestäjän tarjoamiin aikoihin oman viikkonäytön kautta. Viikkonäytön ve-tolaatikoista voi merkitä kyseisen ajan sopivuuden itselleen. Jokainen merkintä tal-lennetaan tietokantaan välittömästi merkinnän jälkeen.

Palvelu	Kuvaus
Viikkonäyttö	Viikkonäyttö sisältää vetolaati- kon sopivuuksien valitsemiseksi niiden aikojen kohdalla, jotka jär- jestäjä on merkinnyt tarjolle.
Kalenteri	Kalenteri näyttää korostetulla taustalla ne, viikot joille aikatau- lu on laadittu.
Laskuri	Ilmoittaa, montako aikojen sopi- vuutta osallistuja on merkinnyt sekä kaikkien järjestäjän tarjoa- mien aikojen lukumäärän.

Osajärjestelmän kutsussa voidaan käyttää seuraavia parametreja yhteisten paramet-rien lisäksi.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
link	String	Ei	Osallistujan kirjautessa sisään kirjautumislinkillä annetaan link parametri. Osallistujan yksilöllinen tunniste, joka viittaa tietokannan Members-tauluun. Tultaessa sivulle reunalistan linkillä, link parametria ei tarvita.
uid	int	Ei	Viitevain kyseisen osallistujan tietoihin Members taulussa. Jos tullaan sisäänkirjautumislinkillä, tämä kenttä on null arvoinen. Arvoa käytetään silloin kun sivulle tullaan reunalistan linkillä.
parent	int	Ei	Viiteavain Groups-taulun tietueeseen, joka viittaaaryhmään, jota osallistuja käsittelee. Jos tullaan sisäänkirjautumislinkillä, tämä kenttä on null arvoinen. Arvoa käytetään silloin kun sivulle tullaan reunalistan linkillä.
id	int	Ei	Viiteavain käsiteltävään aikatauluun. Jos tullaan sisäänkirjautumislinkillä, tämä kenttä on null arvoinen. Arvoa käytetään silloin kun sivulle tullaan reunalistan linkillä.

Osajärjestelmän toiminnallisuus:

- Tunnistaa käyttäjän linkin parametrina saadulla tunnisteella.
- Jos tunnistetta ei kyeta identifioimaan, siirretään käyttäjä etusivulle sisäänkirjautumista varten ja annetaan virheilmoitus.
- Istunnon voimassaolo tarkistetaan ja tarvittaessa siirretään käyttäjä etusivulle sisäänkirjautumista varten ja käyttäjälle annetaan virheilmoitus vanhentuneesta istunnosta.
- Jos parent-parametri on välitetty, haetaan tällä viiteavaimella ryhmä tietokannan Groups-taulusta.
- Jos parametrina saadun ryhmän viiteavaimen perusteella ei löydy ryhmää tietokannasta, annetaan virheilmoitus. Näytetään muuten tyhjä näyttö.
- Jos id-parametri on välitetty, haetaan tällä viiteavaimella aikataulu tietokannan Calendars-taulusta.

- Jos parametrina saadun aikataulun viiteavaimen avulla ei löydetty yhtään aikataulua annetaan osallistujalle varoitus ja ladataan näytölle tyhjä viikkonäyttö.
- Jos järjestäjä ei ole tarjonnut yhtään aikaa tai ei ole merkinnyt yhtään viikkoa kalenterista, niin osallistujalle annetaan varoitus ja ladataan näytölle tyhjä viikkonäyttö.
- Sopivuuksia merkitsevät sanojen kieli vaihtuu kielivalinnan mukaiseksi.
- Osallistujan merkityksessä sopivuutensa johonkin aikataulun vetolaatikkoon, laatikon taustaväri muuttuu valitun sopivuuden mukaan.
- Osallistujan merkityksessä sopivuuden, tarkistetaan ensin tietokannasta onko kyseinen aika vielä tarjottuna. Jos ei ole, annetaan käyttäjälle virheilmoitus ja sivua päivitetään
- Osallistujan merkityksessä sopivuuden tarkistetaan onko järjestäjä lukinnut aikataulun. Jos aikataulu on lukossa, annetaan osallistujalle varoitus.
- Osallistujan merkityksessä sopivuuden, eikä sopivuuden kirjaamiseen ole estettä (kts. 2 edellistä) kirjataan sopivuus Matches-tilaan.
- Osallistuja voi muokata sopivuuksiaan, silloin kun aikataulua ei ole lukittu (nähdään calendar taulun locked kentästä).
- Jos viikonlopulle ei ole tarjolla aikoja, näytetään viikonlopun sarakkeet (lauantai ja sunnuntai) otsikon levyisinä.
- Jos jollekin viikonlopun päivälle on tarjolla aikoja, näytetään kaikki viikonlopun sarakkeet normaalilevyisinä eli sisällön mukaan levennettyinä.
- Jos jollekin viikonlopun päivälle on tarjolla aikoja, mutta yhdellekään arkipäivälle ei ole aikoja tarjolla, näytetään kaikkien arkipäivien sarakkeet otsikon levyisinä.
- Jos jollekin viikonlopun ja arkipäivän päivälle on tarjolla aikoja, näytetään niiden arkipäivien (joilla ei ole tarjolla aikoja) sarakkeet sisällön leveydestä 2/3 leveydellä.
- Osallistujalle näytetään sellaisia aikatauluja, jotka liittyvät niihin ryhmiin, joissa hän on osallistujana.
- Osallistujan tullessa sivulle näytetään se viikkonäyttö, jota osallistuja on käsitellyt viimeksi.
- Jos viimeksi käsiteltyä aikataulua ei tiedetä, näytetään osallistujalle vanhin keskeneräinen aikataulu.
- Järjestäjän tarjoamien aikojen määrä haetaan tietokannan Offers-tilasta.

- Osallistujan valittua onnistuneesti ajan kasvatetaan välittömästi laskurin kokoa yhdellä.
- Osallistujalle näytetään, montako aikaa aikataulussa on tarjolla sekä omien aikojen sopivuusmerkintöjen määrä muodossa: x / y sopivuutta valitsematta, missä x on omien aikojen sopivuudet ja y tarjottujen aikojen määrä.

3.1.12 Osallistujan sovelluslogiikka - prefer.php

Kun osallistuja merkitsee ajan sopivuuden, kutsuu osallistujan viikkonäyttö -osajärjestelmä erillistä sovelluslogiikkaa (prefer.php). Tämä sovelluslogiikka tallentaa ajan sopivuuden tietokantaan tai poistaa sen, jos osallistuja valitsee jo merkityn ajan sopivuuden uudelleen. Käyttämällä erillistä sovelluslogiikkaa ei varsinaista käyttöliittymää tarvitse muodostaa uudestaan, sillä viikkonäytön lomake lähetetään tälle erilliselle sovelluslogiikalle.

Sovelluslogiikka tarjoaa osallistujan viikkonäyttö -osajärjestelmälle seuraavat palvelut:

Palvelu	Kuvaus
Ajan sopivuuden tallennus	Osallistujan valitsema ajan sopivuus tallennetaan Matches-tauluun, jos ajan sopivuutta ei ole vielä merkitty.
Ajan sopivuuden poisto	Jos osallistuja valitsee jo merkityn ajan sopivuuden uudelleen, poistaa järjestelmä sen tietokannasta.

Sovelluslogiikan kutsussa käytetään seuraavia parametreja.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
userID	int	Kyllä	Osallistujan yksilöllinen avain tietokannassa. Avain ilmoittaa, kenen osallistujan ajan sopivuus tallennetaan. Jos osallistuja merkitsee itse sopivuutensa, on tämä avain sama kuin sisäänkirjautuneen osallistujan avain. Jos järjestäjä merkitsee ajan sopivuuden osallistujan puolesta, viittaa tämä avain osallistujaan, ei järjestäjään.
calendarID	int	Kyllä	Aikataulun yksilöllinen avain tietokannassa. Avain ilmoittaa, minkä aikataulun tarjottuun aikaan ajan sopivuus tallennetaan tai poistetaan.
weekday	int	Kyllä	Viikonpäivän numero, missä 1 = maanantai, ..., 7 = sunnuntai. Viikonpäivän numero ilmoittaa, minkä viikonpäivän tarjotusta ajasta on kyse.
period	Time	Kyllä	Viikkonäytöltä valittu aika, joka ilmoittaa, milloin aika tulee tarjolle. Tätä arvoa käytetään myös haettaessa tietokannasta poistettava aika.
pref	int	Kyllä	Sopivuuden paino, joka ilmoittaa, miten hyvin aika sopii vai sopikoo lainkaan. Jos parametrin arvo on nolla, poistetaan aiemmin merkitty ajan sopivuus tietokannasta.

Yleiskuvaus sovelluslogiikasta

Osallistujan viikkonäytön sovelluslogiikka on kuvattu alla olevalla pseudokoodilla. Suluissa olevat numerot viittaavat SQL-lauseeseen, joka on kuvattu koodin lopussa.

Alustaistunto

Lataa komponentit

Tarkista istunnon voimassaolo

Jos istunto on voimassa

 Muodosta yhteys tietokantapalvelimelle

 Jos yhteys muodostui

 Avaa tietokanta

 Jos tietokannan avaaminen onnistui

Jäsennä parametrit
 Tarkista parametrit
 Jos parametrit kelpaavat
 Hae tietokannasta tarjotun ajan avain (1)
 Jos ajan sopivuuden paino on nolla
 Poista ajan sopivuus (2)
 Muutoin
 Tallenna ajan sopivuus (3)
 Sulje tietokanta ja tietokantayhteys

SQL-lauseet

Numero	Lause
1	SELECT ID FROM Offers WHERE CalendarID = (aikataulun avain) AND Period = '(aika)' AND Weekday = (viikonpäivän numero)
2	DELETE FROM Matches WHERE OfferID = (tarjotun ajan avain) AND UserID = (osallistujan avain)
3	INSERT INTO Matches (OfferID,UserID,Pref) VALUES (arvolista)

Toiminnallisuus

- Istunnon voimassaolo tarkistetaan ja jos istunto ei ole voimassa, sovelluslogiikan suoritus keskeytyy ja suoritusta osallistujan viikkonäyttö -osajärjestelmässä jatketaan.
- Jos parametrina annettu viiteavain osallistujaan on pienempi kuin yksi, suoritus keskeytyy.
- Jos parametrina annettu viiteavain aikatauluun on pienempi kuin yksi, suoritus keskeytyy.
- Jos parametrina annettu viikonpäivän numero ei ole joukossa 1, 2, ..., 7, suoritus keskeytyy.
- Jos parametrina annettua aikaa ei ole, suoritus keskeytyy.
- Parametrien arvot tallennetaan ajan sopivuudeksi tietokantaan, jos sopivuuden paino on erisuuri kuin nolla.
- Ajan sopivuus poistetaan tietokannasta parametrien ilmoittamilla arvoilla, jos sopivuuden paino on on nolla.

3.1.13 Osallistuja - member.php

Osallistuja on usein esimerkiksi opiskelija, joka osallistuu harjoitustyökurssille. Kursseja esitetään järjestelmässä ryhmänä ja osallistuja on tämän ryhmän jäsen. Osallistuja voi olla myös henkilökuntaa, esimerkiksi kehityskeskustelua järjestettäessä. Osajärjestelmä tarjoaa järjestäjälle seuraavat palvelut:

Palvelu	Kuvaus
Osallistujan muokkaus	Järjestäjä lisää osallistujan ja muuttaa hänen tietojaan.
Osallistujien lisäys	Järjestäjä lisää useita osallistujia leikepöydältä liittämästään tekstisyötteestä.
Osallistujan poisto	Järjestäjä poistaa osallistujan. Samalla poistuvat osallistujaan viittaavat tietueet.

Osajärjestelmän kutsussa voidaan käyttää seuraavia parametreja yhteisten parametrien lisäksi.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
id	int	Kyllä	Osallistujan yksilöllinen avain tietokannassa, jos tällä avaimella ei löydy osallistujaa tietokannasta tai se ei ole sisäänkirjautuneen järjestäjän ryhmään kuuluva osallistuja, siirtyy järjestelmä uuden osallistujan lisäystilaan ja näyttää järjestäjälle virheilmoituksen.
p	int	Kyllä	Viiteavain Groups-tilin tietueeseen, joka viittaa ryhmään, johon osallistuja kuuluu. Jos avain ei kuulu sisäänkirjautuneen järjestäjän ryhmälle, siirtää järjestelmä järjestäjän uuden ryhmän luontitilaan Ryhmä-sivulle ja näyttää virheilmoituksen.

Toiminnallisuus

- Istunnon voimassaolo tarkistetaan ja tarvittaessa käyttäjä siirretään etusivulle sisäänkirjautumista varten.
- Varmistetaan, että parametrina annettu viiteavain kuuluu sisäänkirjautuneen järjestäjän ryhmälle.
- Jos parametrina annettu viiteavain ei viittaa järjestäjän ryhmään, siirretään järjestäjä uuden ryhmän luontitilaan ja hänelle näytetään virheilmoitus.
- Jos parametrina annetulla avaimella ei löydy osallistujaa, näytetään järjestäjälle virheilmoitus ja uuden osallistujan syöttö alkaa.
- Syötteiden alusta ja lopusta poistetaan välilyönnit.
- Sukunimi ei saa puuttua.
- Sukunimen enimmäispituus on 64 merkkiä.

- Etunimi ei saa puuttua.
- Etunimen enimmäispituus on 64 merkkiä.
- Sähköpostiosoitteen enimmäispituus on 320 merkkiä, jos se on annettu.
- Sähköpostiosoitteen on oltava validi, jos se on annettu (ks. sähköpostikomponentin `invalidEmail`-palvelu).
- Sähköpostiosoite ei saa puuttua, jos se on annettu (ks. myös sähköpostiosoitteen validointi).
- Käyttöliittymän kielikoodi vaihdetaan joko parametrina annettuun tai istuntoomuuttujassa olevaan.
- Kielikoodin on oltava tuettu, oletusarvo on `DEFAULT_LANGUAGE`.
- Uudelle osallistujalle ehdotetaan kielikoodiksi käyttöliittymälle valittua kieli-koodia.
- Uudelle osallistujalle luodaan yksilöllinen tunniste sisäänkirjautumislinkkiä varten.
- Jos osallistujaksi lisättävä henkilö on jonkun ryhmän järjestäjä, ei osallistujan nimeä tai sähköpostiosoitetta voi muuttaa, ellei osallistuja ole sisäänkirjautunut järjestäjä.
- Osallistujaa poistettaessa poistetaan vain viite `Members`-taulusta. Osallistujan tiedot jäävät `Users`-tauluun.
- Jäsennettävä tekstisyöte ei saa puuttua, jos osallistujia lisätään tekstisyötteestä.
- Jäsennettävää tekstisyötettä varten näytetään lomakkeella kentät vain, jos ollaan luomassa uutta osallistujaa.
- Tekstisyöte jäsennetään `CSV`-komponentin palvelulla, joka tallentaa osallistujat.
- Jäsennin palauttaa arvon, joka on suurempi kuin 0, joka on avain aakkosjärjestyksessä viimeiseen tallennettuun osallistujaan, joka näytetään lomakkeella.
- Jos jäsennin palauttaa arvon, joka on pienempi kuin 1, näytetään järjestäjälle vastaava virheilmoitus.
- Poistopainike on näkyvissä vain, jos ollaan muokkaamassa olemassa olevaa osallistujaa.
- Järjestäjä voi poistaa osallistujan.
- Järjestäjää pyydetään vahvistamaan poisto.

- Jos järjestäjä vahvistaa poiston, osallistuja poistetaan tietokannasta ja kaikki viittaukset osallistujaan poistetaan samalla.
- Poiston jälkeen alkaa uuden osallistujan syöttö.
- Uudet osallistujat tallennetaan Users-tauluun.
- Jos Users-taulussa on jo tietue samalla sähköpostiosoitteella, ei uutta tietuetta tallenneta. Käyttöliittymässä näytetään tällöin olemassa olevan osallistujan tiedot.
- Jos sähköpostiosoitetta ei ole annettu, tallennetaan osallistuja uutena tietueena.
- Osallistujan osallistuminen valittuna olevaan ryhmään tallennetaan Members-tauluun viiteavaimina.

3.1.14 Ryhmä - group.php

Ryhmä kokoaa järjestäjän laatimat aikataulut sekä ryhmään kuuluvat osallistujat. Ryhmä voi olla esimerkiksi kurssi, tuutorointitapaaminen tai kehityskeskustelu. Osajärjestelmä tarjoaa järjestäjälle seuraavat palvelut:

Palvelu	Kuvaus
Ryhmän muokkaus	Järjestäjä luo ryhmän ja muuttaa sen tietoja.
Ryhmän poisto	Järjestäjä poistaa ryhmän. Samalla poistuvat ryhmään viittaavat tietueet.

Osajärjestelmän kutsussa voidaan käyttää seuraavia parametreja yhteisten parametrien lisäksi.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
id	int	Kyllä	Ryhmän yksilöllinen avain tietokannassa, jos tällä avaimella ei löydy ryhmää tietokannasta tai se ei ole sisäänkirjautuneen järjestäjän ryhmä, siirtyy järjestelmä uuden ryhmän luontitilaan ja näyttää järjestäjälle virheilmoituksen.

Yleiskuvaus sovelluslogiikasta

Osajärjestelmän sovelluslogiikka on kuvattu alla olevalla pseudokoodilla. Suluissa olevat numerot viittaavat SQL-lauseeseen, joka on kuvattu koodin lopussa.

Alustaistunto

Lataa komponentit

Jäsennä parametrit

Tarkista kielikoodi

Tarkista istunnon voimassaolo

Jos istunto ei ole voimassa

Valmistaudu siirtymään sisäänkirjausosajärjestelmään
 Muutoin jos sisäänkirjautunut ei ole järjestäjä
 Valmistaudu siirtymään sisäänkirjausosajärjestelmään
 Muutoin
 Alusta muuttajat
 Jos järjestäjä on valinnut tallennuksen
 Tarkista syötteet
 Jos syötteet kelpaavat
 Jos ryhmän avain on nolla
 Tallenna ryhmä (1)
 Muutoin
 Päivitä ryhmä (2)
 Muutoin jos järjestäjä on valinnut poiston
 Poista ryhmä (3)
 Jos ryhmän avain ei ole nolla
 Lue ryhmän tiedot tietokannasta (4)
 Jos aikataulu löytyi
 Pura tietue muuttujiin
 Jos ollaan siirtymässä toiseen osajärjestelmään
 Sulje tietokanta
 Siirry osajärjestelmään
 Tallenna sivuhistoria
 Siivoa muuttujien arvot
 Näytä käyttöliittymä
 Terminoi osajärjestelmä

SQL-lauseet

Numero	Lause
1	INSERT INTO Groups (ID,UserID,Name,Updated) VALUES (arvolista)
2	UPDATE Groups SET Name WHERE UserID = (järjestäjän avain) AND ID = (ryhmän avain)
3	DELETE FROM Groups WHERE UserID = (järjestäjän avain) AND ID = (ryhmän avain)
4	SELECT * FROM Groups WHERE UserID = (järjestäjän avain) AND ID = (ryhmän avain)

Toiminnallisuus

- Istunnon voimassaolo tarkistetaan ja tarvittaessa käyttäjä siirretään etusivulle sisäänkirjautumista varten.
- Jos parametrina annetulla avaimella ei löydy ryhmää, näytetään järjestäjälle virheilmoitus ja uuden ryhmän syöttö alkaa.

- Syötteiden alusta ja lopusta poistetaan välilyönnit.
- Ryhmän nimi ei saa puuttua.
- Ryhmän nimen enimmäispituus on 255 merkkiä
- Käyttöliittymän kielikoodi vaihdetaan joko parametrina annettuun tai istuntomuuttujassa olevaan.
- Kielikoodin on oltava tuettu, oletusarvo on `DEFAULT_LANGUAGE`.
- Ryhmän viimeisin päivitysaika päivitetään uutta ryhmään tallennettaessa.
- Poistopainike on näkyvä vain, jos ollaan muokkaamassa olemassa olevaa ryhmää.
- Järjestäjä voi poistaa ryhmän.
- Järjestäjää pyydetään vahvistamaan poisto.
- Jos järjestäjä vahvistaa poiston, ryhmä poistetaan tietokannasta ja kaikki viittaukset ryhmään poistetaan samalla.
- Poiston jälkeen alkaa uuden ryhmän syöttö.
- Ryhmän tiedot tallennetaan Groups-tauluun.
- Järjestäjän ryhmien nimet ovat uniikkeja. Tietokannassa voi olla samannimisiä ryhmiä, mutta tällöin niiden oltava eri järjestäjien.

3.1.15 Uloskirjautuminen - `sign_out.php`

Käyttäjä kirjautuu ulos järjestelmästä valitsemalla Kirjautu ulos -linkin millä tahansa sivulla. Osajärjestelmä tarjoaa käyttäjälle seuraavat palvelut:

Palvelu	Kuvaus
Uloskirjautuminen	Käyttäjä kirjautuu ulos järjestelmästä ja hänen istuntonsa päättyy. Järjestelmä siirtää käyttäjän takaisin etusivulle.

Osajärjestelmä ei käytä omia parametreja.

Yleiskuvaus sovelluslogiikasta

Osajärjestelmän sovelluslogiikka on kuvattu alla olevalla pseudokoodilla.

Nollaa istuntomuuttujat

Tuhoa istunto

Siirry sisäänkirjausosajärjestelmään

Toiminnallisuus

- Päättää istunnon tuhoamalla istunnon ja sen muuttajat.
- Siirtää käyttäjän etusivulle.

3.1.16 Viesti - message.php

Järjestäjä lähettää viestiosajärjestelmällä osallistujille kutsuja merkitä aikojen sopivuutensa sekä ilmoituksia aikataulun valmistumisesta. Järjestäjä voi muokata viestin tekstiä. Muokattu teksti tallentuu tietokantaan samoin kuin viestin otsikkokin. Viestin vastaanottajat valitaan ryhmän osallistujista. Viesti lähetetään vain niille osallistujille, joiden sähköpostiosoite on tiedossa.

Viestissä lähetetään myös osallistujaa koskevaa tietoa. Koska tämä tieto vaihtuu osallistujasta eli viestin vastaanottajasta toiseen, ei sitä ole kirjoitettu viestiin sellaisenaan. Tällainen vaihtuva tieto merkitään erityisillä elementeillä viestissä, jotka järjestelmä korvaa osallistujakohtaisella tiedolla juuri ennen viestin lähetystä. Alla olevassa luettelossa on lueteltu käytettävät elementit.

