

# **Suunnitteludokumentti**

NJC2

Helsinki 26.3.2004

Ohjelmistotuotantoprojekti

HELSINGIN YLIOPISTO  
Tietojenkäsittelytieteen laitos

**Kurssi**

581260 Ohjelmistotuotantoprojekti (6 ov)

**Projektiryhmä**

Eero Anttila  
Olli Jokinen  
Jesse Liukkonen  
Jani Markkanen  
Jere Salonen  
Jouni Tuominen

**Asiakas**

Olli Lahti

**Johtoryhmä**

Juha Taina

**Kotisivu**

<http://www.cs.helsinki.fi/group/njc2/>

**Versiohistoria**

Versio	Päiväys	Tehdyt muutokset
1.0	10.3.2004	Ensimmäinen versio
1.1	15.3.2004	Tietokantakyselyt luokkakuvausten yhteyteen
1.2	18.3.2004	Tietokantakaavion päivitys, luokkakuvausten päivitys
1.3	22.3.2004	Sekvenssikaavioiden lisäys, luokkakaavion ja -kuvauksen päivitys
1.3.1	24.3.2004	Sekvenssikaavioiden selityksiä, luokka- ja tietokantakaavion päivitys
1.3.2	25.3.2004	Dokumentin kirjoitusvirheiden hionta

# Sisältö

<b>1</b>	<b>Johdanto</b>	<b>1</b>
1.1	Dokumentin tarkoitus . . . . .	1
1.2	Tuote ja ympäristö . . . . .	1
1.3	Määritelmät, merkintätavat ja lyhenteet . . . . .	1
1.4	Yleiskatsaus dokumenttiin . . . . .	2
<b>2</b>	<b>Järjestelmän yleiskuvaus</b>	<b>2</b>
2.1	Sovellusalueen kuvaus . . . . .	2
2.2	Järjestelmän liittyminen ympäristöönsä . . . . .	3
2.3	Laiteympäristö . . . . .	3
2.4	Ohjelmistoympäristö . . . . .	3
2.5	Toteutuksen keskeiset reunaehdot . . . . .	3
2.6	Sopimukset ja standardit . . . . .	4
<b>3</b>	<b>Arkkitehtuurin kuvaus</b>	<b>4</b>
3.1	Arkkitehtuurin yleiskuvaus . . . . .	4
3.1.1	Suunnittelumallit . . . . .	5
3.2	Ohjelmiston toiminta esimerkkikäyttötapauksen avulla . . . . .	5
3.2.1	Rekisteröityminen . . . . .	5
3.2.2	Sisäänkirjautuminen . . . . .	6
3.2.3	Profilin muokkaus . . . . .	6
3.2.4	Artikkelin lähettäminen . . . . .	7
3.2.5	Asiantuntijan valinta artikkeliin . . . . .	7
3.2.6	Lausuntopyyntöön vastaaminen . . . . .	8
3.2.7	Lausunnon lähettäminen . . . . .	9
3.2.8	Artikkelin poistaminen . . . . .	9
3.2.9	Käyttäjähallinta . . . . .	9
3.2.10	Artikkelin valinta lehteen . . . . .	9
3.2.11	Viestin lähettäminen päätoimittajalle/toimittajalle . . . . .	10
3.2.12	Päätöksen antaminen artikkelille . . . . .	10
3.2.13	Uloskirjautuminen . . . . .	11
<b>4</b>	<b>Virheen käsittely</b>	<b>11</b>

4.1	Käyttäjän virheiden käsittely . . . . .	11
4.2	Ohjelmistovirheiden käsittely . . . . .	12
4.3	Laitteistovirheiden käsittely . . . . .	12
<b>5</b>	<b>Luokat ja metodit</b>	<b>13</b>
5.1	ControllorServlet . . . . .	13
5.2	DAO . . . . .	15
5.3	SessionData . . . . .	21
5.4	Event . . . . .	22
5.5	HtmlTyokalu . . . . .	22
5.6	TiedostoServlet . . . . .	22
5.7	Tiedosto . . . . .	23
5.8	Postittaja . . . . .	23
5.9	Salaaja . . . . .	23
5.10	DBConnector . . . . .	23
5.11	Tulos . . . . .	24
5.12	Sahkoposti . . . . .	25
5.13	Paivamaara . . . . .	25
5.14	Lausunto . . . . .	26
5.15	Artikkeli . . . . .	27
5.16	Kayttaja . . . . .	28
5.17	Ala . . . . .	29
<b>6</b>	<b>JSP-sivut</b>	<b>30</b>
6.1	Kaikkien käyttäjien sivut . . . . .	30
6.1.1	header.jsp . . . . .	30
6.1.2	footer.jsp . . . . .	30
6.1.3	kehys.jsp . . . . .	30
6.1.4	menu.jsp . . . . .	30
6.1.5	kirjautuminen.jsp . . . . .	31
6.1.6	profilinmuokkaus.jsp . . . . .	31
6.1.7	uloskirjautuminen.jsp . . . . .	31
6.2	Kirjoittajan sivut . . . . .	31
6.2.1	rekisteroityminen.jsp . . . . .	32