Elementti	Kuvaus
<linkki_sopivuussivulle/>	Elementti ilmoittaa yksilöllisen sisäänkirjautumislinkin sijainnin tekstissä. Tämä elementti korvautuu osallistujan tietoihin tallennetulla tunnisteella, jota käytetään parametrina sisäänkirjautumislinkeissä. Varsinainen osoite sivulle on valmiina viestipohjassa ja ainoastaan osoitteen loppuun lisättävä parametri liitetään lähetyksessä.
<aikataulu_aika/>	Elementti ilmoittaa aikataulu-nimisestä aikataulusta varatun tapaamisajan. Elementti muodostuu aikataulun nimestä (aikataulu) sekä merkinnästä _aika. Jos järjestäjä on laatinut esimerkiksi kaksi aikataulua, on sivulla elementit molemmille aikatauluille. Mikäli aikataulusta on varattu useita aikoja, näytetään nämä kaikki varaukset yhdellä linkillä siten, että kukin varaus on omalla rivillään. Aikataulun nimessä oleva mahdollinen välilyönti konvertoidaan alaviivaksi ja nimi muutetaan gemenaksi.

Esimerkki: <henkilökohtainen_tapaaminen_aika/>

Tapaamisaika esitetään viestissä selväkielisenä. Jos varattuja aikoja on useilla eri viikoilla, mutta aina samana viikonpäivänä, näytetään aika muodossa:

Suomeksi	Englanniksi
tiistaisin 13.9.2005 - 11.10.2005 klo 13:20 - 13:40	Tuesdays on 13.9.2005 - 11.10.2005 at 13:20 - 13:40

Kun varattuja aikoja on useilla viikoilla ja eri viikonpäivinä, näytetään aika muodossa:

Suomeksi	Englanniksi
tiistaisin 13.9.2005 - 11.10.2005 klo 13:20 - 13.40	Tuesdays on 13.9.2005 - 11.10.2005 at 13:20 - 13:40
torstaisin 15.9.2005 - 13.10.2005 klo 11:40 - 12.00	Thursdays on 15.9.2005 - 13.10.2005 at 11:40 - 12:00

Kertaluonteiselle tapaamiselle varattu aika näytetään muodossa:

Suomeksi	Englanniksi
torstaina 29.9.2005 klo 8:00 - 10:00	on Thursday 29.9.2005 at 8:00 - 10:00

Monikielisyyden tuki kokoaa ilmaiset kahdesta eri tekstirungosta, joista alkuosassa on viikonpäivä ja loppuosassa varattu aika. Jälkimmäisessä tekstirungossa on kaksi parametria, joista ensimmäinen korvautuu päivämäärällä tai päivämäärävälillä ja toinen kellonajalla. Alla olevassa luettelossa on oletustekstirungot.

Nimi	Kielikoodi	Arvo	Kielikoodi	Arvo
DNA1	fi	maanantaina	en-US	on Monday
DNA2	fi	tiistaina	en-US	on Tuesday
DNA3	fi	keskiviikkona	en-US	on Wednesday
DNA4	fi	torstaina	en-US	on Thursday
DNA5	fi	perjantaina	en-US	on Friday
DNA6	fi	lauantaina	en-US	on Saturday
DNA7	fi	sunnuntaina	en-US	on Sunday
DIN1	fi	maanantaisin	en-US	Mondays
DIN2	fi	tiistaisin	en-US	Tuesdays
DIN3	fi	keskiviikkoisin	en-US	Wednesdays
DIN4	fi	torstaisin	en-US	Thursdays
DIN5	fi	perjantaisin	en-US	Fridays
DIN6	fi	lauantaisin	en-US	Saturdays
DIN7	fi	sunnuntaisin	en-US	Sundays
WHEN	fi	%0 klo %1	en-US	%0 at %1

Alla oleva malli kuvaa järjestelmän toimintaa viestiä lähetettäessä:

Käy läpi kaikki viestin saavat osallistujat Käy läpi kaikki ryhmään kuuluvat aikataulut Jos viestin tekstissä on aikataulun elementti Käy läpi viikonpäivät maanantaista sunnuntaihin Hae tietokannasta ensimmäinen ja viimeinen aikataulun viikko, jossa varaus osallistujalle Jos osallistujalle on varattu aika Käy läpi osallistujalle varatut ajat vuorossa olevana viikonpäivänä Jos ensimmäinen viikko == viimeinen viikko Korvaa elementti päivällä Muutoin Korvaa elementti ajanjaksolla

Osajärjestelmä tarjoaa järjestäjälle seuraavat palvelut:

Palvelu	Kuvaus
Viestin muokkaus	Järjestäjä muokkaa viestin tekstiä ja tallentaa sen tietokantaan.
Viestin lähetys	Järjestäjä valitsee viestin vastaanottajat ja lähettää viestin.
Kutsu	Järjestäjä lähettää valituille ryhmän osallistujille kutsun merkitä aikojen sopivuutensa.
Ilmoitus	Järjestäjä lähettää valituille ryhmän osallistujille ilmoituksen aikataulun valmistumisesta ja varatuista ajoista.

Osajärjestelmän kutsussa voidaan käyttää seuraavia parametreja yhteisten parametrien lisäksi.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
id	String	Kyllä	Ilmaisee viestin tyyppin, joita ovat announcement, invitation, missä announcement = ilmoitus ja invitation = kutsu. Jos parametri puuttuu, järjestelmä valitsee oletukseksi kutsun (invitation).
p	int	Kyllä	Viiteavain Groups-tilin tietueeseen, joka viittaa viestin omistavaan ryhmään. Jos avain ei kuulu sisäänkirjautuneen järjestäjän ryhmälle, siirtää järjestelmä järjestäjän uuden ryhmän luontitilaan Ryhmä-sivulle ja näyttää virheilmoituksen.

Toiminnallisuus

- Istunnon voimassaolo tarkistetaan ja tarvittaessa käyttäjä siirretään etusivulle sisäänkirjautumista varten.
- Viestin tyyppi välitetään parametrina osoitteessa ja siitä otetaan 16 ensimmäistä merkkiä.
- Viestin tyyppi konvertoidaan gemenaksi.
- Jos viestin tyyppi ei ole mikään sallituista arvoista, oletetaan viestin tyyppi kutsu.
- Viestin tyyppi tallennetaan tietokantaan versaalina.
- Varmistetaan, että parametrina annettu viiteavain kuuluu sisäänkirjautuneen järjestäjän ryhmälle.
- Jos parametrina annettu viiteavain ei viittaa järjestäjän ryhmään, siirretään järjestäjä uuden ryhmän luontitilaan ja hänelle näytetään virheilmoitus.
- Jos parametrina annetulla viestin tyyppillä ei löydy tallennettua viestiä, näytetään järjestäjälle oletusviestipohja.

- Viestin vastaanottajalista alustetaan ryhmän osallistujilla.
- Osallistuja näytetään listoilla muodossa Sukunimi, Etunimi (sähköpostiosoitte).
- Osallistujat lajitellaan aakkosjärjestykseen.
- Jos osallistujalla ei ole sähköpostiosoitetta, ei näytetä sulkujakaan.
- Jos osallistujalla ei ole sähköpostiosoitetta, ei häntä voida siirtää Viestin saavat -listalle.
- Järjestäjä valitsee viestin vastaanottajat Viestin saavat -listalta, monivalinta on mahdollista.
- Järjestäjä valitsee osallistujat Viestittä jäävät -listalta, monivalinta on mahdollista.
- Järjestäjä lisää valitsemansa osallistujat Viestin saavat -listalle, jolloin osallistujat siirtyvät Viestittä jäävät -listalta.
- Järjestäjä poistaa valitsemansa vastaanottajat Viestin saavat -listalta, jolloin vastaanottajat siirtyvät Viestittä jäävät -listalle.
- Listojen korkeus on $\max(|\text{Viestittä jäävät}|, |\text{Viestin saavat}|)$.
- Syötteiden (paitsi viestin tekstin) alusta ja lopusta poistetaan välilyönnit.
- Viestin otsikko ei saa puuttua.
- Viestin otsikon enimmäispituus on 255 merkkiä
- Viestin teksti ei saa puuttua.
- Käyttöliittymän kielikoodi vaihdetaan joko parametrina annettuun tai istuntoomuuttujassa olevaan.
- Kielikoodin on oltava tuettu, oletusarvo on `DEFAULT_LANGUAGE`.
- Ryhmän viimeisin päivitysaika päivitetään uutta ryhmään tallennettaessa.
- Poistopainike on näkyvässä vain, jos ollaan muokkaamassa olemassa olevaa ryhmää.
- Järjestäjä voi poistaa ryhmän.
- Järjestäjää pyydetään vahvistamaan poisto.
- Jos järjestäjä vahvistaa poiston, ryhmä poistetaan tietokannasta ja kaikki viitaukset ryhmään poistetaan samalla.
- Poiston jälkeen alkaa uuden ryhmän syöttö.
- Muokatun viestin tiedot tallennetaan Messages-tauluun.

3.2 Tietokanta-arkkitehtuuri

Järjestelmä käyttää sekä relaatiotietokantaa että tekstitiedostoja. Relaatiotietokantana voidaan käyttää ANSI SQL-92 -standardin mukaista tietokantaa, sillä järjestelmän ja tietokannan välillä on yleiskäyttöinen rajapinta eli tietokantakomponentti. Komponentti voidaan ohjelmoida kullekin käytettävälle tietokantapalvelimelle erikseen. Näin järjestelmää ei sidota tiettyyn tietokantaan ja se voidaan ottaa käyttöön useammilla eri laite- ja varusohjelmistoalustoilla.

Vakiomuotoiset sähköpostiviestien pohjatekstit ovat tekstitiedostoissa. Kun järjestäjä valitsee jonkin sähköpostiviestin, tarjoaa järjestelmä ensin järjestäjän aiemmin muokkaamaa viestiä. Jos viestiä ei ole muokattu, tarjoaa järjestelmä tekstitiedostossa olevaa vakiopohjaa. Järjestelmä ei tee muutoksia vakiopohjaan vaan tallentaa muokatun tekstin ja otsikon tietokantaan. Tallennetut viestit ovat ryhmän aliluokkia ja liittyvät siten aina johonkin järjestäjän luomaan ryhmään. Tietokannassa voi olla samanaikaisesti useita muokattuja sähköpostiviestejä, joita ovat muokanneet useat eri järjestäjät. Muokatut viestit poistuvat tietokannasta, kun ryhmä poistetaan joko käsin tai automaattisesti.

Tietokantasuunnittelussa on pyritty minimoimaan tietokantasidonnaisten ominaisuuksien käyttöä. Järjestelmä tarkistaa kenttien sallitut arvot ennen niiden tallentamista tietokantaan, joten vastaavia tietokantapalvelimen tarjoamia ominaisuuksia ei käytetä.

3.3 Viite-eheys

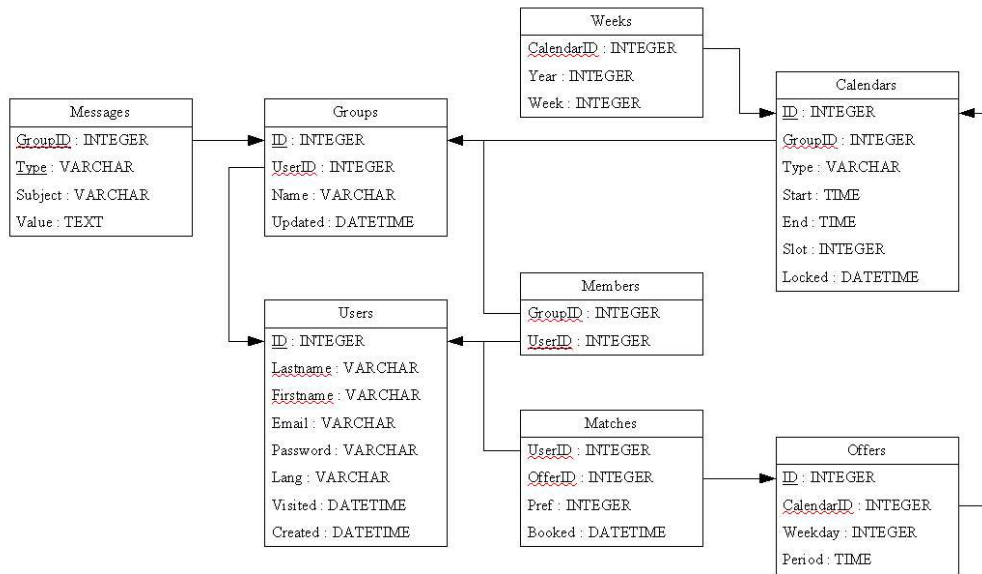
Kun järjestelmä poistaa tietokannasta tietueita, poistuvat samalla kaikki ne tietueet, jotka viittaavat poistettavaan tietueeseen.

3.4 Tietokantakuvaus

Taulujen kuvauksissa ilmoitetaan sarakkeen nimen, tyyppin ja pituuden lisäksi, ovatko NULL-arvot sallittuja. NULL-arvo esitetään käyttäjälle pääsääntöisesti tyhjänä merkkijonona tai kokonaislukuna 0. Taulujen avaimista ja indekseistä ilmoitetaan indeksoitu sarake ja onko se yksilöivä eli uniikki. Viiteavaimista ilmoitetaan myös taulu, johon viitataan. Kaikki avaimet ovat nollaa suurempia. Jos avaimen arvo on nolla, on kyseessä uusi tietue eikä tällaista tietuetta ole vielä tallennettu tietokantaan.

Alla olevassa kaaviokuvassa on kuvattu tietokantakaavio:

Järjestelmä käyttää kuvan 6 mukaisia aputauluja koodien ja monikielisyiden tuen tietojen tallentamiseen.



Kuva 5: Tietokantakaavio.

Codes	Values	Locals
<u>Collection</u> : VARCHAR	<u>Collection</u> : VARCHAR	<u>Name</u> : VARCHAR
<u>Name</u> : VARCHAR	<u>Name</u> : VARCHAR	<u>Lang</u> : VARCHAR
Value : TEXT	<u>Lang</u> : VARCHAR	Value : TEXT
Sorting : INTEGER	Value : TEXT	
	Sorting : INTEGER	

Kuva 6: Aputaulut

3.4.1 Calendars - taulu

Calendars-aulussa ovat aikataulut niin henkilökohtaisille kuin ryhmätapaamisillekin. Aikataulun tyypit ovat kielisidonnaisia, sillä tyyppien arvot ovat esillä käyttöliittymässä.

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
ID	INTEGER		Ei	Yksilöllinen avain
GroupID	INTEGER		Ei	Viiteavain Groups-tauluun
Name	VARCHAR	255	Ei	Kalenterin nimi
Type	VARCHAR	16	Ei	Aikataulun tyyppi, jonka sallitut arvot ovat PERSONAL, GROUP, missä PERSONAL = henkilökohtainen tapaaminen ja GROUP = ryhmätapaaminen
Start	TIME		Ei	Viikon aloitusaika
End	TIME		Ei	Viikon lopetusaika
Slot	INTEGER		Ei	Tuntijako minuutteina
Locked	DATETIME		Kyllä	Aikaleima, jolloin järjestäjä aloitti aikataulun manuaalisen järjestyksen, kun manuaalinen järjestely päätetään, asetetaan arvoksi NULL

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
ID	Pääavain		
GroupID	Viiteavain		Groups

3.4.2 Calendar_History-taulu

Calendar_History-taulussa on tieto osallistujan viimeksi käsittelemästä aikataulusta. Järjestelmä tallentaa tähän tauluun viiteavaimen Users-tauluun sekä avaimen aikatauluun, kun osallistuja ryhtyy merkitsemään aikojen sopivuuksiaan. Kun osallistuja seuraavan kerran ryhtyy merkitsemään aikojen sopivuuksiaan eikä osajärjestelmää kutsuttaessa ole annettu aikataulun avainta, hakee järjestelmä avaimen Calendar_History-taulusta, mikäli sellainen löytyy.

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
UserID	INTEGER		Ei	Viiteavain Users-tauluun
CalendarID	INTEGER		Ei	Viiteavain Calendars-tauluun

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
UserID, CalendarID	Pääavain		
UserID	Viiteavain		Users
CalendarID	Viiteavain		Calendars

3.4.3 Codes -taulu

Codes-taulussa on järjestelmän vaatimat koodit, jotka eivät ole kielisidonnaisia. Koodit ovat järjestelmän käyttämiä arvoja, joita ei haluta ohjelmoida itse ohjelmakoodiin. Esimerkiksi kielikoodit, joita järjestelmän monikielisyys tuki tunnistaa, luetellaan LANGUAGE-kokoelmassa. Koodit kootaan ryhmiksi, joista käytetään nimitystä kokoelma. Esimerkiksi järjestäjille sallitut sähköpostiosoitteiden toimialueet on koottu DOMAIN-nimiseen kokoelmaan. Koodit voidaan esittää käyttöliittymässä halutussa järjestyksessä, joka on tallennettu Sorting-kenttään.

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
Collection	VARCHAR	16	Ei	Nimi kokoelmalle, joka kokoaa yhteenkuuluvat koodit ryhmäksi
Name	VARCHAR	16	Ei	Koodin nimi
Value	TEXT		Ei	Koodin arvo
Sorting	INTEGER		Kyllä	Lajittelujärjestys, jolla tietyn kokoelman koodit esitetään käyttöliittymässä

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
Collection,Name	Pääavain		

3.4.4 Groups -taulu

Groups-taulussa on ryhmät, jotka kokoavat järjestäjän laatimat aikataulut sekä ryhmän osallistujat. Ryhmä voidaan käsittää esimerkiksi kurssina, kehityskeskusteluna tai muuna tapaamisena.

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
ID	INTEGER		Ei	Yksilöllinen avain
UserID	INTEGER		Ei	Viiteavain Users-tauluun
Name	VARCHAR	255	Ei	Ryhmän nimi
Updated	DATETIME		Ei	Ryhmän tietojen viimeisin muutospäivä, ks. automaattinen poisto.

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
ID	Pääavain		
UserID	Viiteavain		Users

3.4.5 Locales -taulu

Locales-aulussa on monikielisyyden tuen vaatimat tekstirungot. Jokainen tekstirun-ko tunnistetaan yksilöllisellä nimellä sekä kielikoodilla. Kielikoodi sisältää varsinaisen kielikoodin lisäksi mahdollisen maakoodin väliviivalla erotettuna. Tuetut kieli-koodit luetellaan Codes-aulussa. Alla olevassa taulussa on lueteltu oletuskielikoodit. Lajittelujärjestyksellä ilmaistaan käyttöliittymälle, missä järjestyksessä kieli-valinnat näytetään. Katso myös erillinen kuvaus monikielisyyden tuen komponentista.

Kokoelma	Nimi	Arvo	Lajittelujärjestys
LANGUAGE	fi	suomeksi	1
LANGUAGE	en-US	In English	2

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
Name	VARCHAR	4	Ei	Tekstirungon yksilöllinen nimi
Lang	VARCHAR	5	Ei	Kielikoodi
Value	TEXT		Ei	Arvo eli tekstirun-ko

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
Name, Lang	Pääavain		

3.4.6 Matches -taulu

Matches-auluun tallennetaan osallistujien merkitsemät aikojen sopivuudet. Tiedon puuttuminen tietokannasta tarkoittaa, ettei osallistujan sopivuus ole tiedossa. Algoritmi tai järjestäjä varaa ajan, jolloin Booked-kenttään tallennetaan aikaleima merkiksi ajan varauksesta. Jos kentän arvo on NULL, merkitsee se ettei kyseistä aikaa ole varattu.

Sopivuusarvot on tallennettu Values-auluun. Sopivuudet ovat kielisidonnaisia, sillä sopivuuden selväkielinen esitysmuoto riippuu valitusta käyttöliittymän kielestä. Alla olevassa taulussa on lueteltu oletussopivuudet. Nimi on sama kuin sopivuuden paino ja on siis sama kuin Matches-aulun Pref-kentän arvo. Lajittelujärjestys ilmoittaa, missä järjestyksessä arvot esitetään käyttöliittymän lomakkeella.

Aikojen järjestelyosajärjestelmän on tiedettävä, mikä sopivuuksista "ei sopiva". Tällöin järjestäjä ei voi siirtää varausta tällaiselle ajalle. Ei sopiva -sopivuus on tallennettu Codes-kooditaulun PREF-kokoelmaan NOT_SUITABLE-nimellä ja sen arvo on tallennettu ei sopivan -sopivuuden arvo.

Kokoelma	Nimi	Kielikoodi	Arvo	Kielikoodi	Arvo	Lajittelu- järjestys
PREF	0	fi	ei tiedossa	en-US	unknown	5
PREF	1	fi	erinomaisesti	en-US	perfectly	1
PREF	2	fi	hyvin	en-US	well	2
PREF	3	fi	välttävästi	en-US	fairly	3
PREF	4	fi	ei sovi	en-US	not suitable	4

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
UserID	INTEGER		Ei	Viiteavain Users-tauluun
OfferID	INTEGER		Ei	Viiteavain Offers-tauluun
Pref	INTEGER		Ei	Ajan sopivuus, jonka sallitut arvot ovat 1, 2, 3, 4
Booked	DATETIME		Kyllä	Aikaleima, jolloin algoritmi tai järjestäjä on varannut ajan osallistujalle tai NULL, jos aikaa ei ole varattu

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
UserID, OfferID	Pääavain		
UserID	Viiteavain		Users
OfferID	Viiteavain		Offers

3.4.7 Messages-taulu

Messages-tauluun tallennetaan järjestäjän muokkaamat sähköpostiviestit.

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
GroupID	INTEGER		Ei	Viiteavain Groups-tauluun
Type	VARCHAR	16	Ei	Viestin tyyppi, jonka sallitut arvot ovat INVITATION, ANNOUNCEMENT, missä INVITATION = kutsu ja ANNOUNCEMENT = ilmoitus
Subject	VARCHAR	255	Kyllä	Viestin otsikko
Value	TEXT		Kyllä	Viestin teksti

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
GroupID, Type	Pääavain		
GroupID	Viiteavain		Groups

3.4.8 Offers-taulu

Offers-taulussa ovat tarjotut ajat. Ajan puuttuminen tietokannasta merkitsee, ettei kyseisenä aikana ole tapaamisaikaa tarjolla.

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
ID	INTEGER		Ei	Yksilöllinen avain
CalendarID	INTEGER		Ei	Viiteavain Calendars-tiluun
Weekday	INTEGER		Ei	Viikonpäivän numero, jonka sallitut arvot ovat 1, 2, 3, 4, 5, 6, 7, missä 1 = maanantai, 2 = tiistai, ..., 7 = sunnuntai
Period	TIME		Ei	Tarjotun ajan aloitusaika

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
ID	Pääavain		
CalendarID	Viiteavain		Calendars

3.4.9 Users -taulu

Users-tiluun tallennetaan rekisteröityneiden käyttäjien sekä ryhmien osallistujien tiedot. Käyttäjä voi halutessaan valita käyttöliittymän kielikoodin tai järjestäjä valitsee osallistujalle kielikoodin valmiiksi. Kielikoodi tallennetaan Lang-kenttään, ja se sisältää varsinaisen kielikoodin lisäksi mahdollisen maakoodin väliviivalla erotettuna.

Käyttäjä on Järjestäjän roolissa, mikäli hänen sähköpostiosoitteensa kuuluu ennalta määriteltyihin toimialueisiin. Jos osoite ei kuulu järjestäjille sallittuihin toimialueisiin, on käyttäjä Osallistujan roolissa. Kun järjestäjä lisää osallistujia ryhmään, tallentuu näistä tieto erilliseen Members-tiluun viiteavaimina. Toimialueet luetellaan Codes-tilussa. Alla olevassa luettelossa on oletustoimialueet:

Kokoelma	Nimi	Arvo	Lajittelujärjestys
DOMAIN	CS	cs.helsinki.fi	
DOMAIN	HELSINKI	helsinki.fi	

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
ID	INTEGER		Ei	Yksilöllinen avain
Lastname	VARCHAR	64	Ei	Käyttäjän sukunimi
Firstname	VARCHAR	64	Ei	Käyttäjän etunimi
Email	VARCHAR	320	Ei	Käyttäjän sähköpostiosoite
Password	VARCHAR	32	Ei	Käyttäjän salasana
GUID	VARCHAR	32	Ei	Osallistujan yksilöllinen tunnistus sisäänkirjautumislinkkiä varten
Lang	VARCHAR	5	Kyllä	Käyttäjän valitsema käyttöliittymän kielikoodi
Visited	DATETIME		Kyllä	Käyttäjän viimeisin sisäänkirjautumisaika, ks. automaattinen poisto
Created	DATETIME		Ei	Käyttäjän rekisteröitymispäivämäärä, ks. automaattinen poisto

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
ID	Pääavain		
Email	Indeksi	Kyllä	
GUID	Indeksi	Kyllä	

3.4.10 Values -taulu

Values-taulussa on järjestelmän vaatimat, kielisidonnaiset koodit. Näitä koodeja käytetään vastaavalla tavalla kuin Codes-taulun koodeja, mutta jokaiselle tuetulle kielelle on lokalisoitu versio koodista. Kielikoodi sisältää varsinaisen kielikoodin lisäksi mahdollisen maakoodin väliviivalla erotettuna.

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
Collection	VARCHAR	16	Ei	Nimi kokoelmalle, joka kokoaa yhteenkuuluvat koodit ryhmäksi
Name	VARCHAR	16	Ei	Koodin nimi
Lang	VARCHAR	5	Ei	Kielikoodi
Value	TEXT		Ei	Koodin arvo
Sorting	INTEGER		Kyllä	Lajittelujärjestys, jolla tietyn kokoelman koodit esitetään käyttöliittymässä

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
Collection, Name, Lang	Pääavain		

3.4.11 Weeks-taulu

Aikataulun kesto viikkoina ilmaistaan Weeks-taulun tietueilla. Järjestäjä valitsee viikot, jolloin tapaamisia järjestetään. Viikon puuttuminen tietokannasta merkitsee, ettei kyseisellä viikolla ole tapaamisia. Aikataulu voi ulottua vuodenvaihteen yli.

Taulun kentät:

Nimi	Tyyppi	Pituus	NULL sallittu	Kuvaus
CalendarID	INTEGER		Ei	Viiteavain Calendars-tilaan
Year	INTEGER		Ei	Vuosiluku, jota käytetään osoittamaan aikataulun kesto vuodenvaihteen yli
Week	INTEGER		Ei	Viikon numero, jolloin tapaamisia järjestetään

Taulun avaimet ja indeksit:

Sarake	Tyyppi	Yksilöivä	Taulu
CalendarID, Year, Week	Pääavain		
CalendarID	Viiteavain		Calendars

3.5 Tiedostokuvaus

Sähköpostiviestien pohjatekstitiedostot ovat tekstitiedostoja, joissa on pelkkä viestipohja. Sähköpostiviestin otsikkoa ei ole tallennettu tiedostoon vaan järjestäjä kirjoittaa sen viestikohtaisesti. Tiedostoille asetetut vaatimukset on lueteltu alla olevassa luettelossa:

Attribuutti	Arvo	Kuvaus
Kansio	/res	Tekstitiedostot sijoitetaan juurikansion alle luotuen res-nimiseen kansioon, josta käytetään dokumentoinnissa nimitystä resurssikansio.
Nimet	gemena	Kansioiden ja tiedostojen nimet kirjoitetaan pienillä kirjaimilla.
Tarkenne	.txt	Tiedoston tarkenne on txt.
Koko	1 000 000 tavua	Tiedoston enimmäiskoon oletusarvo on 1 000 000 tavua. Koko voidaan muuttaa sähköpostikomponentin ohjelmakoodia muuttamalla.
Merkistö	ISO 8859-1	Tiedostojen tekstien suositellaan noudattavan ISO 8859-1-standardia.

Järjestelmä käyttää seuraavia tekstitiedostoja:

Nimi	Tiedosto	Kuvaus
Ilmoitus	announcement.txt	Osallistujille lähetettävä ilmoitus aikataulujen valmistumisesta
Kutsu	invitation.txt	Osallistujille lähetettävä kutsu
Rekisteröityminen	signup.txt	Järjestelmä lähettää käyttäjälle salasanan rekisteröitymisen jälkeen.
Salasana	password.txt	Järjestelmä lähettää käyttäjälle unohtamansa salasanan, joka on tallennettu Users-tauluun.
Valmius	ready.txt	Järjestelmä lähettää automaattisesti ilmoituksen järjestäjälle, kun kaikki osallistujat ovat merkinneet aikojen sopivuutensa.

3.5.1 Ilmoitus-tiedosto

Järjestäjä lähettää osallistujille ilmoituksen valmistuneista aikatauluista muokkamalla tarvittaessa Ilmoitus-tiedostoa. Järjestäjä lisää viestin tekstiin elementtejä, jotka ilmaisevat järjestelmälle, mihin kohtaan viestiä lisätään tieto varatuista ajoista. Elementin nimi kostuu aikataulun nimestä ja "_aika-loppuliitteestä. Järjestäjä kirjoittaa viestin otsikon.

Esimerkki elementistä: <henkilökohtaiset_tapaamiset_aika/>

3.5.2 Kutsu-tiedosto

Järjestäjä kutsuu osallistujat merkitsemään omat aikojen sopivuutensa lähettämän Kutsun. Järjestäjä voi halutessaan muokata viestin tekstiä lisäämällä elementtejä, jotka ilmaisevat järjestelmälle, mihin kohtaan viestiä lisätään osallistujan yksilöllinen tunniste sisäänkirjautumislinkkiä varten. Elementin nimi on <henkilökohtainen_linkki/>. Järjestäjä kirjoittaa viestin otsikon.

3.5.3 Rekisteröityminen-tiedosto

Kun käyttäjä on rekisteröitynyt järjestelmään, lähettää se käyttäjän ilmoittamaan sähköpostiosoitteeseen generoidun salasanan seuraavia sisäänkirjautumisia varten.

Salasanan sijainti tekstissä ilmoitetaan <password/>-elementillä. Järjestelmä käyttää viestin otsikkona oletusotsikkoa:

Nimi	Kielikoodi	Arvo
YPWD	fi	Salasanasi Aija-järjestelmään – Your password to Aija system
YPWD	en-US	Your password to Aija system – Salasanasi Aija-järjestelmään

3.5.4 Salasana-tiedosto

Kun käyttäjä pyytää järjestelmää lähettämään salasanansa, lähettää järjestelmä sen käyttämällä Salasana-tiedoston tekstiä viestissä. Salasanan sijainti tekstissä ilmoitetaan <password/>-elementillä. Järjestelmä käyttää viestin otsikkona oletusotsikkoa:

Nimi	Kielikoodi	Arvo
YPWD	fi	Salasanasi Aija-järjestelmään – Your password to Aija system
YPWD	en-US	Your password to Aija system – Salasanasi Aija-järjestelmään

3.5.5 Valmius-tiedosto

Kun kaikki osallistujat ovat merkinneet aikojen sopivuutensa, lähettää järjestelmä tästä tiedon järjestäjälle käyttämällä Valmius-tiedoston tekstiä viestissä. Järjestelmä täydentää viestissä olevan linkin lisäämällä siihen parametrina avaimen, joka viittaa aikatauluun. Parametrin sijainti tekstissä ilmoitetaan <id/>-elementillä. Järjestelmä käyttää viestin otsikkona oletusotsikkoa:

Nimi	Kielikoodi	Arvo
TRDY	fi	%0 aikataulu on valmiina – %0 timetable is ready
TRDY	en-US	%0 timetable is ready – %0 aikataulu on valmiina

Tekstirungossa oleva parametri korvautuu aikataulun nimellä.

4 Komponenttien sijoitusjärjestys

Osa komponenteista on riippuvaisia muiden komponenttien palveluista, esimerkiksi alustuskomponentti (initialize.php) käyttää tietokantakomponentin (db.php) palveluita. Komponentit on sijoitettava seuraavassa järjestyksessä:

1. db.php

2. email.php
3. i18n.php
4. initialize.php
5. muut komponentit

Terminointikomponentti (`terminate.php`) on aina sijoitettava osajärjestelmän loppuun, sillä tässä komponentissa suoritetaan osajärjestelmän terminointitoimenpiteet, kuten esimerkiksi tietokantayhteyden sulkeminen.

5 Alustuskomponentti

Alustuskomponentin tehtävänä on suorittaa osajärjestelmän alustustoimenpiteitä, kuten tietokantayhteyden muodostus.

5.1 Yleiskuvaus sovelluslogiikasta

Komponentin sovelluslogiikka on kuvattu alla olevalla pseudokoodilla.

```
Aseta PHP-ympäristömuuttuja
Aseta vakiot
Muodosta yhteys tietokantapalvelimelle
Jos yhteyden muodostus epäonnistui
    Valmistaudu esittämään virheilmoitus
Avaa tietokanta
Jos tietokannan avaus epäonnistui
    Valmistaudu esittämään virheilmoitus
```

5.2 initialize.php-komponentin käyttö

Skriptinä toteutettu `initialize.php`-komponentti liitetään sovelluslogiikkaan sivun alustusosassa `include-direktiivillä`.

Esimerkki: `include "inc/initialize.php";`

Koska komponentti luo yhteyden tietokantaan, on se sijoitettava `db.php`-komponentin jälkeen, jotta tietokantakomponentin tarjoamat palvelut olisivat varmasti tarjolla ennen alustustoimenpiteitä.

5.3 Vakiot

Alustuskomponentti alustaa seuraavat vakiot osajärjestelmän käytettäväksi. Nämä vakiot ovat globaaleja, jolloin kaikki osajärjestelmät voivat käyttää niitä.

Vakion nimi	Tietotyyppi	Kuvaus
DEFAULT_LANGUAGE	String	Käyttöliittymän oletuskielikoodi, jos kielikoodia ei kyetä tunnistamaan. Oletuskielikoodin on oltava jokin tuetuista kielistä, ts. oletuskielikoodilla pitää olla monikielisyyden tuki ja vastaavat tekstirungot Locales-aulussa.
PATH_BASE	String	Sivuston juurikansion osoite, joka ilmoittaa, missä kansiossa sovellus sijaitsee. Tämän vakion arvoa muuttamalla sovellus voidaan asentaa myös toisen sivuston alikansioon, esimerkiksi osoitteeseen /exercies/ohtu/. Vakion arvo alkaa ja päättyy aina kauttaviivaan.
PATH_LOG	String	Lokikansion osoite, joka ilmoittaa, missä kansiossa sijaitsevat lokitiedostot. Vakion arvo alkaa ja päättyy aina kauttaviivaan.
PATH_RES	String	Resurssikansion osoite, joka ilmoittaa, missä kansiossa sijaitsevat resurssitiedostot. Vakion arvo alkaa ja päättyy aina kauttaviivaan.
OS_WELL_BEHAVING	Boolean	Ilmoittaa, mikäli sivusto on asennettu sellaiselle käyttöjärjestelmälle, jonka sähköpostipalveluissa ei voi käyttää osoitteen muokkausta, ks. myös sähköpostikomponentti.

5.4 Alustustoimet

Alustuskomponentti suorittaa seuraavat alustustoimet tässä järjestyksessä:

Järjestys	Toimi	Kuvaus
1	track_errors = 1	PHP:n ympäristömuuttuja track_errors arvoksi asetetaan 1, jolloin PHP palauttaa mahdollisen viimeisimmän virhetilanteen funktiokutsun jälkeen. Virheiden jäljitystä käytetään erityisesti silloin, kun halutaan selvittää, onnistuiko sähköpostiviestin lähetys PHP:n mail()-funktioilla. Tällöin PHP-muuttujassa \$php_errormsg on virheilmoitus.
2	Tietokantayhteyden luonti	Komponentti luo yhteyden tietokantapalvelimelle dbConnect()-palvelulla. Käyttäjä saa virheilmoituksen, mikäli yhteyden muodostus epäonnistuu.
3	Tietokannan avaus	Komponentti avaa tietokannan dbOpen()-palvelulla. Käyttäjä saa virheilmoituksen, mikäli tietokannan avaus epäonnistuu.

5.5 Palvelut

Alustuskomponentti tarjoaa seuraavat palvelut.

5.5.1 getCodeDefault

Palauttaa kooditauluun tallennetun oletusarvon. Oletusarvot tallennetaan DEFAULT-kokoelmaan Codes-taulussa. Oletusarvoja kutsutaan koodin nimellä.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
name	String	Kyllä	Oletusarvon nimi
default	String	Ei	Oletusarvo, jos koodin arvoa ei kyetä palauttamaan

Palauttaa String-tyyppisen arvon.

Esimerkki: `$default = getCodeDefault("SLOT", "20");`

5.5.2 getCodeValue

Palauttaa kooditauluun tallennetun arvon annetusta kokoelmasta koodin nimellä.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
collection	String	Kyllä	Koodikokoelman nimi
name	String	Kyllä	Koodin nimi
default	String	Ei	Oletusarvo, jos koodin arvoa ei kyetä palauttamaan

Palauttaa String-tyyppisen arvon.

Esimerkki: `$value = getCodeValue("PREF", "NOT_SUITABLE", "4");`

5.5.3 getGUID

Palauttaa yksilöllisen tunnisteen, GUID (Globally Unique Identifier).

Palvelulla ei ole parametreja.

Palauttaa 32-merkkisen String-tyyppisen yksilöllisen tunnisteen. Tunnisteessa ei käytetä merkkejä I, 1, O eikä 0 epäselvyyksien välttämiseksi.

Esimerkki: `$id = getGUID();`

Ks. taulukon käyttö `getParameterString-` ja `getParameterInteger-` palvelujen yhteydessä.

5.5.4 getParameterInteger

Palauttaa nimetyn parametrin arvon taulukkomuuttujasta ja konvertoi arvon positiiviseksi kokonaisluvuksi. Jos arvoa ei ole, palvelu palauttaa arvon samannimisestä istuntomuuttujasta `$_POST[]`. Jos tätäkään arvoa ei ole, palvelu palauttaa oletusarvon, joka välitetään parametrina.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
queryStr	String[]	Kyllä	Taulukko parametreista ja niiden arvoista, jotka on välitetty osoiterivillä.
name	String	Kyllä	Parametrin nimi taulukkomuuttujassa
default	int	Ei	Oletusarvo, jos parametrin arvoa ei kyetä palauttamaan

Palauttaa positiivisen kokonaisluvun.

Esimerkki: `$id = getParameterInteger($queryString, "id");`

5.5.5 getParameterString

Palauttaa nimetyn parametrin arvon taulukkomuuttujasta. Jos arvoa ei ole, palvelu palauttaa arvon samannimisestä istuntomuuttujasta `$_POST[]`. Jos tätäkään arvoa ei ole, palvelu palauttaa oletusarvon, joka välitetään parametrina.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
queryStr	String[]	Kyllä	Taulukko parametreista ja niiden arvoista, jotka on välitetty osoiterivillä.
name	String	Kyllä	Parametrin nimi taulukkomuuttujassa
maxLength	int	Kyllä	Katkaisee parametrin arvon annettuun enimmäispituuteen
default	String	Ei	Oletusarvo, jos parametrin arvoa ei kyetä palauttamaan

Palauttaa merkkijonon, joka on lyhennetty parametrina annettuun maksimipituuteen.

Esimerkki: `$lang = getParameterString($queryString, "lang", 5, "fi");`

5.5.6 `getPreviousPage`

Palauttaa järjestäjän tai osallistujan viimeksi käsittelemän sivun osoitteen tai oletussivun, jos tallennettua osoitetta ei ole.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
userid	int	Kyllä	Viiteavain käyttäjään Users-taulussa
default	String	Kyllä	Oletusarvo, jos historia-arvoa ei kyetä palauttamaan

Palauttaa String-tyyppisen arvon.

Esimerkki: `$linkTo = getPreviousPage($id, "group.php");`

5.5.7 `getQueryString`

Palauttaa osoiterivillä välitetyt parametrit taulukkomuuttujassa. PHP-muuttuja `$_SERVER["QUERY_STRING"]` palauttaa osoiterivin parametrit yhtenä merkkijonona, jonka palvelu jäsentää nimi-arvo-parillisiksi merkkijonoiksi taulukkomuuttujassa.

Palvelulla ei ole parametreja.

Palauttaa arvot String-tyyppisenä taulukkona, missä elementteinä on osoiterivillä välitetyt parametrit.

Esimerkki: `$queryString = getQueryString();`

Ks. taulukon käyttö `getParameterString`- ja `getParameterInteger`-palvelujen yhteydessä.

5.5.8 getUserRole

Palauttaa käyttäjän roolin nimen. Roolit on tallennettu tietokannan Codes-tauluun seuraavasti:

Kokoelma	Nimi	Arvo	Lajittelu-järjestys
ROLE	ORGANIZER	Organizer	
ROLE	PARTICIPANT	Participant	

Palvelulla on seuraavat parametrit:

Parametri	Tietotyyppi	Pakollinen	Kuvaus
user	int	Kyllä	Avain käyttäjän tietueeseen Users-taulussa, jos käyttäjän tietoja ei löydetä annetulla avaimella, palvelu palauttaa rooliksi "PARTICIPANT"

Palauttaa merkkijonon "ORGANIZER", jos sisäänkirjautuneen käyttäjän sähköpostiosoite kuuluu järjestäjille sallittujen toimialueiden joukkoon (ks. myös DOMAIN-kokoelma). Palvelu palauttaa "PARTICIPANT"-merkkijonon, jos sähköpostiosoite ei kuulu joukkoon tai käyttäjää ei löydy järjestelmästä tai käyttäjä ei ole kirjautunut sisään.

Palvelu käyttää tietokantakomponentin palveluja.

Esimerkki: `$isOrganizer = (getUserRole() == "ORGANIZER");`

5.5.9 relocate

Kirjoittaa HTML-otsakkeeseen uudelleensijoituskäskyn, jolla palvelin siirtää käyttäjän parametrina annettuun osoitteeseen.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
url	String	Kyllä	Sivu tai osoite, johon käyttäjä halutaan siirtää

Palvelu ei palauta mitään.

Esimerkki: `relocate("index.php");`

5.5.10 savePreviousPage

Tallentaa järjestäjän tai osallistujan viimeksi käsittelemän sivun osoitteen.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
userid	int	Kyllä	Viiteavain käyttäjään Users-taulussa
URL	String	Kyllä	Osoite parametreineen viimeksi käsitelty sivuun

Tämä palvelu ei palauta mitään.

Esimerkki: `savePreviousPage($_SESSION["user"], "group.php?id=7");`

5.5.11 threeDots

Katkaisee merkkijonon enintään haluttuun pituuteen ja yrittää katkaista monisanaiset merkkijonot katkaisukohtaa lähimmästä välilyönnistä. Lisää palautettavan merkkijonon loppuun kolme pistettä (...).

Parametri	Tietotyyppi	Pakollinen	Kuvaus
value	String	Kyllä	Katkaistava merkkijono, jos merkkijono on pitempi kuin maksimipituus
maxLength	int	Kyllä	Sallittu enimmäispituus

Palauttaa enimmäispituuteen katkaistun merkkijonon, jonka loppuun on lisätty kolme pistettä. Jos merkkijono ei ylitä sallittua enimmäispituutta, palvelu palauttaa sen sellaisenaan.

Esimerkki: `echo threeDots("Some very long sentence to be truncated.", 25);`

Palauttaa: Some very long sentence...

6 Tietokantakomponentti

Erillinen tietokantakomponentti tarjoaa järjestelmälle tietokantapalvelut. Komponenttia vaihtamalla järjestelmässä voidaan käyttää eri tietokantoja. Järjestelmä on tietokantariippumaton kutsumalla tietokantakomponentin palveluja.

6.1 db.php-komponentin käyttö

Skriptinä toteutettu db.php-komponentti liitetään sovelluslogiikkaan sivun alustusosassa include-direktiivillä.

Esimerkki: `include "inc/db.php";`

6.2 Virhekäsittely

Jos tietokantapalvelun suorituksen aikana syntyy virhe, esimerkiksi yhteys tietokantaan katkeaa tai parametrina annettu SQL-lause on virheellinen, välittää palvelu virheestä tiedon dbErrCode- ja dbErrText-palveluilla. Jokaisen palvelukutsun jälkeen on tarkistettava, mikäli kutsun jälkeen on saatavilla virhekoodi esimerkiksi dbErrCode-palvelulla. Esimerkiksi tietueita tallennettaessa ja päivitetessä voi syntyä duplikaatti eli tallennettava tieto on jo ennestään tietokannassa ja uusi tieto

rikkoisi viite-eheyksen. Tällöin tallennuskutsun jälkeen on tarkistettava dbErrCode-palvelun palauttama virhekoodi. Jos se on DB_ERR_DUPLICATE, on tallennettavassa tietueessa duplikaatti ja suoritus keskeytyy.

6.3 Tietokannan luonti

Tietokanta luodaan seuraavilla SQL-lauseilla. Luontijärjestys on oleellinen, sillä tauluissa on viiteavaimia muihin tauluihin. Taulujen lisäksi on luotava sekvenssi tai sen toimintaa imitoiva tietokantaobjekti.

```
CREATE TABLE Codes (
  Collection VARCHAR(16) NOT NULL,
  Name VARCHAR(16) NOT NULL,
  Value TEXT NOT NULL,
  Sorting INT,
  PRIMARY KEY (Collection,Name)
);
```

```
CREATE TABLE Values (
  Collection VARCHAR(16) NOT NULL,
  Name VARCHAR(16) NOT NULL,
  Lang VARCHAR(5) NOT NULL,
  Value TEXT NOT NULL,
  Sorting INT,
  PRIMARY KEY (Collection,Name,Lang)
);
```

```
CREATE TABLE Locales (
  Name VARCHAR(4) NOT NULL,
  Lang VARCHAR(5) NOT NULL,
  Value TEXT NOT NULL,
  PRIMARY KEY (Name,Lang)
);
```

```
CREATE TABLE Users (
  ID INTEGER NOT NULL,
  Lastname VARCHAR(64),
  Firstname VARCHAR(64),
  Email VARCHAR(255) NOT NULL,
  Password VARCHAR(32) NOT NULL,
  GUID VARCHAR(32),
  Lang VARCHAR(5),
  Visited DATETIME,
  Created DATETIME NOT NULL,
  PRIMARY KEY (ID),
```



```
UNIQUE INDEX (GUID)
);
```

```
CREATE TABLE Groups (
  ID INTEGER NOT NULL,
  UserID INTEGER NOT NULL,
  Name VARCHAR(255) NOT NULL,
  Updated DATETIME NOT NULL,
  PRIMARY KEY (ID),
  FOREIGN KEY (UserID) REFERENCES Users (ID) ON DELETE CASCADE,
  UNIQUE INDEX (UserID,Name)
);
```

```
CREATE TABLE Messages (
  GroupID INTEGER NOT NULL,
  Type VARCHAR(16) NOT NULL,
  Subject VARCHAR(255),
  Value TEXT,
  PRIMARY KEY (GroupID,Type),
  FOREIGN KEY (GroupID) REFERENCES Groups (ID) ON DELETE CASCADE
);
```

```
CREATE TABLE Calendars (
  ID INTEGER NOT NULL,
  GroupID INTEGER NOT NULL,
  Name VARCHAR(255) NOT NULL,
  Type VARCHAR(16) NOT NULL,
  Start TIME NOT NULL,
  Finish TIME NOT NULL,
  Slot INTEGER NOT NULL,
  Locked DATETIME,
  PRIMARY KEY (ID),
  FOREIGN KEY (GroupID) REFERENCES Groups (ID) ON DELETE CASCADE
);
```

```
CREATE TABLE Members (
  GroupID INTEGER NOT NULL,
  UserID INTEGER NOT NULL,
  PRIMARY KEY (GroupID,UserID),
  FOREIGN KEY (GroupID) REFERENCES Groups (ID) ON DELETE CASCADE,
  FOREIGN KEY (UserID) REFERENCES Users (ID) ON DELETE CASCADE
);
```

```
CREATE TABLE Offers (
```

```
ID INTEGER NOT NULL,  
CalendarID INTEGER NOT NULL,  
Weekday INTEGER NOT NULL,  
Period TIME NOT NULL,  
PRIMARY KEY (ID),  
FOREIGN KEY (CalendarID) REFERENCES Calendars (ID) ON DELETE CASCA-  
DE  
);
```

```
CREATE TABLE Matches (  
UserID INTEGER NOT NULL,  
OfferID INTEGER NOT NULL,  
Pref INTEGER NOT NULL,  
Booked DATETIME,  
PRIMARY KEY (UserID,OfferID),  
FOREIGN KEY (UserID) REFERENCES Users (ID) ON DELETE CASCADE,  
FOREIGN KEY (OfferID) REFERENCES Offers (ID) ON DELETE CASCADE  
);
```

```
CREATE TABLE Weeks (  
CalendarID INTEGER NOT NULL,  
Year INTEGER NOT NULL,  
Week INTEGER NOT NULL,  
PRIMARY KEY (CalendarID,Year,Week),  
FOREIGN KEY (CalendarID) REFERENCES Calendars (ID) ON DELETE CASCA-  
DE  
);
```

```
CREATE TABLE Calendar_History (  
UserID INTEGER NOT NULL,  
CalendarID INTEGER NOT NULL,  
PRIMARY KEY (UserID,CalendarID),  
FOREIGN KEY (UserID) REFERENCES Users (ID) ON DELETE CASCADE,  
FOREIGN KEY (CalendarID) REFERENCES Calendars (ID) ON DELETE CASCA-  
DE  
);
```

```
CREATE TABLE Group_History (  
UserID INTEGER NOT NULL,  
GroupID INTEGER NOT NULL,  
PRIMARY KEY (UserID,GroupID),  
FOREIGN KEY (UserID) REFERENCES Users (ID) ON DELETE CASCADE,  
FOREIGN KEY (GroupID) REFERENCES Groups (ID) ON DELETE CASCADE  
);
```

```
CREATE TABLE Page_History (  
  UserID INTEGER NOT NULL,  
  URL TEXT NOT NULL,  
  PRIMARY KEY (UserID),  
  FOREIGN KEY (UserID) REFERENCES Ohtu_Users (ID) ON DELETE CASCADE,  
);
```

6.4 Vakiot

Järjestelmä osoittaa tietokannan objekteja, kuten tauluja, vakioarvoilla. Vakiot korvaavat suorat viittaukset tietokannan objekteihin, jolloin esimerkiksi taulut voidaan nimetä järjestelmän vaatimuksista riippumatta.

Komponentissa on seuraavat vakiot:

Vakion nimi	Tietotyyppi	Tietokannan objekti
DB_ERR_DUPLICATE	int	Virhekoodin numero, joka ilmoittaa, mikäli tietokantaan yritettiin tallentaa duplikaatti
DB_NAME	int	Tietokannan nimi
DB_PASSWORD	String	Salasana, jolla kirjaudutaan tietokantapalvelimelle
DB_PROVIDER	String	Tietokantapalvelun nimi, jota voidaan näyttää esimerkiksi virheilmoituksissa
DB_SERVER	String	URI tietokantapalvelimelle, paikalliseen palvelimeen viitataan esimerkiksi localhost-osoitteella
DB_SYSDATE	String	Aikaleimafunktion nimi, funktio saattaa olla erilainen eri tietokannoissa
DB_USERNAME	String	Käyttäjätunnus, jolla kirjaudutaan tietokantapalvelimelle
SEQUENCE_ID	String	Sekvenssin nimi
TABLE_CALENDAR_HISTORY	String	Aikataulujen historiataulun nimi
TABLE_CALENDARS	String	Aikataulu-taulun nimi
TABLE_CODES	String	Kooditaulun nimi
TABLE_GROUP_HISTORY	String	Ryhmiä historiataulun nimi
TABLE_GROUPS	String	Ryhmätaulun nimi
TABLE_LOCALES	String	Tekstitaulun nimi
TABLE_MATCHES	String	Sopivuus-taulun nimi
TABLE_MEMBERS	String	Osallistujataulun nimi
TABLE_MESSAGES	String	Viestitaulun nimi
TABLE_OFFERS	String	Tarjotut ajat -taulun nimi
TABLE_PAGE_HISTORY	String	Sivuhistoriataulu, johon tallennetaan järjestäjän tai osallistujan viimeksi käsittelemä sivu
TABLE_USERS	String	Käyttäjätaulun nimi
TABLE_VALUES	String	Kielisidonnaisen kooditaulun nimi
TABLE_WEEKS	String	Viikkotaulun nimi

Esimerkki: `$rsSet = dbQuery("SELECT * FROM ". TABLE_CODES . "WHERE Collection=LANGUAGE ");`

6.5 Palvelut

Tietokantakomponentti tarjoaa seuraavat palvelut.

6.5.1 dbBegin

Aloittaa transaktion. Tällä palvelulla ei ole parametreja eikä se palauta mitään.

6.5.2 dbCommit

Päättää transaktion ja kommitoi tietokantaan tehdyt muutokset. Tällä palvelulla ei ole parametreja eikä se palauta mitään.

6.5.3 dbRollback

Peruuttaa transaktion. Tällä palvelulla ei ole parametreja eikä se palauta mitään.

6.5.4 dbClose

Sulkee avatun tietokantaresurssin, kuten tietokannan.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
resource	Object	Kyllä	Suljettava resurssiobjekti

Tämä palvelu ei palauta mitään.

Esimerkkikutsu: `dbClose($dbConn);`

6.5.5 dbConcatenate

Katenoi parametrina annetut merkkijonot yhdeksi merkkijonoksi, joka puolestaan muodostaa samat merkkijonot katenoivan tietokantafunktiokutsun. Katenoitavat merkkijonot välitetään funktiolle parametritaulukossa.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
\$strArray	String[]	Kyllä	Merkkijonot katenoiva tietokantaoperaatio

Palauttaa merkkijonon, joka katenoi parametritaulukossa olevat merkkijonot käytämällä tietokannan katenointifunktiota tai -metodia.

Esimerkkikutsu:

```
$arr[] = "%";
```

```
$arr[] = "C.Value";
```

```
$arr[] = "%";
```

```
$sql = "SELECT * FROM Codes WHERE Name LIKE ". dbConcatenate($arr);
```

6.5.6 dbConnect

Muodostaa yhteyden tietokantapalvelimelle.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
server	String	Kyllä	Viittaus tietokantapalvelimeen
port	String	Kyllä	Yhteyden porttinumero
username	String	Kyllä	Käyttäjätunnus
password	String	Kyllä	Salasana

Palauttaa objektin, joka viittaa avoimena olevaan tietokantayhteyteen tai null, jos tietokantayhteyttä ei kyetty muodostamaan. Jos porttinumeroa ei käytetä, välitetään palvelulle tyhjä merkkijono.

Esimerkkikutsu: `$dbConn = dbConnect("localhost", , "admin", "XYZ123");`

6.5.7 dbErrCode

Palauttaa viimeisimmän tietokantakäsittelyssä syntyneen virhekoodin.

Tällä palvelulla ei ole parametreja.

Palauttaa virhekoodin int-tyyppisenä kokonaislukuna.

Esimerkkikutsu: `echo dbErrCode();`

6.5.8 dbErrText

Palauttaa viimeisimmän tietokantakäsittelyssä syntyneen virheilmoitustekstin.

Tällä palvelulla ei ole parametreja.

Palauttaa String-tyyppisen virheilmoitustekstin.

Esimerkkikutsu: `echo dbErrText();`

6.5.9 dbExecute

Suorittaa tietokantaoperaation.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
sql	String	Kyllä	Suoritettava SQL-lause

Tämä palvelu ei palauta mitään.

Esimerkkikutsu: `dbExecute("INSERT INTO SomeTable (ID,Value) VALUES (123, somevalue)");`

6.5.10 dbFree

Vapauttaa varatun tietokantaresurssin, kuten avatun tulosjoukon.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
resource	Object	Kyllä	Vapautettava resurssiobjekti

Tämä palvelu ei palauta mitään.

Esimerkkikutsu: `dbFree($rsSet);`

6.5.11 dbGetID

Kutsuu tietokannan sekvenssiobjektia, joka palauttaa järjestyksessä seuraavan vapaan yksilöllisen avainarvon.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
sequence	String	Kyllä	Sekvenssin nimi

Palauttaa avainarvon positiivisena kokonaislukuna, joka on suurempi kuin nolla tai arvon, joka on pienempi kuin 1, jos sekvenssin kutsu epäonnistui.

Esimerkkikutsu: `$id = dbGetID("someseries");`

6.5.12 dbGetRow

Palauttaa järjestyksessä seuraavan monikon tulosjoukosta.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
resultSet	ResultSet	Kyllä	Tulosjoukko

Palauttaa monikon String-tyyppisenä taulukkona, missä elementteinä on tietueen kenttien arvot.

Esimerkkikutsu: `$rsItem = dbGetRow($rsSet);`

Viittaus tietueen kentän "Name"arvoon: `echo $rsItem["Name"];`

6.5.13 dbNumRows

Palauttaa monikoiden määrän tulosjoukossa.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
resultSet	ResultSet	Kyllä	Tulosjoukko

Palauttaa monikoiden määrän tulosjoukossa int-tyyppisenä kokonaislukuna.

Esimerkkikutsu: `$cnt = dbNumRows($rsSet);`

6.5.14 dbOpen

Avaa tietokantapalvelimella olevan tietokannan.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
name	String	Kyllä	Avattavan tietokannan nimi

Palauttaa tietokantaobjektin, joka viittaa avoinna olevaan tietokantaan tai null, jos tietokantaa ei kyetty avaamaan.

6.5.15 dbQuery

Suorittaa kyselyn.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
sql	String	Kyllä	SQL-kysely

Palauttaa ResultSet-tyyppisen tulosjoukon tai null, jos kysely epäonnistui.

Esimerkkikutsu: `$rsSet = dbQuery("SELECT * FROM SomeTable");`

7 Terminointikomponentti

Osajärjestelmien suoritus päättyy terminointikomponenttiin. Komponentti suorittaa tarvittavat terminointitoimenpiteet, kuten esimerkiksi tietokannan sulkemisen.

7.1 terminate.php-komponentin käyttö

Skriptinä toteutettu terminate.php-komponentti liitetään sovelluslogiikkaan sivun loppuun include-direktiivillä.

Esimerkki: `include "inc/terminate.php";`

Koska komponentti katkaisee yhteyden tietokantaan, on se sijoitettava osajärjestelmän viimeiseksi suoritettavaksi osaksi.

8 Monikielisyyden tuen komponentti

Järjestelmässä on monikielisyyden tuki. Monikieliset tekstit on tallennettu tietokantaan, josta järjestelmä noutaa halutun tekstin yksilöllisellä nimellä ja kielikoodilla. Järjestelmän oletuskieli on esitelty alustuskomponentissa globaalina vakiona `DEFAULT_LANGUAGE`. Järjestelmän tukemat kielet ovat suomi ja englanti amerikkalaisella dialektilla. Näiden kielikoodit ovat `fi` (suomi) ja `en-US` (englanti).

8.1 i18n.php-komponentin käyttö

Skriptinä toteutettu i18n.php-komponentti liitetään sovelluslogiikkaan sivun alustusosassa include-direktiivillä.

Esimerkki: `include "inc/i18n.php";`

8.2 Kielikoodi

Kielikoodi voi olla kaksiosainen, jolloin eri osat erotetaan väliviivalla. Ensiosassa on varsinainen kielikoodi ja valinnaisessa toisio-osassa on maakoodi. Järjestelmä käyttää ISO 639 standardin mukaisia kaksikirjaimisia kielikoodeja ja ISO 3166 standardin mukaisia kaksikirjaimisia maakoodeja.

Kielikoodin syntaksi: `kielikoodi[-maakoodi]`

Esimerkkejä: `fi`, `en-US`

8.3 Tekstin osoitus

Järjestelmä osoittaa halutun tekstin yksilöllisellä nimellä ja kielikoodilla. Tekstin erikielisillä versioilla on sama yksilöllinen nimi, mutta eri kielikoodi. Näiden arvojen yhdistelmä muodostaa yksilöllisen avaimen.

Esimerkkejä:

Nimi	Kielikoodi	Teksti
EADR	fi	Sähköpostiosoite
EADR	en-US	Email Address

8.4 Parametrointi

Käyttöliittymässä esitettävä teksti saattaa sisältää vaihtuvia tietoja tekstin sisällä. Jotta tekstiä ei tarvitse koota useista osista, voidaan tekstirunkoon lisätä merkki parametrilla. Parametrien arvot välitetään funktion kutsussa ja tekstirungossa olevat parametrit korvataan välitetyillä arvoilla. Parametrit merkitään tekstirunkoon muodossa `%n`, missä `n` on positiivinen kokonaisluku väliltä 0 - 9. Tekstirungossa voi olla useita parametreja samanaikaisesti ja sama parametri voi esiintyä moneen kertaan.

Esimerkki: `Tänään on %0.`

Tulos parametrin arvolla `"maanantai"`: `Tänään on maanantai.`

Esimerkkikutsu:

```
$arr[0] = "maanantai"; echo localText("fi", "TODY", $arr);
```

8.5 Tekstien ylläpito

Tekstejä ylläpidetään erillisessä XML-tiedostossa, jolloin ne voidaan tuoda suoraan valittuun tietokantaan. Tiedoston muoto on:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<texts>
<text>
<name>nimi</name>
<value lang="kielikoodi" >teksti</value>
</text>
</texts>
```

Esimerkki:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<texts>
<text>
<name>PSWD</name>
<value lang="fi" >Salasana</value>
<value lang="en-US" >Password</value>
</text>
<text>
<name>EADR</name>
<value lang="fi" >Sähköpostiosoite</value>
<value lang="en-US" >Email Address</value>
</text>
</texts>
```

8.6 Palvelut

Yhteys tietokantaan on avattava ennen monikielisyyden tuen palvelujen kutsua. Monikielisyyden tuki tarjoaa seuraavat palvelut.

8.6.1 defaultLanguage

Tarkistaa ja palauttaa oletuskielikoodin.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
lang	String	Kyllä	Kielikoodi, joka halutaan tarkistaa
default	String	Kyllä	Oletuskielikoodi, jos parametrina annettua kielikoodia ei kyetä tunnistamaan

Palauttaa String-tyyppisen tarkistetun kielikoodin tai oletusarvon, jos kielikoodia ei ole välitetty parametrina tai sitä ei kyetä tunnistamaan.

Esimerkkikutsu: `$lang = defaultLanguage("en-US", "fi");`

8.6.2 languageDrop

Laatii alaspudotusvalikon tuetuista kielistä. Luettelossa näytetään "TONGUE-kokoelman koodit Codes-taulusta. "TONGUE-kokoelmassa on lueteltu tuetut kielikoodit sekä niiden selväkieliset kuvaukset perusmuodossa. "LANGUAGE-kokoelmassa kielikoodit on lueteltu siten, että niiden selväkieliset kuvaukset ovat translatiivimuodossa.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
lang	String	Kyllä	Kielikoodi, joka on parhaillaan valittuna ja joka valitaan oletuksena listalta

Palauttaa valmiin HTML-lomakekentän String-tyyppisenä merkkijonona. Kieli voidaan valita alaspudotusvalikosta. Parhaillaan valittuna oleva kieli on valittu valmiiksi.

Esimerkkikutsu: `echo languageDrop("fi");`

8.6.3 languageList

Laatii linkkiluettelon tuetuista kielistä. Luettelossa näytetään "LANGUAGE-kokoelman koodit Codes-taulusta.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
lang	String	Kyllä	Kielikoodi, joka on parhaillaan valittuna ja jonka yhteydessä ei näytetä linkkiä, koska käyttäjä ei voi valita kyseistä kielikoodia
parameters	String	Ei	

Palauttaa valmiin HTML-taulukon String-tyyppisenä merkkijonona. Kieli voidaan valita luettelosta linkillä. Parhaillaan valittuna olevassa kielessä ei ole linkkiä.

Esimerkkikutsu: `echo languageList("fi");`

8.6.4 localText

Noutaa tietokannasta tekstirungon ja palauttaa sen tarvittaessa muokattuna kutsuvälle proseduurille. Parametrien arvotaulukko on valinnainen parametri ja se voidaan jättää välittämättä.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
lang	String	Kyllä	Kielikoodi, jossa on mukana mahdollinen maakoodi
name	String	Kyllä	Tekstirungon yksilöllinen nimi
replacements	String[]	Ei	Taulukko parametrien arvoista

Palauttaa String-tyyppisen lokalisoitua tekstiä, johon on liitetty mahdolliset parametrien arvot. Jos palvelu ei kykene palauttamaan pyydettyä tekstiä, palauttaa se "missing-merkkijonon, joka näytetään "missingText-tyylielementillä. Mikäli tietokannan avaamisessa syntyy virhetilanne, käytetään yleistä "FAIL"-nimeä, joka palauttaa fi-kielikoodilla tekstiä "Virheilmoitus" ja en-US -kielikoodilla tekstiä "Error Message".

Esimerkkikutsu: `echo localText("fi", "BSIN", $values);`

9 Kalenterikomponentti

Kalenterikomponentti tulostaa kuukausikalenterin. Kalenterista voidaan merkitä viikkoja jotka ilmoittavat aikataulun keston. Valitut viikot näkyvät kalenterissa korostetusti.

9.1 calendar.php -komponentin käyttö

Calendar.php -komponentti toteutetaan skriptinä ja se liitetään sovelluslogiikkaan include -direktiivillä.

Esimerkki: `include "inc/calendar.php";`

9.2 Palvelut

Kalenterikomponentti tarjoaa seuraavat palvelut.

`showCalendar`

Tuottaa pyydetyn kalenterin valmiina HTML-tilana. Kalenterin sisältö riippuu annettavista parametreista. Viikonpäivät ja kuukausien nimet haetaan tietokannasta lang -parametrin perusteella. Ennen kalenterin tulostamista tietokannan Week -taulusta haetaan ne viikot, joille on merkittynä id -parametrin osoittama aikataulu. Kalenteri tulostetaan viikko kerrallaan. Tulostettavat viikot alkavat start -parametrin osoittamasta aikaleimasta, kuitenkin aikaisin ajankohta on kuluva ajankohdan aikaleima. Editable -parametri määrää sijoitetaanko jokaisen viikon perään painike, jota kutsumalla voidaan muokata kyseisen viikon merkintää. Mikäli viikolla ei ole merkintää painikkeen kutsuminen tuottaa sille merkinnän tietokannan Weeks -tauluun. Jos viikolla on merkintä se poistetaan Weeks -taulusta.

Kalenterissa korostetaan viikot eritavoilla sen mukaan onko niille luotuna aikataulu.

lua. Aikataulullisten viikkojen viikonpäivät korostetaan CSS-tyylin `bgchosenday` -luokalla. Aikatauluttomien viikkojen päivät korostetaan `bgday` -luokalla. Myös kuuluva päivä korostetaan CSS-tyylin `bgcurrday` -luokalla. Mikäli kalenteri ei ole muokattavissa se on aikataulun keston mittainen. Kalentereissa on mahdollista selailta viikkoja eteen ja taaksepäin, siten että aikaisin viikko on kuluva viikko ja viimeinen viikko on kuluva päivästä kaksi vuotta eteenpäin. Selailupainikkeet sijoitetaan kalenterin oikeaan ylä- ja alakulmaan. Ylempi painike lataa sivun uudelleen antaen palvelulle parametriksi edellistä kertaa viikkoa aikaisemman aikaleiman. Alempi painike lataa sivun uudelleen antaen palvelulle parametriksi edellistä kertaa viikkoa myöhemmän aikaleiman.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
<code>id</code>	Int	Kyllä	Aikataulun yksilöllinen avain tietokannassa, jolla aikatauluun kuuluvat viikot noudetaan esitettäväksi tietokannassa
<code>lang</code>	String	Ei	Käyttöliitymän kielikoodi, käyttää järjestelmän oletusarvona <code>DEFAULT_LANGUAGE</code> .
<code>start</code>	Int	Ei	Unix tyylinen aikaleima, joka esittää mistä viikosta kalenterin esittäminen aloitetaan. Oletusarvona kuluva päivän aikaleima.
<code>editable</code>	Boolean	Ei	Ilmoittaa, voidaanko kalenterista valita viikkoja. Jos arvo on <code>true</code> , voi järjestäjä valita kalenterista aikataulun keston viikkoina. Arvolla <code>false</code> , kalenterissa ei ole viikkojen valintamahdollisuutta.

Jos `id`-parametrin arvo on pienempi kuin 1 aikataulua ei löydy annetulla avaimella, tai käyttäjällä ei ole oikeutta kyseisen aikataulun kalenterin selailuun, ei kalenterissa näytetä aikataulun kestoja viikkoina. Jos arvo on suurempi kuin 0 ja järjestäjällä on oikeus aikataulun katsomiseen, näytetään vastaavan aikataulun kesto viikkoina. Virheellisen `lang` -parametriin reagoiminen on kuvattu monikielisyyden tuessa. `start` -parametri ei voi olla aikaisempi kuin kuluva päivän aikaleima, ja virheellisen arvon sattuessa kalenterin alkaa tästä aikaleimasta, joka on myös parametrin oletusarvo. `Editable` -parametrin oletusarvona on `false`.

Tämä palvelu palauttaa kalenterin HTML-tilana.

Esimerkkikutsu: `showCalendar("testi","fi",$timestamp,true);`

Tarvittavat SQL-lauseet.

1. `SELECT week, year FROM weeks WHERE calendarid = käyttäjätunnus ORDER BY year, week`

10 Reunalistakomponentti

Reunalistakomponentin avulla näytetään käyttäjälle linkkeinä erilliset sivut, joilta löytyvät järjestelmän käytettävät toiminnot. Komponentti tulostetaan käyttöliittymään sivun vasempaan laitaan. Reunalista on puurakenteinen ja käyttäjä voi halutessaan sulkea ja uudelleen avata rakenteen osia. Oletusarvoisesti puurakenne on auki. Reunalista helpottaa käyttäjän navigointia järjestelmässä sekä auttaa hahmottamaan järjestelmän kokonaisuutta. Reunalista kertoo myös käyttäjän tämän hetkisen sijaintisivun järjestelmässä.

10.1 sidepane.php-komponentin käyttö

Skriptinä toteutettu sidepane.php-komponentti liitetään sovelluslogiikkaan sivun alustusosassa include-direktiivillä.

Esimerkki: `include "inc/ sidepane.php";`

10.2 Palvelut

Reunalistakomponentti tarjoaa seuraavat palvelut

10.2.1 activeLinkTo(\$userID)

Palvelu palauttaa arvonaan String-jonon, jossa on sivun nimi ja tarvittavat parametrit, esimerkiksi `member.php?p=33&id=28`

missä `p` = `parentID` eli viiteavain `Groups`-tauluun (`Members.GroupID`) ja `id` = osallistujan avain (`Members`-taulussa eli `Members.ID`).

Jos palvelu ei löydä aktiivista tietuetta, niin se palauttaa tyhjän merkkijonon.

10.2.2 sidePane(\$id, \$lang, \$itemID, \$parentID)

Palvelu palauttaa arvonaan HTML-taulukon, jossa on valmis reunalista.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
id	int	Kyllä	Käyttäjän ID, jonka avulla saadaan tieto käyttäjän järjestämisistä ja osallistumisista
lang	string	Kyllä	Kielikoodi, joka on valittuna
itemID	string	Kyllä	Kertoo ryhmän toiminnon nimen, esim. Uusi aikataulu
parentID	string	Kyllä	Kertoo ryhmän ID:n, johon liittyviä linkkejä tulostetaan näytölle

11 Järjestelmän esittely komponentti

Järjestelmän esittely esittää järjestelmän keskeisimmät toiminnot näyttökuvasarjana. Esittelyn on tarkoitus antaa uudelle käyttäjälle kuva siitä, mistä järjestelmässä on kysymys ja mitä sillä voi tehdä. Linkki järjestelmän esittelyyn on järjestelmän etusivulla sisäänkirjautusmissivulla.

Järjestelmän esittelyssä kuvataan seuraavat käyttötapaukset:

1. Rekisteröityminen
2. Sisään kirjautuminen
3. Unohtuneen salasanan lähetys sähköpostiin
4. Uuden ryhmän luominen
5. Uuden aikataulun luominen ryhmälle
6. Aikojen tarjoaminen uuteen aikatauluun
7. Osallistujien lisääminen ryhmään
8. Viestin lähettäminen osallistujille
9. Osallistujan sopivuuksien syöttäminen
10. Järjestäjän järjestelynäyttö

11. Uloskirjautuminen

Näyttökuvasarja toimii siten, että jokaisesta kuvasta on aina linkki seuraavaan. Jotta käyttäjän ei tarvitse selata kaikkia välissä olevia kuvia läpi päästäkseen haluamaansa näyttökuvaa, näyttökuvasarjan ylälaidassa on valintalaatikko (`<select>`), josta voi siirtyä suoraan haluamansa toiminnon näyttökuvaa.

12 Sähköpostikomponentti

Erillinen sähköpostikomponentti tarjoaa järjestelmälle sähköpostipalvelut. Järjestelmä tukee eri sähköpostijärjestelmiä komponenttia vaihtamalla. Palvelut noudattavat RFC 2821 -määrittelyä.

12.1 email.php-komponentin käyttö

Skriptinä toteutettu email.php-komponentti liitetään sovelluslogiikkaan sivun alustusosassa include-direktiivillä.

Esimerkki: `include "inc/ email.php";`

12.2 Palvelut

Komponentti tarjoaa seuraavat palvelut.

12.2.1 getFile

Lukee sähköpostiviestin rungon tekstitiedostosta. Palvelu lukee tekstitiedostosta ensimmäiset 1 000 000 tavua. Jos tiedosto on suurempi, voidaan määrää kasvattaa muuttamalla raja-arvoa komponentin ohjelmakoodissa.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
file	String	Kyllä	Tiedoston nimi polkuineen

Palauttaa tekstitiedoston sisällön String-tyyppisenä arvona.

Esimerkkikutsu: `$body = getFile("res/message_body.txt");`

12.2.2 invalidEmail

Tarkistaa sähköpostiosoitteen.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
address	String	Kyllä	Tarkistettava sähköpostiosoite

Palauttaa boolean-tyyppisen arvon true, jos sähköpostiosoite ei noudata RFC 2821-määrittelyä tai boolean-tyyppisen arvon false, jos osoite on määrittelyn mukainen.

Esimerkkikutsu: `$isOk = invalidEmail("tero.testaaja@hotmail.com");`

12.2.3 sendMail

Lähetää sähköpostiviestin. Jos jokin valinnaisista parametreista jätetään käyttämättä, välitetään arvoksi tyhjä merkkijono.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
fromName	String	Kyllä	Valinnainen lähettäjän nimi
fromEmail	String	Kyllä	Lähettäjän sähköpostiosoite
toName	String	Kyllä	Valinnainen vastaanottajan nimi
toEmail	String	Kyllä	Vastaanottajan sähköpostiosoite
bccName	String	Kyllä	Valinnainen piilokopion vastaanottajan nimi
bccEmail	String	Kyllä	Valinnainen piilokopion vastaanottajan sähköpostiosoite
replyName	String	Kyllä	Valinnainen vastauksen vastaanottajan nimi
replyEmail	String	Kyllä	Valinnainen vastauksen vastaanottajan sähköpostiosoite
subject	String	Kyllä	Viestin otsikko
body	String	Kyllä	Viestin teksti
html	boolean	Kyllä	Boolean-tyyppinen arvo true, jos viesti on HTML-muodossa tai false, jos viesti on tekstimuodossa

Palauttaa String-tyyppisen virheilmoituksen, jos lähetys epäonnistuu tai tyhjän merk-

kijonon lähetyksen onnistuessa.

```
Esimerkkikutsu: $fail = sendMail("Tero Testaaja", "tero.testaaja@hotmail.com",
"Ville Vastaanottaja", "ville.vastaanottaja@yahoo.com", , , , "Testilähetys", $bo-
dy, false);
```

12.2.4 wrapAddress

Ympäroï sähköpostiosoitteen kulmasulkujen sisään nimen yhteydessä, esimerkiksi muotoon Tero Testaaja <tero.testaaja@hotmail.com>. Mikäli nimessä on @-merkki, muutetaan tämä "at -merkkijonoksi, koska sähköpostien lähetyksfunktio sallii vain yhden @-merkin esiintymän koko osoitteessa.

Eräissä käyttöjärjestelmissä, kuten Windowsissa, osoitteen muokkausta ei voi käyttää. Jos järjestelmää käytetään tällaisissa käyttöjärjestelmissä, asetetaan yleisen OS_WELL_BEHAVING-vakion arvo boolean-tyyppiseksi false-arvoksi. Muissa käyttöjärjestelmissä, kuten Linuxissa, vakion arvoksi asetetaan boolean-tyyppinen true-arvo. Yleiset vakiot on lueteltu initialize.php-komponentissa.

Parametri	Tietotyyppi	Pakollinen	Kuvaus
name	String	Kyllä	Sähköpostiosoitteen yhteydessä näytettävä käyttäjäväläinen nimi
address	String	Kyllä	Varsinainen sähköpostiosoite

Palauttaa muokatun osoitteen String-tyyppisenä arvona.

```
Esimerkkikutsu: $adr = wrapAddress("Tero Testaaja", "tero.testaaja@hotmail.com");
```

13 Viestityskomponentti

Viestityskomponentti välittää käyttäjälle järjestelmän ilmoituksia, kuten virheilmoitukset ja kehotteet. Viestilajeja on neljä, virheilmoitukset, varoitukset, informatiiviset tiedotteet sekä kehotteet. Virheilmoitukset kertovat järjestelmässä syntyneistä hallituista virhetilanteista. Varoitukset huomauttavat käyttäjää tilanteesta, joka ei ole järjestelmän toiminnan kannalta kriittinen. Järjestelmä voi esimerkiksi varoittaa käyttäjää puuttuvasta nimestä, vaikka se ei olisikaan pakollinen tieto. Informatiiviset tiedotteet kertovat käyttäjälle tietoja järjestelmän toiminnasta, esimerkiksi toiminnon suorituksen päättymisestä. Kehotteet pyytävät käyttäjää vahvistamaan jonkin toimenpiteen, kuten esimerkiksi tietojen poiston.

13.1 message.php-komponentin käyttö

Skriptinä toteutettu message.php-komponentti liitetään käyttöliittymälogiikkaan include-direktiivillä. Komponentti sijoitetaan siihen kohtaan käyttöliittymää, jossa viestit halutaan näyttää.

Esimerkki: `include "inc/message.php";`

13.2 Palvelut

Viestityskomponentti tarjoaa palvelunaan valmiin HTML-taulukon, jossa on haluttu viesti. Käyttöliittymässä näytettävä viesti on laadittava ennen palvelun kutsua.

13.2.1 Viestit

Viestit kootaan taulukkomuuttujaan, joka välitetään komponentille. Komponentti puolestaan näyttää taulukkomuuttujan elementeissä olevat viestit riippuen siitä, missä muuttujassa viestit ovat. Oheisessa taulukossa on kuvattu muuttujien käyttö.

Taulukko- muuttuja	Tietotyyppi	Kuvaus
fail	String[]	Tähän taulukkomuuttujaan kootaan virheilmoitukset. Ensimmäinen virheilmoitus tulee taulukon indeksiin 0.
info	String[]	Tähän taulukkomuuttujaan kootaan tiedotteet. Ensimmäinen tiedote tulee taulukon indeksiin 0.
warn	String[]	Tähän taulukkomuuttujaan kootaan varoitukset. Ensimmäinen varoitus tulee taulukon indeksiin 0.

Esimerkki: `$fail[] = "Suorituksen aikana on syntynyt virhetilanne.";`

Mikäli taulukkomuuttujassa ei ole elementtejä, jää vastaava viestiosio näyttämättä käyttöliittymässä.

13.2.2 Kehotteet

Viestityskomponentti näyttää käyttäjälle myös kehoitteita. Esimerkiksi poisto vahvistus on tällainen kehote. Kehote näytetään, mikäli käyttäjä on painanut jotain painiketta, joka saa aikaan kehoitteen esityksen. Oheisessa taulukossa on kuvattu painikkeet ja niiden kehoitteet.

Painike	Kehote	Kuvaus
submit- delete	Poiston vahvistus	Käyttäjä on painanut submit-delete-nimistä painiketta, jolloin käyttöliittymässä esitetään poiston vahvistuskehote.

13.2.3 Poiston vahvistus

Kehottaa käyttäjää vahvistamaan poiston. Käyttäjä painaa jotain kehoitteessa olevaa painiketta, jotka on nimetty alla olevan taulukon Parametri-sarakkeen mukaisesti.

Parametri	Tyyppi	Kuvaus
submit-delete	sisään	Käyttäjä on painanut HTML-lomakkeella olevaa submit-delete-painiketta, jolloin viestityskomponentti näyttää vastaavan kehoitteen.
submit-remove	ulos	Käyttäjä vastaa kehoitteeseen painamalla submit-remove-painiketta, jolloin sovelluslogiikka voi esimerkiksi poistaa tiedot lopullisesti.
submit-cancel	ulos	Käyttäjä vastaa kehoitteeseen painamalla submit-cancel-painiketta, mikä indikoi käyttäjän peruneen toiminnon.

14 CSV jäsentäjä csv.php

CSV jäsentäjä jäsentää tekstisyötteestä osallistujien tiedot ja lisää osallistujat tietokantaan.

14.1 Palvelut

Komponentti jäsentää ja tallentaa tietokantaan osallistujien tietoja tekstisyötteestä parseCSV-palvelulla.

Esimerkki: `$tmpID = parseCSV($parent, $tmpCSV);`

Parametri	Tietotyyppi	Pakollinen	Kuvaus
parent	String	Kyllä	Viiteavain ryhmään, jonka osallistujaksi jäsennetty tieto tallennetaan. Jos avaimella ei kyetä identifioimaan ryhmää, palvelu palauttaa virheilmoituksen ja keskeyttää suorituksensa.
tmpCSV	String	Kyllä	Jäsennettävä tekstisyöte

Palaute	Kuvaus
> 1	Jäsennys on onnistunut. Arvo viittaa tietokantaan tallennettuun tietueeseen, joka oli aakosjärjestyksessä viimeinen tekstisyötössä.
< 0	Jäsennys on epäonnistunut, antaa palutteena sen rivin numeron negatiivisena, jonka jäsenys ei onnistunut.
0	Syöte on tyhjä tai parent-parametri ei viittaa tietokannassa olevaan ryhmään.

14.2 Toiminnallisuus

- Tarkistaa, onko syöte tyhjä. Jos syötettä ei ole, toiminto keskeytyy ja kutsuneelle proseduurille palautetaan virheilmoitus.
- Mahdolliset tekstisyötteen otsikkorivit ohitetaan.
- Kerää tmpCSV tiedoista ensimmäisen, toisen ja kahdeksannen (Etunimi, sukunimi, email) sarakkeen tiedot.
- Jos jäsentämisessä ei ilmennyt ongelmia, tiedot tallennetaan tietokantaan Users-tauluun ja lisäksi kirjoittaa lisätyn henkilön osallistujaksi Members tauluun.

15 Aikojenjakoalgoritmi `matchingAlgorithm.php`

Aikojenjakoalgoritmi laskee aikojenjakoehdotuksen järjestäjän tarjoamien aikojen ja osallistujien ilmoittamien sopivuuksien perusteella. Seuraavassa luvussa on kuvattu unkarilaisen algoritmin toimintaa tarkemmin. Komponentti käyttää `hungarian.php` tiedoston sisältämän `hungarian`-luokan palveluita hyväkseen.

- Tutkitaan onko aikataulu ryhmä vai henkilökohtaisia tapaamisia varten.
- Etsitään tietokannasta osallistujien ilmoittamat sopivuudet ja järjestäjän ilmoittamat ajat. Muodostetaan näiden tietojen perusteella kustannusmatriisi.
- Jos kyseessä on henkilökohtaisten tapaamisten aikataulu, ratkaistaan unkarilaisella menetelmällä paras aikojenjakoehdotus. Kirjoitetaan tulos tietokantaan.
- Jos kyseessä on ryhmäaikataulu, lasketaan mihin järjestäjän tarjoamaan aikaan on eniten ilmottautumisia. Kirjoitetaan tulos tietokantaan.

15.1 Palvelut

Komponentti tekee aikojenjakoehdotuksen arrange-palvelulla.

Esimerkki arrangen käytöstä: `$ok = arrange($id, $p);`

Parametri	Tietotyyppi	Pakollinen	Kuvaus
p	int	Kyllä	Viiteavain Groups-tilin tietueeseen.
id	int	Kyllä	Viiteavain käsiteltävään aikatauluun.

Palaute	Tyyppi	Kuvaus
ok	boolean	Palauttaa arvon false, jos järjestelyä tai sen tallentamista ei suoritettu loppuun. Jos järjestyshetken laskeminen onnistui ja ehdotus tallennettiin tietokantaan onnistuneesti palautetaan arvo true.

15.2 Toiminnallisuus

- Etsii ryhmän tiedot viiteavaimen perusteella, jos epäonnistui palautetaan arvo false.
- Etsii aikataulun tiedot viiteavaimen perusteella, jos epäonnistui palautetaan arvo false.
- Jos järjestäjä on lukinnut aikataulun, josta tieto saadaan Calendars-tilin locked kentästä, algoritmi ei laske uutta ehdotusta ja palauttaa arvon false.
- Hakee viiteavaimien perusteella aikataulun järjestäjän tarjoamat ajat, sekä osallistujien ilmoittamat sopivuudet. Muodostaa hakutuloksen perusteella kustannusmatriisin.
- Tutkii onko aikataulu ryhmä vai henkilökohtaisia tapaamisia varten.
- Jos kyseessä on henkilökohtaisten tapaamisten aikataulu, ratkaisee unkarilaisella menetelmällä parhaan aikojenjakoehdotuksen.
- Jos kyseessä on ryhmäaikataulu, lasketaan mihin järjestäjän tarjoamaan aikaan on eniten ilmoittautumisia.
- Kirjoittaa lasketun aikojenjakoehdotuksen Matches-tilin, booked kenttään.
- Jos suoritus loppui onnistuneesti palauttaa arvon true.

15.3 Tietokantakyselyt

select locked from Calendars where id=".\$id.
"select type from Calendars where id=".\$id.
select UserID from Members where GroupID=".\$parent."order by UserID
select id from offers where CalendarID=".\$id."order by id
SELECT Matches.UserID, Matches.OfferID, Matches.Pref FROM Matches, Offers where UserID in (SELECT UserID from Members where GroupID=".\$parent.") and OfferID=ID and CalendarID=".\$id."order by UserID,OfferID
update Matches set booked=now() where userid=".\$uid."and offerid=".\$oid
update Matches set booked=NULL where userid=".\$uid."and offerid=".\$oid."

16 Unkarilaisen metodin toiminta

Unkarilaisen metodin "paperi ja kynä"version kehitti vuonna 1955 H.W. Kuhn, vuonna 1957 J. Munkres kehitti algoritmin joka oli algoritmisesti suoritettavissa. Unkarilainen metodi etsii sellaisen kaksijakoisen verkon parituksen, jossa valittu paritus on (jos mahdollista) täydellinen ja parituksen kaarien summa on mahdollisimman pieni. Unkarilainen metodi käsittelee kaksiulotteista syötematriisia. Tässä luvussa kuvataan sanallisesti pääpiirteittäin, unkarilaisen metodin toiminta ja sen vaatimat alkutoimet.

Unkarilaisen algoritmin suoritus käynnistetään kutsumalla hungarian-luokan funktiota hungarianAlg. Funktion kutsu tapahtuu seuraavasti:

```
$result = $a->hungarianAlg($inputMatrix, $height, $width)
```

- \$result on matriisi joka sisältää parhaan parituksen. Matriisissa 1:sellä merkityt alkiot kuuluvat paritukseen 0:lla merkityt eivät.
- \$a on luokan hungarian ilmentymä.
- \$inputMatrix on tietokannasta koottujen tietojen perusteella muodostettu kustannusmatriisi.
- \$height on kustannusmatriisin korkeus.
- \$width on kustannusmatriisin leveys.

Kustannusmatriisin vaakarivillä ovat tarjotut ajat, pystysarakkeella osallistujat. Matriisin alkio vastaa osallistujan ilmoittamaa sopivuutta järjestäjän tarjoamaan aikaan.

esimerkki:

tarjotut ajat

$$\text{osallistujat} \begin{pmatrix} 1 & 3 & 3 \\ 2 & 2 & 4 \\ 4 & 1 & 4 \end{pmatrix}$$

16.1 Unkarilainen metodin suoritus. Alkutoimet

1. Jos matriisi ei ole neliömatriisi, täytyy matriisi täydentää neliömatriisiksi lisäämällä 0 rivejä tai 0 sarakkeita tarpeen mukaan.

Esimerkki:

$$\begin{pmatrix} 1 & 3 \\ 2 & 1 \\ 3 & 2 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 3 & 0 \\ 2 & 1 & 0 \\ 3 & 2 & 0 \end{pmatrix}$$

2. Muunnetaan kustannusmatriisia seuraavasti: muutetaan 0 painoiset kaaret 10 painoisiksi kaariksi.

Esimerkki:

$$\begin{pmatrix} 1 & 3 & 3 \\ 2 & 0 & 4 \\ 4 & 1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 3 & 3 \\ 2 & 10 & 4 \\ 4 & 1 & 10 \end{pmatrix}$$

16.2 Unkarilainen metodin suoritus. Toiminta pääpiirteittäin.

1. askel 1) Vähennetään jokaisen rivin alkion arvoa rivin pienimmällä arvolla.

esimerkki:

$$\begin{pmatrix} 1 & 3 & 3 \\ 2 & 2 & 4 \\ 4 & 1 & 4 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 2 & 2 \\ 0 & 0 & 2 \\ 3 & 0 & 3 \end{pmatrix}$$

2. askel 2) Etsitään matriisista riippumattomia 0-arvoisia alkioita ja merkitään ne tähdellä. Ei huomioida niitä nollia, joiden kanssa samalla rivillä tai sarakkeella on jo tähdellä merkitty 0. Tätä toistetaan, kunnes kaikki nollat on käsitelty. Käsitely etenee vasemmalta oikealle ja ylhäältä alas.

esimerkki:

$$\begin{pmatrix} 1 & 3 & 0 \\ 2 & 0 & 1 \\ 4 & 0 & 5 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 3 & 0* \\ 2 & 0* & 1 \\ 4 & 0 & 5 \end{pmatrix}$$

3. askel 3) Peitetään jokainen sarake, joka sisältää tähdellä merkityn nollan. Jos peitettyjä sarakkeita on matriisin leveyden verran, on löydetty ratkaisu.

-> siirrytään askeleeseen 7.

Jos 0*:ä on vähemmän kuin matriisin rivejä

-> siirrytään askeleeseen 4.

4. askel 4)

Etsitään peittämätön 0. Jos peittämätöntä nollaa ei löydy

->siirrytään askeleeseen 6.

Jos peittämätön nolla löytyy.

-> merkitään löydetty 0 heittomerkillä ('), kutsutaan tätä alkioita P0:aksi.

Jos samalla rivillä P0:n kanssa on tähdellä merkitty 0

-> poistetaan tähdellä merkityn nollan kohdalta sarakkeite ja lisätään P0:n riville rivipeite ja toistetaan askel 4.

Jos P0:n rivillä ei ole tähdellä merkittyä 0:aa

->siirrytään vaiheeseen 5, jolle annetaan parametrina löydetyn P0:n sijainti.

5. askel 5)

Etsitään matriisista vaihteleva polku jonka alkioina vaihtelevat 0* ja 0' alkioita. Lähdetään liikkeelle vaiheessa 4 löydetystä P0:sta. Etsitään tähdellä merkitty nolla S0 samalta sarakkeelta P0:n kanssa. Sitten etsitään S1:n riviltä 0':a

P_1 , tämän jälkeen etsitään P_1 :n sarakkeelta taas tähdellä merkittyä nollaa jne. On siis löydetty polku $P_0, S_0, P_1, S_1, \dots, P_n$. Poistetaan tähdellä merkityistä nolista tähtimerkintä ja muutetaan kaikki 0:t tähdellä merkityiksi. Lopuksi poistetaan kaikki ' merkinnät sekä poistetaan kaikki peitteet ja siirrytään vaiheeseen 3.

6. askel 6) Etsitään matriisin pienin peittämätön alkio. Lisätään pienin peittämätön alkio kaikkiin kahdesti peitettyihin alkioihin (siis ne alkiot joita peittää sekä rivi- että sarakepeite). Vähennetään kaikista peittämättömistä alkiosta pienimmän alkion arvo. Siirrytään vaiheeseen 4.

7. askel 7)

Matriisin tähdellä merkittyjen nollien kohdalla ovat ne kaaret jotka sisältyvät paritukseen. Lasketaan ratkaisun yhteispaino `alteredCost` matriisissa ja lasketaan kyseisen parituksen kaarien arvojen hajonta. Jos parituksen yhteispaino on pienempi tai yhtäsuuri kuin `best-muuttujassa` oleva ja hajonta on pienempi kuin `bestVariation` muuttujassa oleva, tallennetaan paritus `bestMask` muuttujaan. Kasvatetaan `count`-muuttujan arvoa yhdellä. Jos `count`-muuttujan arvo on kolme palautetaan löydetty paras paritus ja lopetetaan, muuten aloitetaan uusi kierros unkarilaisen menetelmän suorittamista paremman ratkaisun löytämiseksi.

16.2.1 Unkarilaisen metodissa käytettävät tietorakenteet

Tässä luvussa kuvataan, tärkeimmät tietorakenteet, mitä Unkarilaisen algoritmin toteuttamiseen tarvitaan.

Nimi	Tyyppi	Kuvaus
originalCost	int[][]	Kustannusmatriisi, joka on muodostettu tietokantakyselyn avulla. Ei välttämättä neliömatriisi.
augmentedCost	int[][]	Neliömatriisiksi tydensetty kustannusmatriisi. 0 arvoiset alkiot muutettu 10 arvoisiksi.
cost	int[][]	Kustannusmatriisi jota unkarilaisen metodin funktiot käyttävät, muodostettu aluksi augmentedCost matriisin perusteella. Toisen kierroksen alkaessa cost matriisin arvo luetaan alteredCost matriisista.
alteredCost	int[][]	Matriisi jonka yhtä alkion arvoa on muutettu 5:seksi uuden parituksen löytämisen takaamiseksi, jokaisen unkarilaisen metodin suorituskierron jälkeen. cost matriisin arvo luetaan tästä matriisista ensimmäisen suorituskierron jälkeen.
mask	int[][]	Matriisi johon merkitään 1:llä tähdellä merkityt nollat ja 2:lla heittomerkillä merkityt nollat. Saman kokoinen kuin cost matriisi.
rowCover	int[]	Yhtä pitkä kuin kustannusmatriisi cost on korkea. Merkitään rivin indeksin kohtaan 1, jos rivillä on peite. Jos rivillä ei ole peitettä merkitään 0.
colCover	int[]	Yhtä pitkä kuin kustannusmatriisi on leveä. Merkitään sarakkeen indeksin kohtaan 1, jos sarake on peitetty. Jos sarake ei ole peitetty merkitään 0.
step	int	Kertoo missä askeleessa unkarilainen algoritmi on menossa.
xP0	int	P0:n x-koordinaatti. Arvo asetetaan askeleessa 4 ja tietoa käytetään askeleessa 5.
yP0	int	P0:n y-koordinaatti. Arvo asetetaan askeleessa 4 ja tietoa käytetään askeleessa 5.
count	int	Kertoo montako kierrosta unkarilaista algoritmia on suoritettu.
best	int	Löydetyin Parhaan löydetyin parituksen painojen summa.
bestVariation	float	Parhaan löydetyin parituksen arvojen hajonta.
bestmaskcount	int	Parhaan löydetyin parituksen mask-matriisi. Palautetaan

16.2.2 hajonnan laskeminen

Jos parituksen yhteispainojen perusteella ei pystytä tekemään eroa kahden aikojen välillä, käytetään paremmuuden vertailussa parituksien hajontaa vertailuperusteena. Mitä pienempi hajonta on sitä parempi. Hajonta lasketaan seuraavasti:

- Lasketaan paritukseen kuuluvien kaarien arvojen keskiarvo.
- Lasketaan kaarien arvojen poikkeama keskiarvosta.

17 Käyttöliittymäkuvaus

17.1 Yleiskuvaus

Käyttöliittymäsuunnittelussa on pyritty toimimaan hyvien käyttöliittymäsuunnitteluperiaatteiden mukaisesti. Käyttöliittymä on suunniteltu web-selaimella toimivaksi. Eri selainten väliset erot käyttöliittymän piirtämisessä pyritään minimoimaan noudattamalla käyttöliittymän toteutuksessa yleisiä standardeja (CSS1, DTD, HTML 4.01).

Dokumentin lopussa on kuvattu käyttöliittymän keskeisimmät sivut näyttökuvina.

17.1.1 Sivun metatiedot

Jokaisen sivun `<head>`-elementtien väliin kirjoitetaan sivun metatiedot

- `<meta http-equiv="content-language" content="<?php echo $lang ?>">`
- `<meta http-equiv="content-type" content="text/html; charset=iso-8859-1">`
- `<meta name="description" content="Aija - <?php echo localText($lang, 'TITLE') ?>">`
- `<meta name="keywords" content="Aija">`
- `<meta name="robots" content="index, follow">`
- `<meta name="robots" content="noarchive">`
- `<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">`

Osa metatiedoista on yhteisiä kaikille sivuille ja nämä on tallennettu `meta.php`-komponenttiin `inc`-kansiossa. Tämä komponentti liitetään osaksi käyttöliittymää korvaamalla staattiset metatiedot direktiivillä:

```
include "inc/meta.php";
```

17.1.2 HTML:n valmiiden elementtien käyttäminen

Seuraavassa on kuvattu HTML-elementtien käyttö käyttöliittymässä. Tavallinen teksti on aina kappale-elementtien sisällä olevaa (`<p></p>`). Lomakkeen kenttien otsakkeet ovat aina normaalia lihavoitua (``) tekstiä.

Kuvan 16 numeroinnin selitykset

Numero	Käytetty elementti
1.	<code><h5></code>
2.	<code><h3></code>

Kuvan 17 numeroinnin selitykset

Numero	Käytetty elementti
1.	<code><code></code>

17.1.3 Komponenttien asemointi käyttöliittymässä

Komponenttien asemointi käyttöliittymässä on kuvattu alla olevissa kuvissa.

17.1.4 CSS-tyylimäärittelyjen käyttö

CSS-tyylimäärittelyjä käytetään käyttöliittymän niissä osissa, joissa ei käytetä HTML:n valmiita elementtejä.

Kaikille komponenteille yhteiset tyylimäärittelyt

Elementti	Käyttökohde
<code>.bgpref0</code>	Sopivuuden 0 taustaväri
<code>.fspref0</code>	Sopivuuden 0 fontti
<code>.bgpref1</code>	Sopivuuden 1 taustaväri
<code>.fspref1</code>	Sopivuuden 1 fontti
<code>.bgpref2</code>	Sopivuuden 2 taustaväri
<code>.fspref2</code>	Sopivuuden 2 fontti
<code>.bgpref3</code>	Sopivuuden 3 taustaväri
<code>.fspref3</code>	Sopivuuden 3 fontti
<code>.bgpref4</code>	Sopivuuden 4 taustaväri
<code>.fspref4</code>	Sopivuuden 4 fontti

Reunalistakomponentin tyylimäärittelyt

Elementti	Kuvaus
.bgsidepane	Reunalistan taustaväri
.fssidepane	Reunalistan tekstin koko
a.sidepane	Reunalistan linkkitekstin väri
a.sidepane:active	Reunalistan aktiivisen linkkitekstin väri
a.sidepane:visited	Reunalistan vierailun linkkitekstin väri

Viestinlähetyskomponentin tyylimäärittelyt

Elementti	Kuvaus
.bgfail	Virheviestin taustaväri
.bgwarn	Varoitusviestin taustaväri
.bginfo	Tiedotusviestin taustaväri
.bgconfirm	Varmistusviestin taustaväri
.fsmessage	Viestin fontti

Viikkonäyttökomenttien tyylimäärittelyt

Elementti	Kuvaus
.bgweekcalendar	Viikkonäytön taustaväri
.bgday	Viikonpäivän taustaväri
.fsday	Viikonpäivän fontti
.bgtime	Kellonajan taustaväri
.fstime	Kellonajan fontti
.fscell	Solujen tekstifontti
.bgempty	Tyhjän solun taustaväri järjestäjän luodessa aikataulua
.bgoffer	Tarjotun ajan taustaväri
.bgnotime	Tyhjä solu osallistujan viikkonäytöllä

Järjestäjän järjestelynäytön tyylimäärittelyt

Järjestelijän järjestelynäytössä piirretään myös viikkonäyttö, joten kaikki viikkonäyttökomenttien tyylimäärittelyt ovat käytettävissä myös tälle komponentille.

Elementti	Kuvaus
.borReserved	Reunamäärittely solulle, jossa osallistujalla on jo varattu aika.
.borBooked	Reunamäärittely solulle, jossa solulla on päällekkäisvaraus
.bgmember	Määrittely solulle, jossa on osallistujan nimi
.fsmember	Määrittely tekstille solussa, jossa on osallistujan nimi

Kalenterikomponentin tyylimäärittelyt

Elementti	Kuvaus
.bgcalendar	Kalenterin taustaväri
.bgcurrdays	Meneillään olevan päivän taustaväri
.weekseparator	Viikkojen välissä olevan erotinviihan määrittely
.bgchosenday	Taustaväri päivälle, joka on valittuna aikatauluun kalenterissa
.bgday	Taustaväri päivälle, joka ei sisälly valittuun viikkoon kalenterissa
.fsday	Päiväsolujen tekstifontti
.bgmonth	Kuukausitekstin solun taustaväri
.fsmonth	Kuukausitekstin fontti

18 Testaussuunnitelma

18.1 Johdanto

Tämän dokumentin tarkoitus on määritellä testaukseen ja laadunarviointiin liittyviä keskeisiä kysymyksiä ja luoda suuntaviivat järjestelmälliselle ohjelmistotestaukselle, jota tullaan suorittamaan toteutusvaiheen aikana ja sen jälkeen.

Tuotettava ohjelmisto, Aija -aikojenjakojärjestelmä, on Helsingin yliopiston, tietojenkäsittelytieteen laitoksentietojenkäsittelytieteen opintoihin kuuluvan ohjelmistotuotantoprojektin (60v) tuote. On tarkoituksenmukaista määritellä testiaineisto ja testausprosessi huolella, jotta mahdollisimman moni virheistä pystytään havaitsemaan, paikallistamaan ja korjaamaan ennen ohjelmiston julkaisua. Testijärjestelyjen on huomioitava projektityöhön liittyvä ajallisten resurssien niukkuus. Mahdollisimman kattavaan lopputulokseen pyritään systematisoimalla testauskäytännöt ja määrittämällä testiaineisto, joka kattaa mahdollisimman monta toisistaan poikkeavaa syötetyyppeä/tapausta.

Olisi epärealistista olettaa, että ohjelmiston kaikki virheet voitaisiin testausprosessissa havaita ja korjata. Testausprosessin päämääränä voidaankin katsoa olevan ennen kaikkea systemaattinen ja tehokas vianetsintä, joka löytää (ja korjaa) mahdollisimman paljon virheitä testaukselle varatussa ajassa.

18.2 Testauksen kohde ja tavoitteet

Testauksen kohteena on Aija-aikojenjakojärjestelmä, jonka tarkempi kuvaus löytyy määrittely- ja suunnitteludokumenteista. Aija-järjestelmä voidaan jakaa neljään peruskomponenttiin: käyttöliittymä, sovelluslogiikka, tietokantarajapinta ja sähköpostirajapinta. Lisäksi järjestelmään kuuluu joitakin automaattisesti suoritettavia ajastettuja toimintoja, jotka hyödyntävät toiminnassaan sovelluslogiikkaa sekä tietokanta- ja sähköpostirajapintoja.

Testauksen tavoitteena on pyrkiä todentamaan ohjelmiston oikeellinen, vaatimus-

määrittelyn mukainen toiminta. Tämä tarkoittaa sitä, että vaatimusmäärittelyn käyttötapausten esittämät syötteet tuottavat aina niitä vastaavat vaatimusmäärittelyssä määritellyt tulosteet.

Pelkkä tulosteiden oikeellisuus on kuitenkin vain oikeellisen toiminnan suppea osa-alue. On syytä kiinnittää huomiota myös järjestelmän vakauteen, vikasietoisuuteen, käyttäjäystävällisyyteen ja luotettavuuteen.

Vakaudella tarkoitetaan tässä yleisesti ohjelmiston ominaisuutta säilyttää toimintakuntoisuutensa käyttöajan kasvaessa; vikasietoisuus käsittää ohjelmiston kyvyn reagoida oikein esimerkiksi virheellisiin syötteisiin tai pahantahtoiseen käyttäjään käyttäjäystävällisyys kattaa ohjelmiston kyvyn ennakoida käyttäjän tarpeita ja kommunikoida riittävällä tarkkuudella käyttäjän kanssa esimerkiksi virhetapauksissa luotettavuudella tarkoitetaan ohjelmiston sisäisen logiikan koherenssia tulosteina tuotetun informaation oikeellisuutta ja luotettavuutta.

18.3 Testausympäristö

Testausympäristönä toimii pääasiassa ohjelmiston kehitysympäristö, Helsingin yliopiston tietojenkäsittelytieteen laitoksen tietokantapalvelin alkokrunni.

18.4 Testausstrategia

Ohjelmiston järjestelmällinen testaus sisältää kolme vaihetta: yksikkötestaus, integrointitestaus ja validointitestaus. Yksikkötestauksessa jokaisen ohjelmistoon liittyvän komponentin sisäinen toiminta testataan itsenäisesti. Integrointitestauksessa keskitytään tarkastelemaan komponenttien välisten suhteiden toimivuutta. Ohjelmiston vaatimusmäärittelyä vastaava toiminnallisuus käyttöliittymätasolla testataan validointitestauksessa

Yksikkötestauksessa käytetään kunkin komponentin kohdalla sen ohjelmoijan määrittelemää testiaineistoa ja -menetelmää. Tarkoituksena on pyrkiä yksikkötestauksessa haaraumakattavuuteen. Erityistä huomiota tulee kiinnittää ohjelman toistolauseita täsmentäviin ehtoihin ja ehtojen rajatilanteisiin. Myös yksikön sisällä käytettävät tietorakenteet tulee testata huolellisesti.

Integrointitestauksessa pyritään verifioimaan luokkien ja komponenttien välisten rajapintojen toiminnallisuus. Tarkastelun kohteena ovat erityisesti rajapintafunktiolle toimitettavat syötteet ja niiden palautusarvot, sekä rajapinnan yhdistämien moduulien käyttäytyminen ja kommunikaatio toisesta osapuolesta johtuvissa virhetilanteissa. Yhtenäisen ja luotettavan virheidenkäsittelyn todentaminen on eräs integrointitestauksen tavoitteista.

Validointitestauksessa osajärjestelmien muodostamaa kokonaisuutta, ohjelmistotuotetta, testataan käyttäjän näkökulmasta. Testiaineistona käytetään käsin tai automatisoidusti tuotettua aineistoa. Osan testausta varten tuotetusta aineistosta tulee olla virheellistä tai vajavaista, osan vastata määrittelydokumentissa esitettyjä oi-

keellisuusnormeja. Validointitestauksen tarkoituksena on löytää epäselviä tilanteita tai suoranaisia virheitä, joilla on vaikutusta käyttäjän työnkulkuun.

Jotta testaus olisi systemaattista, testauksen edistymisestä pidetään yllä testauspäiväkirjaa, joka sisältää tiedon löydetystä virheistä löytäneen ja löytöaikoineen, tiedon käytetystä testistä ja mahdollisista korjausehdotuksista sekä virheen tarkemman kuvauksen.

18.4.1 Ohjelmakoodissa havaitut puutteellisuudet ja virheellinen toiminnallisuus

Yksikkötestauksen aikana havaitut virheet pyritään korjaamaan välittömästi. Integraatiotestauksen aikana havaitut virheet raportoidaan ryhmälle ja komponentin vastuuhenkilölle. Vastuuhenkilön tulee korjata komponentin toiminnallisuus vastaamaan suunnitteludokumentissa esitettyä ja raportoida tehty korjaus. Integroitintestauksen raportit liitetään testausdokumentin liitteeksi.

Validointitestauksen aikana havaitut virheet raportoidaan testausdokumenttiin ja ne tulee korjata sekä testi uusia mahdollisuuksien mukaan.

18.5 Testauksen organisointi ja raportointi

Yksikkötestauksen testausvastuu on jaettu projektiryhmäläisten vastuualueiden mukaan. Jokainen ryhmän jäsen huolehtii oman ohjelmakoodinsa oikeellisuudesta ja toteuttamiensa komponenttien sisäisestä koherenssista. Yksikkötestaus suoritetaan ohjelmoinnin ohessa ja testausprosessi raportoidaan vain yleisellä tasolla. Komponentin ollessa valmis, eli kun se toteuttaa määrittelydokumentissa sille esitetyt toiminnallisuusvaatimukset, tulee komponentin vastuuhenkilön liittää testausdokumenttiin maininta tästä. Testaus perustuu haaraumakattavuuden menetelmään. Menetelmää varten lähdekoodin metodeista tehdään vuokaavio seuraavasti:

- Kaaviolla on alkusolmu, joka vastaa metodiin tuloa (metodin otsikkoa). Alkusolmuun ei tule särmiä.
- Kaaviolla on joukko loppusolmuja, jotka vastaavat metodista poistumisia (return-lauseita). Loppusolmusta ei lähde särmiä
- Jokaisesta metodin lauseesta tulee solmu.
- Jokaisesta mahdollisesta siirtymästä lauseesta toiseen tulee särmä vastaavien lauseiden solmujen välille.
- Korkean tason rakenteet voivat vaatia useita solmuja ja särmiä.
- Ehtojen atomiset lausekkeet voidaan erotella omiksi solmuiksi. Särmit lausekesolmujen välillä kuvaavat ehtojen suoritusjärjestystä.

- Jos kaaviossa on joukko peräkkäisiä solmuja, joissa ei ole haaraumia, nämä voidaan koota yhteen segmentiksi.
- Jos kaaviossa on monta loppusolmua, näistä kannattaa liittää särmät virtuaaliseen metodin loppusomuun L.
- Korkean tason rakenteiden loppumista voidaan kuvata virtuaalisella rakenteen loppusolmulla E.

Haaraumakattavuuden kattavuuskriteeri määrittelee, kuinka suuressa osassa metodin vuokaavion särmiä täytyy käydä testuksessa. Haaraumakattavuuskriteerille riittää, että joillain testitapauksilla saadaan käytyä läpi kaikki haaraumat. Jos haaraumaan, kuten if- lauseeseen, liittyy monta ehtoa, niin haaraumakattavuuskriteeri vaatii vain kaksi ehtojen kombinaatiota: yhden testin true-haaralle ja yhden false-haaralle.

Yksikkötestausraporttien avulla pyritään lähinnä tiedottamaan muille projektiryhmäläisille, että raporttiin liittyvä komponentti on läpäissyt vaaditut yksikkötestit ja valmis integroitavaksi järjestelmään.

Integrointitestausta tullaan suorittamaan sitä mukaan kun se on mahdollista. Varsinaista integrointitestitapausta ei kuitenkaan voida suorittaa hyväksyttävästi ennen kuin kaikista testitapaukseen liittyvistä komponenteista on olemassa valmistumisraportti. Jokaisesta integrointitestitapausten suorittamisesta liitetään testausdokumenttiin raportti. Mikäli komponentissa tapahtuu muutoksia, on niihin liittyvät testitapaukset soveltuvilta osin suoritettava uudelleen ja olemassa olevia raportteja päivitettävä. Integrointitestauksen vastuualueet pyritään jakamaan siten, ettei kukaan ryhmän jäsen testaa (ainoana henkilönä) itse tuottamiaan komponentteja.

Validointitestaukseen osallistuvat kaikki ryhmän jäsenet. Kukin suoritettu testitapaus tulee raportoida tässä dokumentissa esitetyn mallin mukaisesti. Raporttien tulee sisältää testitapausten tunniste, käytetty aineisto, havaitut virheet tai epäselvyydet. Ryhmän yksittäisten jäsenten raporteista kootaan lopuksi testausdokumentti, jossa tulokset ja ohjelmiston toiminnassa ilmenneet ongelmatilanteet käydään läpi.

Testausdokumentin perusteella voidaan arvioida ohjelmistoprosessin määrittely-, suunnittelu- ja toteutusvaiheen laadukkuutta. Mikäli merkittäviä virheitä tai puutteita löytyy paljon, voi tällä olla merkitystä koko ohjelmiston rakenteen suhteen. Silloin on syytä tarkastella kriittisesti ohjelmistoa ja määrittellä sen luotettavuuden, käytettävyyden ja laadukkuuden aste uudelleen. Toisaalta testausvaiheen suurten virheiden puute voi viitata, paitsi ohjelmiston laadukkuuteen, myös testausprosessin riittämättömyyteen. Jälkimmäinen vaihtoehto on syytä pyrkiä eliminoimaan huolellisella testitapausten määrittelyllä ja testaustulosten tarkalla raportoinnilla.

18.6 Testausvaiheet

Yksikkötestaus ja integrointitestausta ovat testauksen osa-alueita, joiden puitteissa voidaan suorittaa ohjelmistotestausta jo toteutusvaiheen alusta lähtien.

Tuotanto- ja testausprosessin loppupuolella testaus tulee painottumaan integrointi- ja validointitestaukseen. Jälkimmäinen tapahtuu joitakin poikkeuksia lukuun ottamatta käyttöliittymäsivujen kautta.

18.7 Yksikkötestaus

Yksikkötestauksen pyrkimyksenä on todentaa ohjelmiston komponenttien sisäinen toimivuus. Tarkastelun kohteena on funktioiden logiikka, käytettyjen tietorakenteiden toimivuus. Luokkatestaus on vastuupohjaista, eli luokka tarjoaa joukon palveluita ja on vastuussa siitä, että palvelut toimivat oikein. Palveluiden toiminnan ja poikkeustilanteiden hallinnan varmistus on luokkatestauksen ydin. Vastuupohjaisen testauksen tukena käytetään toteutus pohjaista testausta, jossa testitapaukset johdetaan testattavan luokan toteutuksesta.

Yksikkötestausta suunniteltaessa päädyttiin siihen, että yksikön (komponentin) testausvastuu on komponentin kirjoittajan. Kirjoittaja määrittelee itse testiaineiston, mutta testien haaraumakattavuuden vaatimus on oleellinen. On mahdollista, että monimutkaisten tai järjestelmän toiminnalle keskeisten komponenttien yksikkötestaukseen osallistuu myös toinen projektiryhmäläinen. Ideaalitapauksessa yksittäisen komponentin yksikkötestauksen suorittajia on useita, eivätkä komponentin kirjoittajat osallistu oman ohjelmakoodinsa testaamiseen, mutta projektin käytettävissä oleva aika ei ehkä riitä tähän.

Testausraportoinnin painopiste on integrointi- ja validointitestauksen raportoinnissa. Yksikkötestaus tullaan raportoimaan hyvin yleisellä tasolla, lukuun ottamatta niitä komponentteja, jotka ovat järjestelmän kannalta keskeisiä (esim. tietokanta, algoritmin toiminta). Ne puutteet, jotka havaitaan testeissä, mutta projektin päättymisen vuoksi jäävät korjaamatta, raportoidaan tarkasti testausdokumentissä.

18.7.1 Integrointitestaus

Integrointivaiheessa olemassa olevat komponentit kootaan kokonaisuudeksi, lopulliseksi ohjelmistoksi. Integrointitestauksen tavoitteena on todentaa yhdistettävien ohjelmistokomponenttien välisen kommunikaation oikeellisuus ja yhteensopivuus, suunnitteludokumentissa esitetyn perusteella.

Aijan integrointitestauksessa käytetään ns. bottom-up lähestymistapaa. Tämän metodin lähtökohtana on ohjelmistokokonaisuuden toiminnallisuuden testaus sen primitiivisimmistä komponenteista lähtien, toteutuksen alimmalta tasolta korkeimmalle edeten. Komponenttien rajapintoja testataan ajureilla. Nämä simuloivat korkeamman tason komponentteja, joihin alemman tason komponentit yhdistyvät. Ajurit kutsuvat rajapintafunktioita testitapausten määrittelemien parametrein ja tarkkailevat funktiokutsujen palautusarvoja.

Bottom-up - testausmenetelmän käytöstä seuraa, että ohjelmointivirheitä sisältävät komponentit on mahdollista paikallistaa verrattain luotettavasti. Mikäli testitapauk-

sessä määritelty syöte ei tuota odotettua tulostetta, on vika suurella todennäköisyydellä vain liitettyssä komponentissa, ei sen alapuolisessa hierarkiassa.

Kukin suoritettu testitapaus tuloksineen tulee kirjata testausdokumenttiin. Tapaukset, joissa ohjelmiston käyttäytyminen poikkesi oletetusta, otetaan lähempään tarkasteluun ja niihin liittyvät testit suoritetaan korjausten jälkeen uudelleen.

18.7.2 Komponenttien yhdistäminen

Koska integrointitestausta painottuu toteutusvaiheen loppupuolelle, tullaan käytännön testausta suorittamaan myös olemassa olevista luokista käsin. Yleensä tällöin kuitenkin rajoitetaan hyvälaatuisiin testitapauksiin. Jotta komponenttihierarkia (tai sen osa) olisi validi, tulee kaikkien hierarkian juuren lasten rajapinnat kuitenkin testata ajureiden avulla.

Integrointitestausta tavoitteena on suunnitteludokumentissa esitetty toimiva järjestelmäkokonaisuus, joka on koottu ainoastaan testatuista komponenteista. Yksittäisen komponentin integrointitestausta voidaan katsoa suoritetuksi, kun kaikki komponenttiin liittyvät testitapaukset on suoritettu siten, että syöte/tuloste-parit vastaavat testitapauksessa esitettyjä. Tämän jälkeen komponentti voidaan liittää osaksi järjestelmää.

Mikäli toiminnallisuudessa havaitaan virhe, tulee virheen tuottaneen komponentin vastuuhenkilön tai henkilöiden paikallistaa ja korjata vika. Tämän jälkeen komponentin integrointitestausta suoritetaan soveltuvilta osin uudelleen.

18.8 Testattavat rajapintafunktiot

Tarkoitus on keskittyä paitsi oikeellisilla arvoilla suoritettuihin testeihin, myös sellaisiin tapauksiin, jotka häilyvät oikeellisuuden rajoilla. Tällaisia ovat esimerkiksi pienet poikkeamat välitettyjen olioiden sisällössä ja tyhjäarvoiset parametrit.

Osa rajapinnoista testataan myös selvästi virheellisillä syötteillä, kuten tietotyypiltään tai tietosisällöltään vääränlaisilla parametreilla. Pääsääntöisesti tällaisia tapauksia ei järjestelmässä voi esiintyä, sillä käyttäjäsyötteet suodatetaan jo käyttöliittymäsivujen puolella.

18.8.1 Validointitestausta

Validointitestausta on ohjelmistotuotteen testausta loppukäyttäjän näkökulmasta. Validointitestausta tarkoituksena on varmistaa, että käyttäjän välittämä ja käyttäjälle välittyvä tieto vastaa määrittelydokumentin esittämiä vaatimuksia. Lisäksi pyritään todentamaan järjestelmän luotettava, tai ainakin ennakoitava toiminta virhetilanteissa sekä virhetilanteesta kertovat viestit. Validointitestausta suoritetaan pääasiassa käyttötapauksien perusteella, jolloin eri käyttötapauksiin liittyvinä syöteinä käytetään sekä oikeellisia, määrittelydokumentissa esitettyjä, että virheellisiä,

muodoltaan tai kattavuudeltaan vajavaisia syötteitä. Myös määrittelydokumentin sisällöstä poikkeavia, selvästi 201dvirheellisiä 201d käyttötapauksia testataan. Esimerkiksi käyttäjän yritykset korvata välitettävät parametriarvot jollakin muulla tiedolla.

Validointitestauksen perustana toimivat määrittelydokumentin käyttötapaukset ja suunnitteludokumentin käyttöliittymäsuunnitelma. Jokainen käyttötapaus tulee voida suorittaa onnistuneesti, jotta käyttöliittymäkomponentin testaustulos olisi hyväksyvä. Lisäksi järjestelmän on pysyttävänä vakaana myös siinä tapauksessa, että käyttäjäsyötteet poikkeava odotetusta.

Jokaisen käyttöliittymäsivun tulee läpäistä seuraavat alkeistestit:

- Vaadittu staattinen tieto tulee näkyä käyttöliittymäsivulla
- Kaikkien suunnitteludokumentissa esitettyjen syötekenttien tulee olla olemassa sivuilla (ks. määrittely- ja suunnitteludokumentti)
- Syötekenttiin syötetyn tiedon tietotyyppi tulee tarkistaa ennen tiedon välittämistä eteenpäin järjestelmässä (JavaScript tai parametrit tarkistava php-koodi).
- Tietokantasensitiivisten merkkien suodattaminen tai varustaminen etuliitteellä.
- Välitettävien tietotyyppien tulee vastata suunnitteludokumentin rajapintamäärittelyssä esitettyä.
- Järjestelmän täytyy käyttäytyä tasapainoisesti tapauksissa, joissa yksi tai useampi syötekenttä on jätetty tyhjäksi (kentän sisältö voi olla oleellista tietoa, mutta ei välttämättä).
- Virhetilanteet ja niiden syyt tulee kertoa käyttäjälle selvästi

18.9 Järjestelmätestaus

Järjestelmätestaus tehdään integrointitestauksen jälkeen. Siinä järjestelmää testataan kokonaisuutena. Ennen järjestelmätestausta kaikki järjestelmään kuuluvat komponentit on testattu erikseen. Järjestelmätestaus on vastuupohjaista ja testaus tapahtuu käyttöliittymän kautta.

18.9.1 Testausmenetelmät

Testausmenetelmänä käytetään Extended Use Case Testiä (EUCT), joka käyttää laajennettuja käyttötapauksia. Tavalliset käyttötapaukset määrittelevät toimintatavat, osallistuvat sidosryhmät ja käyttötapaukseen vaikuttavat syöte- ja tulostiedot. Laajennetut käyttötapaukset määrittelevät lisäksi:

- käyttötapaukseen liittyvät muuttujat arvoalueineen
- käyttötapauksen syöte- ja tulostietojen suhteet
- käyttötapauksen esiintymistiheyden suhteessa muihin käyttötapauksiin
- käyttötapauksien keskinäisen suoritusjärjestyksen

Käyttötapauksien syötteiden ja tulosten yhdistelmistä rakennetaan päätöstaulu (decision table). Jokainen päätöstaulun rivi kertoo yhden toimintatavan syötteet ja tulokset. Jokainen sarake kertoo käyttötapauksen yhden syötteen tai tuloksen nimen ja tyyppin.

Päätöstauluesimerkki

N:o	Salasana	Syötetty salasana	Järjestelmän kuittaus	Tila	Vastaus viesti	Järjestelmän toiminta
1	eino007				Anna salasana	Salasana-kenttä täytettäväsi
2	eino007	eino007	ok	open	Tervetuloa Aija-järjestelmään!	Vie oikeuksien mukaiselle sivulle
3	eino007	007eino		clsd	Virheellinen salasana!	Salasana-kenttä täytettäväsi
4	eino007	eino007		clsd	Ei yhteyttä tietokantaan!	Kirjautuminen estyy.

Testipaketti sisältää yhden true- ja yhden false-testin jokaiselle päätöstaulun riville.

- True testissä kaikki ehdot ovat tosia
- False testissä ainakin yksi ehdoista on epätosi
- False testit saadaan yleensä päätöstaulun jonkin muun rivin testitapauksista

Extended Use Case Testiä varten täytyy seuraavien ehtojen täytyä:

Aloitusehdot.

- Käyttötapauksista on johdettu laajennetut käyttötapaukset
- Integroititestausta on saatu päätökseen
- Jokaisen toteutetun toiminnan täytyy sisältyä ainakin yhteen käyttötapaukseen

Lopetusehdot:

- Kaikista testikäyttötapauksista on päätöstaulut

- Jokaisen päätöstaulun jokaiselle riville on tehty vähintään true- ja false- testi

Covered in CRUD täydentää EUCT:n testitapauksia sellaisissa tilanteissa, missä käyttötapaukset eivät kata kaikkia testattavan järjestelmän syöte- ja tulosolioiden perusoperaatioita. Perusoperaatioita ovat luonti (C), luku (R), päivitys (U) ja poisto (D).

18.9.2 Testauksen resurssointi

Käytännössä täysin kattava testaus johtaa liian laajaa testipakettiin, jota ei ole mahdollista tämän projektin puitteissa toteuttaa. Resursseja varataan kullekin käyttötapaukselle suhteessa sen esiintymistodennäköisyyteen ja haittavaikutuksen suuruuteen.

18.9.3 Muut testattavat alueet

Toiminnallisten vaatimusten jälkeen testataan ei-toiminnalliset vaatimukset, jotka ovat järjestelmän laatuvaatimuksia. Seuraavat ei-toiminnalliset vaatimukset tulee ainakin testata:

- Yhteensopivuusvaatimukset kertovat, että järjestelmä toimii saumattomasti sille tarkoitettussa ympäristössä.
- Suorituskykyvaatimukset pitävät sisällään kuormitustestauksen, paljoustestauksen ja rasiustestauksen
- Eheys- ja vikasietoisuusvaatimukset
- Käytettävyysvaatimukset

Aija - aikojenjakojärjestelmä
[Ajantekijän sivut](#)

Kotisivut
[Ajantekijä](#)

Kirjaudu sisään

Kirjoita alla oleviin kenttiin sähköpostiosoitteesi ja henkilökohtainen salasanasi. Paina lopuksi **Kirjaudu sisään** -painiketta.

Sähköpostiosoitteesi:

Salasanasi:

Järjestelmän etusivu

Rekisteröidy käyttäjäksi

Kirjaudu järjestelmän käyttäjäksi sähköpostiosoitteellasi. Jos haluat ylläpitää ryhmää ja aikatauluja, on sähköpostiosoitteesi kuuluttava johonkin seuraavista toimialueista: [cs.helsinki.fi](#), [helsinki.fi](#). Voit tarvittaessa vaihtaa sähköpostiosoitteesi rekisteröitymisen jälkeen. Kun olet valmis, paina **Rekisteröidy**-painiketta. Lähetämme sinulle salasanan antamaasi sähköpostiosoitteeseen.

Sähköpostiosoitteesi:

Unohtunut salasana

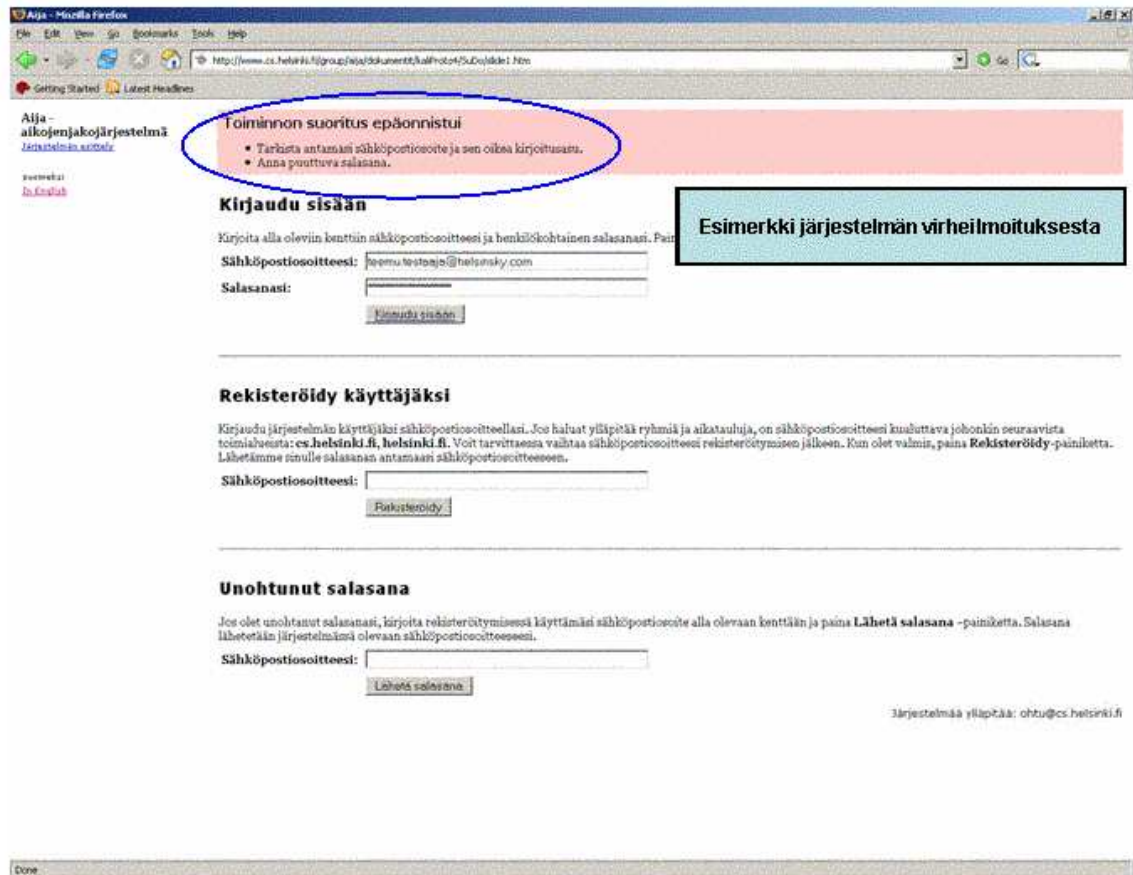
Jos olet unohtanut salasanasi, kirjoita rekisteröitymisesi käyttämäsi sähköpostiosoite alla olevaan kenttään ja paina **Lähetä salasana** -painiketta. Salasana lähetetään järjestelmässä olevaan sähköpostiosoitteeseen.

Sähköpostiosoitteesi:

Järjestelmää ylläpitää: ottu@cs.helsinki.fi

Done

Kuva 7: Etusivu.



Kuva 8: Virheilmoitus.

The screenshot shows a web browser window displaying a scheduling application. The page title is "Ajaja - Uusi aikataulu - Mozilla Firefox". The address bar shows the URL: "http://www.cshelsinki.fi/group/veljokunta/3allf4c204752c9jaka1.htm".

On the left, there is a sidebar with the heading "Ajaja - aikojenjakojärjestelmä". Below it are several navigation links: "Tulokset", "Ajaja", "Musiikkitiedot", "Osallistujat", "Uusi rajaus", "Kassa", "Ilmoitus", "Musiikkilista", "Ajankäytöt", "Lisäpalvelut", "Suomeksi", and "In English".

The main content area contains a form with the following fields:

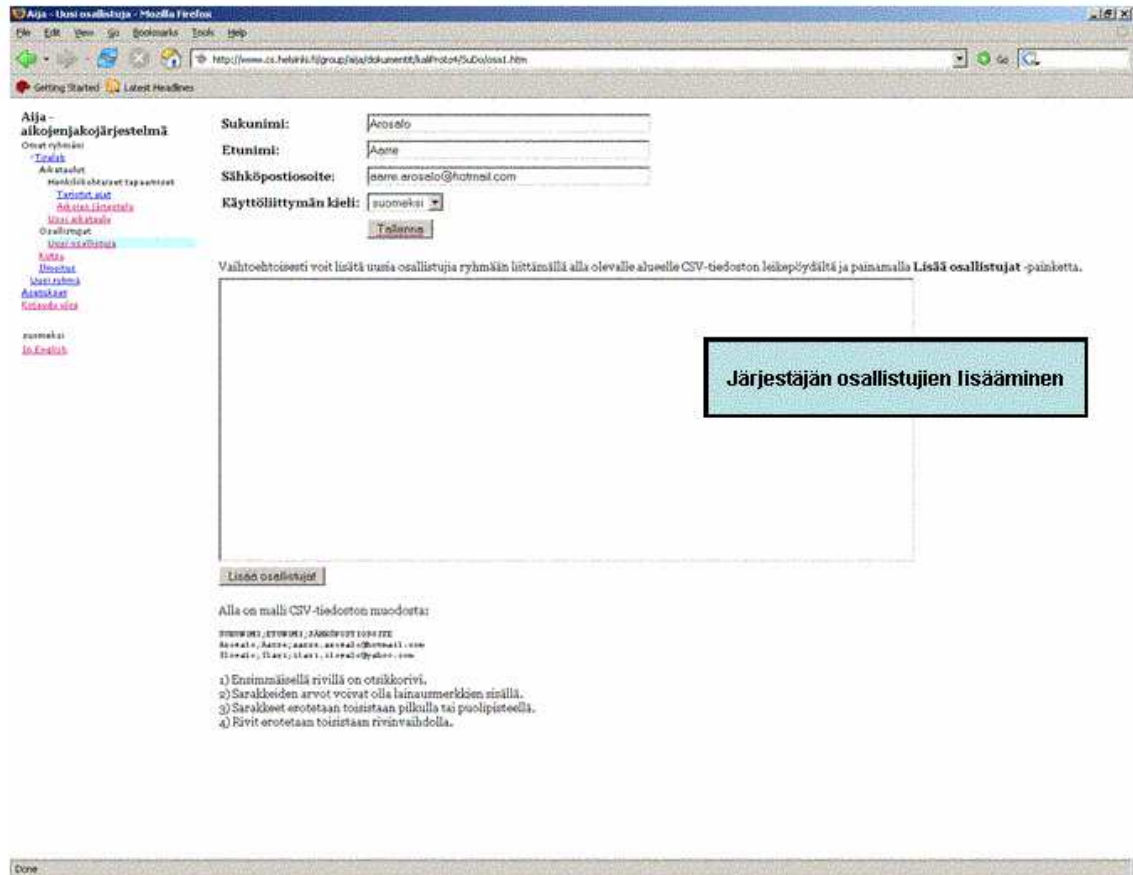
- Nimi:** Henkilökohtaiset tapaamiset
- Tyyppi:** Henkilötapaaminen
- Jako:** 20 min

 Below the form, there is a section for "o tarjottua aikaa, o osallistujaa" with a grid of days and times. The grid is organized by month and day. The days shown are:

- kuu (June): 1-5
- 07 (July): 6-12
- 08 (August): 13-19
- 09 (September): 20-26
- lok (October): 27-31
- 11 (November): 1-7
- 12 (December): 8-14
- 01 (January): 15-21
- 02 (February): 22-28
- 03 (March): 29-31
- 04 (April): 1-7
- 05 (May): 8-14
- 06 (June): 15-21
- 07 (July): 22-28
- 08 (August): 29-31
- 09 (September): 1-7
- 10 (October): 8-14
- 11 (November): 15-21
- 12 (December): 22-28
- 01 (January): 29-31

On the right side of the grid, there is a vertical column of time slots ranging from 8.00 to 19.40 in 20-minute increments. A text box with a black border and white background is overlaid on the grid, containing the text "Järjestäjän uuden aikataulun luominen".

Kuva 9: Aikataulun luonti.



Kuva 10: Osallistujien lisääminen.

Aija - Sopivuusjärjestelmä - Mozilla Firefox

http://www.cs.helsinki.fi/group/aia/dokumentit/AijaPohotus/5/0/omata.htm

Getting Started | Latest Headlines

Aija - aikojenjakojärjestelmä
Oven ryhmään
+ Tilat
Oven osallistujien
- Omu

[Hakijoiden arvioinnit \(Aija\)](#)
[Ehdotusajastus \(Aija\)](#)
[Asennus](#)
[Käyttöohje](#)

Nämä ovat tarjottuja aikoja. Valitse aikaa sopivuuksien mukaan.

o tarjottuja aikoja, o osallistujia

ma ti ke to pe la su

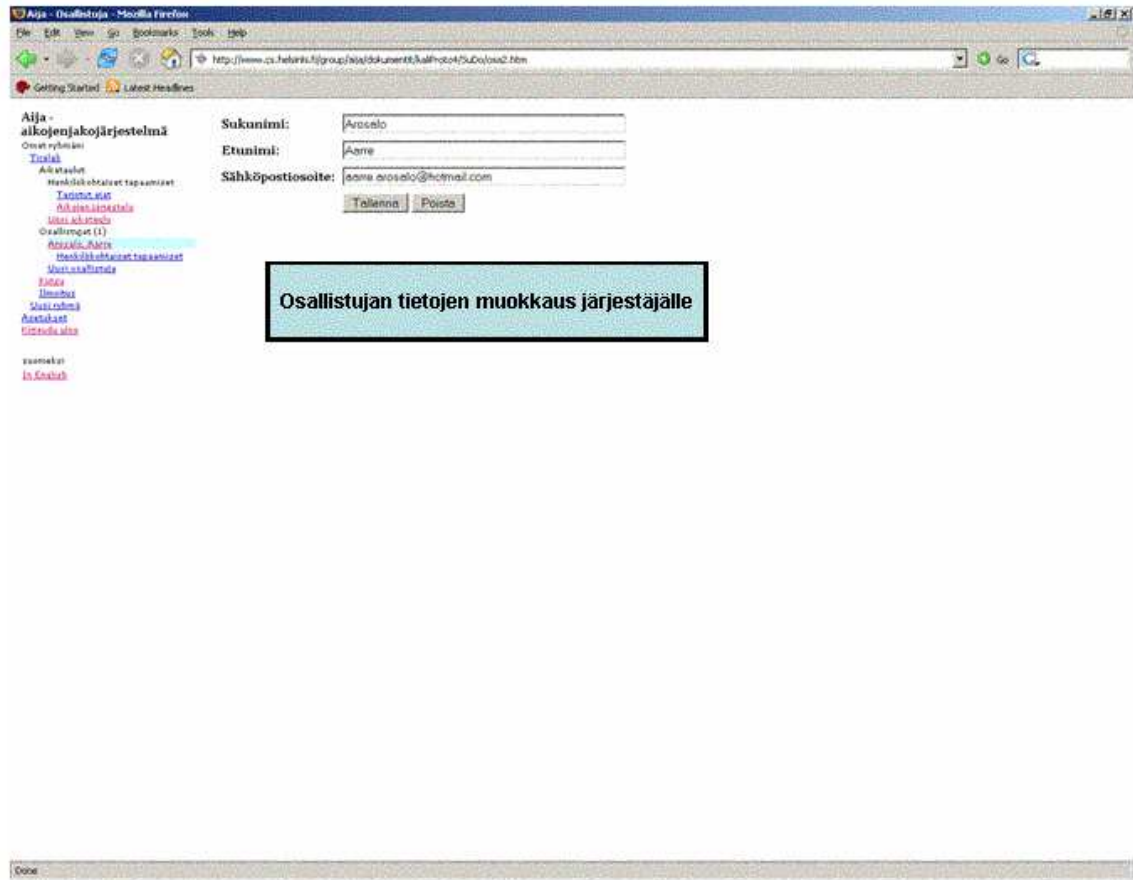
05 6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 1 2 3
05 4 5 6 7 8 9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31
05 1 2 3 4 5 6 7
08 8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
05 29 30 31 1 2 3 4
08 5 6 7 8 9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
05 26 27 28 29 30 1 2
08 3 4 5 6 7 8 9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30

	ma	ti	ke	to	pe	la	su
9.00				ei tiedossa			
9.20		ei tiedossa		ei tiedossa			
9.40		ei tiedossa					
10.00	ei tiedossa						
10.20	ei tiedossa						
10.40	ei tiedossa						
11.00	ei tiedossa						
11.20	ei tiedossa				ei tiedossa		
11.40	ei tiedossa				ei tiedossa		
12.00					ei tiedossa		
12.20					ei tiedossa		
12.40							
13.00							
13.20							
13.40							
14.00	ei tiedossa						
14.20	ei tiedossa						
14.40	ei tiedossa						
15.00	ei tiedossa						
15.20	ei tiedossa						
15.40	ei tiedossa						

Code

Osallistujan sopivuuksien valitseminen

Kuva 11: Sopivuudet.

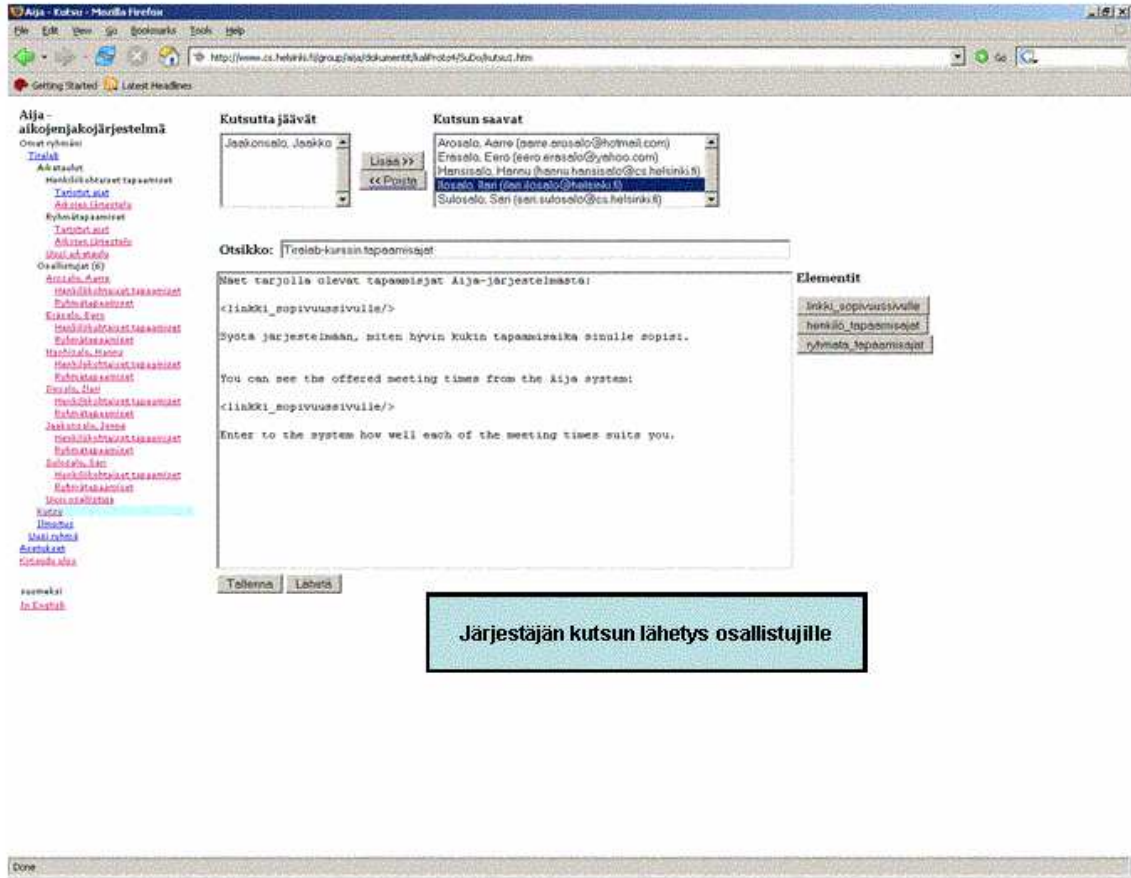


Kuva 12: Osallistujan tietojen muokkaus.

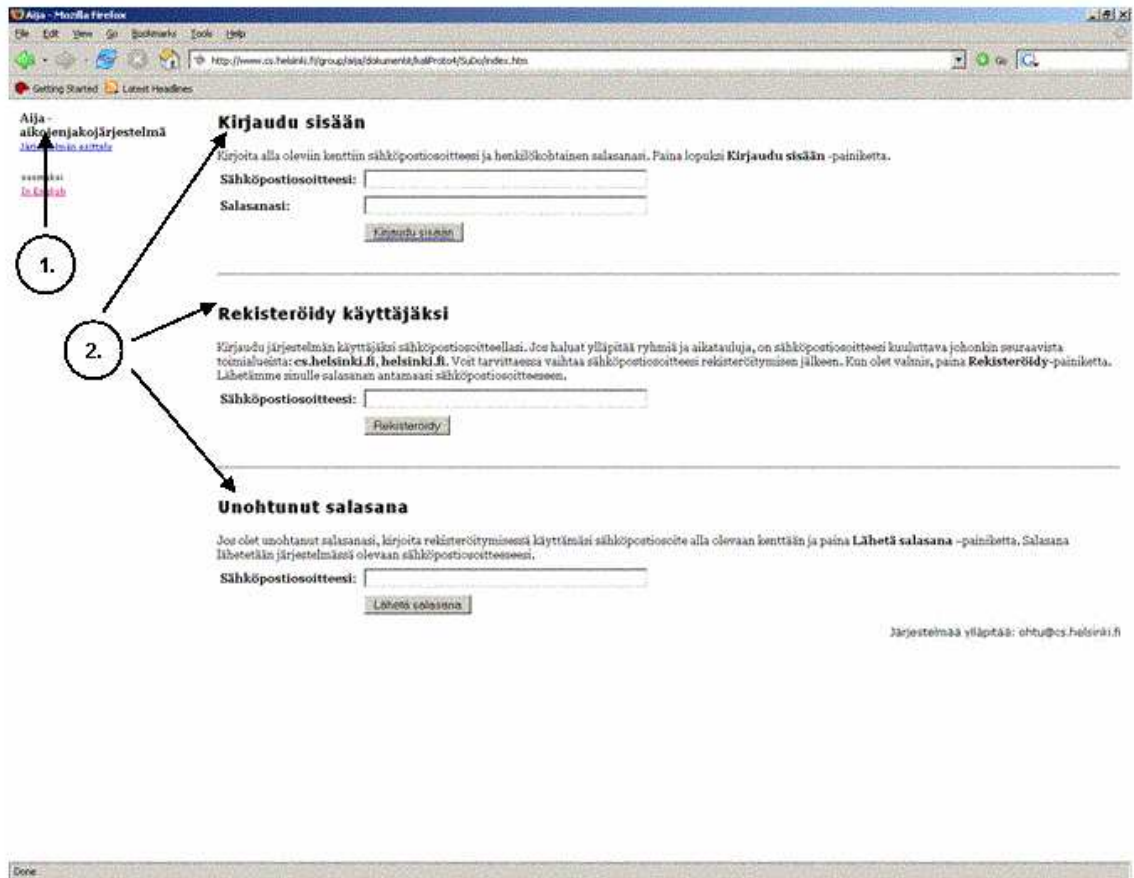
The screenshot shows a web browser window with a calendar application. A yellow callout box highlights a notification: "Ei voi muuttaa tärjöttien aikojen sopivuuksia, sillä tapaamiskoja ollaan parhaita varamassa tai ne on jo varattu". A blue callout box points to a specific time slot (11:20) with the text "Esimerkki järjestelmän ilmoitustiestä käyttäjälle". The calendar grid shows various time slots with status indicators like "erittämättä", "hyvä", "ei tiedossa", and "välttämät".

Time	Ma	Ti
9:00		
9:20		erittämättä
9:40		hyvä
10:00		erittämättä
10:20		erittämättä
10:40		erittämättä
11:00		erittämättä
11:20	hyvä	erittämättä
11:40	välttämät	erittämättä
12:00		välttämät
12:20		välttämät
12:40		erittämättä
13:00		erittämättä
13:20		erittämättä
13:40		erittämättä
14:00	ei tiedossa	ei tiedossa
14:20	ei tiedossa	ei tiedossa
14:40	ei tiedossa	ei tiedossa
15:00	ei tiedossa	ei tiedossa
15:20	ei tiedossa	ei tiedossa
15:40	ei tiedossa	ei tiedossa

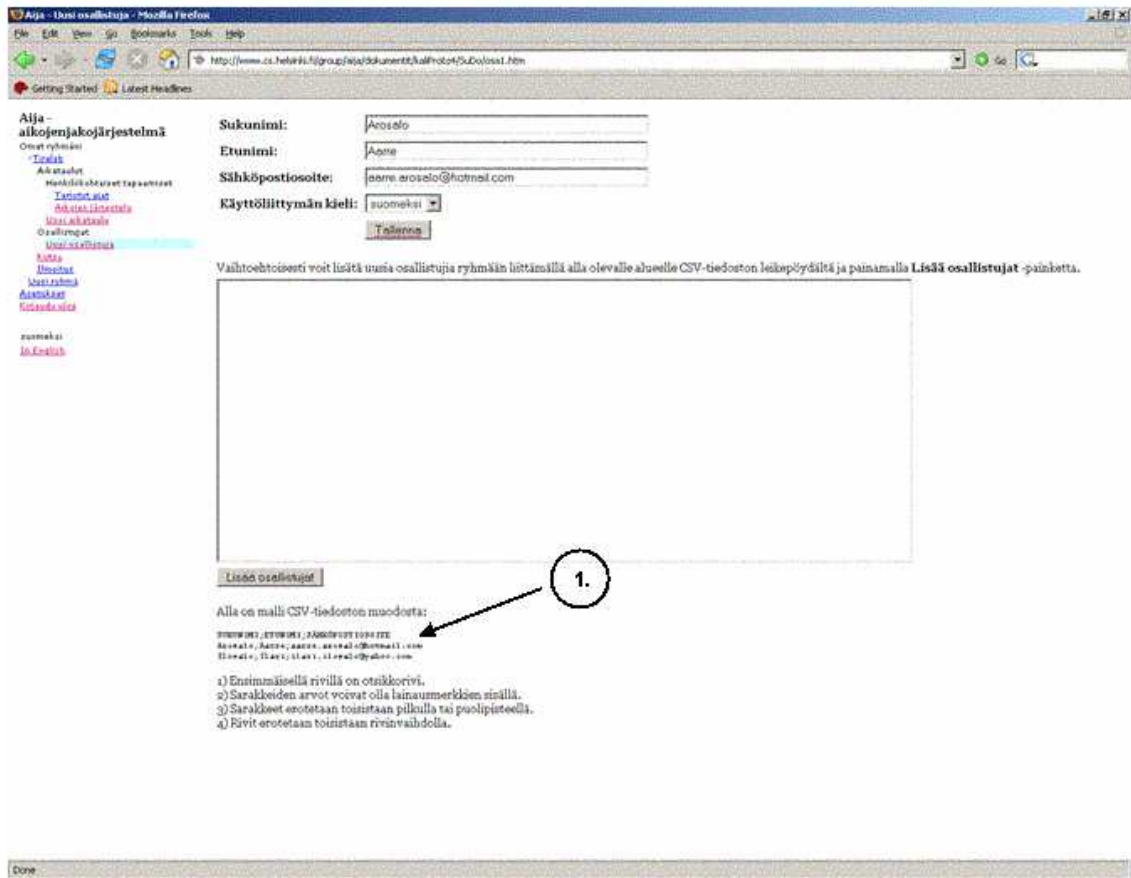
Kuva 14: Käyttäjailmoitus.



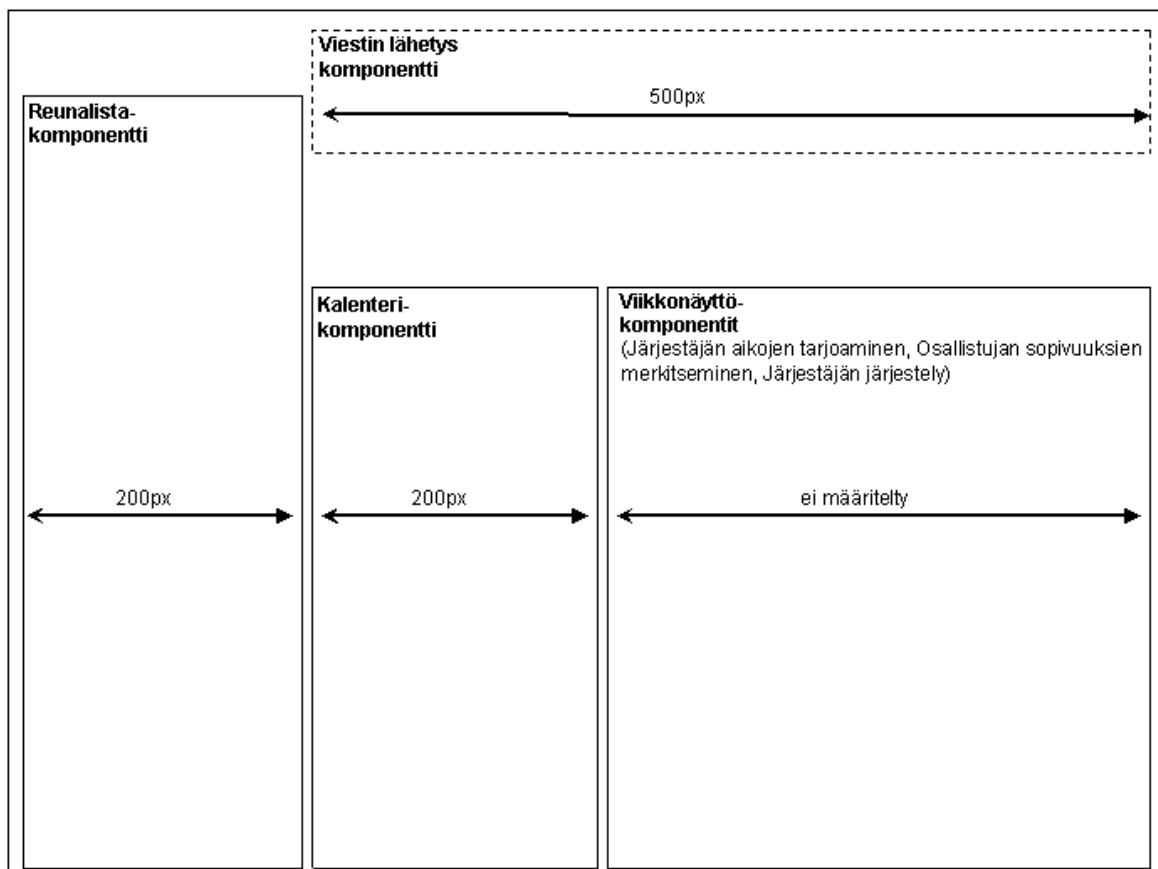
Kuva 15: Kutsun lähetytys.



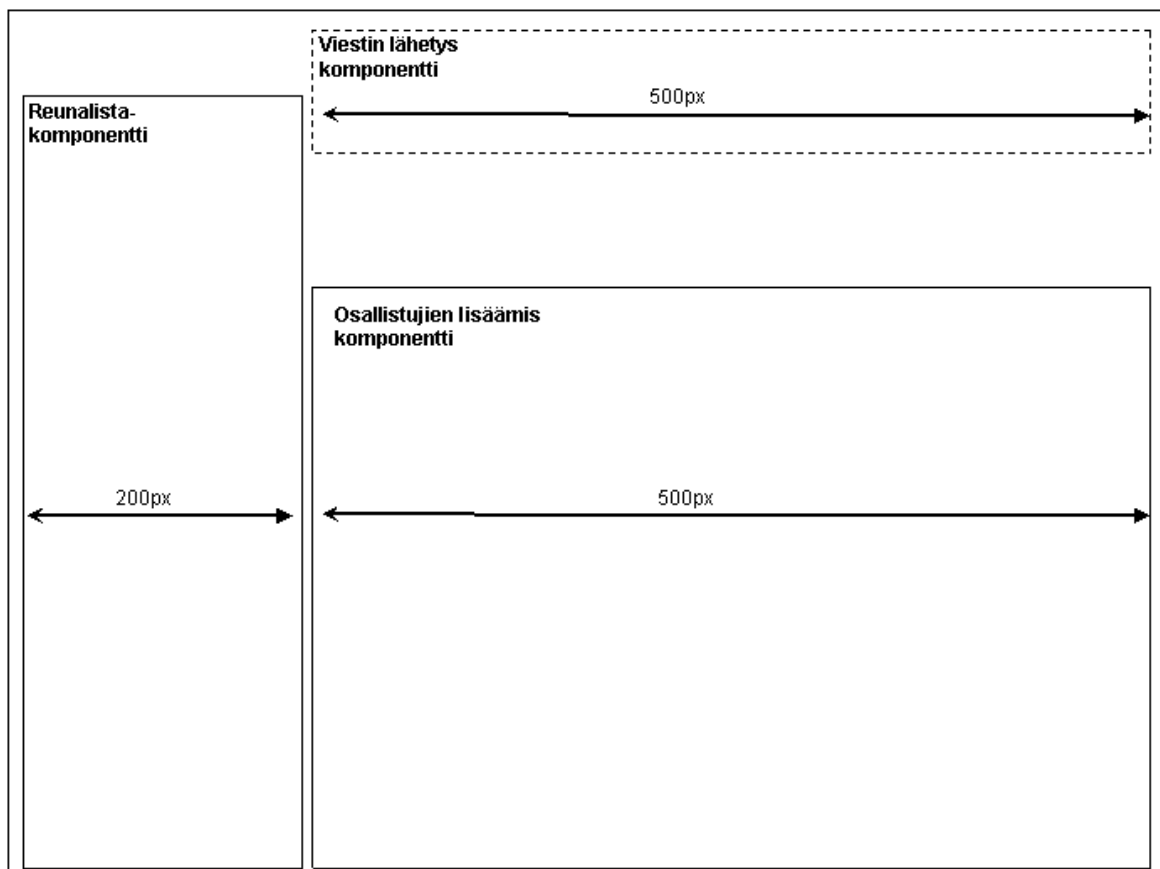
Kuva 16: Elementit.



Kuva 17: Elementit.



Kuva 18: Asemointi.



Kuva 19: Asemointi.