6.2.2	omatartikkelit.jsp . . . . .	32
6.3	Asiantuntijoiden sivut . . . . .	32
6.3.1	artikkelienarvostelu.jsp . . . . .	33
6.4	Toimituksen sivut . . . . .	33
6.4.1	artikkelienhallinta.jsp . . . . .	33
6.4.2	lisaaasiantuntija.jsp . . . . .	34
6.4.3	lehdentila.jsp . . . . .	34
6.4.4	kayttajahallinta.jsp . . . . .	34
6.4.5	raportit.jsp . . . . .	34
6.4.6	sahkoposti.jsp . . . . .	35
6.4.7	paatoimittajanakyma.jsp . . . . .	35
<b>7</b>	<b>Tietokantakuvaus</b>	<b>35</b>
7.1	Tietokannan rakenne . . . . .	35
7.2	Tietokantakyselyt ja transaktioiden sitoutuminen . . . . .	36
7.3	Käyttäjä . . . . .	37
7.4	Asiantuntija . . . . .	38
7.5	Osaamisala . . . . .	38
7.6	Ala . . . . .	38
7.7	Sähköposti . . . . .	39
7.8	Lehti . . . . .	39
7.9	Artikkeli . . . . .	39
7.10	Versio . . . . .	40
7.11	Lausuntopyyntö . . . . .	40
7.12	Lausunto . . . . .	41
7.13	Liite . . . . .	41
7.14	Päätös . . . . .	42
7.15	Loki . . . . .	42
7.16	Järjestelmään liittyvien tiedostojen tallennus . . . . .	42
<b>8</b>	<b>Käyttöliittymä</b>	<b>43</b>
8.1	Kaikki käyttäjäryhmät . . . . .	43
8.1.1	Kirjautuminen . . . . .	43
8.2	Kirjoittaja . . . . .	44

8.2.1	Rekisteröityminen . . . . .	44
8.2.2	Omat artikkelit . . . . .	44
8.2.3	Profilin muokkaus . . . . .	45
8.3	Asiantuntija . . . . .	46
8.3.1	Artikkelien arvostelu . . . . .	46
8.3.2	Profilin muokkaus . . . . .	46
8.4	Toimitus . . . . .	47
8.4.1	Artikkelien hallinta . . . . .	47
8.4.2	Lehden tila . . . . .	48
8.4.3	Käyttäjähallinta . . . . .	49
8.4.4	Raportit . . . . .	50
8.4.5	Profilin muokkaus . . . . .	50
8.4.6	Päätoimittajanäkymä . . . . .	51
<b>9</b>	<b>Järjestelmästä saatavat raportit</b>	<b>51</b>
<b>10</b>	<b>Erityiset tekniset ratkaisut</b>	<b>52</b>
10.1	Salasanan tallennus . . . . .	52
10.2	Sähköpostin lähetys . . . . .	52
10.3	Tiedostojen käsittely . . . . .	52
10.4	JSP-sivujen header . . . . .	53

## **Liitteet**

**1 Event-luokan vakiot**

**2 Käyttöliittymähahmotelmat**

# 1 Johdanto

Tämä dokumentti on osa tietojenkäsittelytieteen laitoksen NJC2-ryhmän ohjelmistotuotantoprojektia. Tarkoituksena on Nordic Journal of Computing -lehden toimeksiantamana tuottaa väline, jolla tehostetaan kyseisen lehden julkaisuprojektia.

Valmis tuote tulee Nordic Journal of Computing -lehden toimituksen käyttöön. Tuotteen oikeudet on luovutettu Helsingin yliopistolle, joka julkaisee tuotteen GNU General Public Licensen tai GNU Lesser General Public Licensen alaisuudessa.

## 1.1 Dokumentin tarkoitus

Tämän suunnitteludokumentin tarkoituksena on tarkentaa määrittelydokumentin tiedot suunnitteluvaiheen vaatimaan muotoon. Valmiista dokumentista on käytävä ilmi, miten ohjelmiston määrittely ja vaatimukset tullaan toteuttamaan. Suunnitelmien on oltava niin tarkkoja, että toteutus onnistuu suoraviivaisesti niiden perusteella.

## 1.2 Tuote ja ympäristö

Projektin tarkoituksena on toteuttaa Nordic Journal of Computing -lehden toimitukselle väline, jolla nopeutetaan ja helpotetaan lehden julkaisuprosessia. Tuotteen on määrä yksinkertaistaa NJC -lehden julkaisuprosessia erityisesti kirjanpidon ja kommunikoinnin osalta.

NJC-lehden toimituksen lisäksi tuote yksinkertaistaa myös lehteen kirjoittavien ja kirjoitusten tarkastajina toimivien asiantuntijoiden työtä. Molemmille sidosryhmille tarjotaan HTML-lomakkeisiin perustuvat käyttöliittymät, joiden kautta he kommunikoivat toimituksen kanssa. Tuotteen käyttämiseen vaaditaan kehyksiä (frameset) tukeva WWW-selain ja Internet-yhteys.

## 1.3 Määritelmät, merkintätavat ja lyhenteet

Seuraavassa taulukossa mainitaan dokumentissa käytettävät vieraat lyhenteet, määritelmät ja merkintätavat.



Termi/lyhenne	Merkitys
CSS	Cascading Style Sheets, tyylisivut HTML-kielen ulkoasun määrittelyyn
DAO	Data Access Object, suunnittelumalli (design pattern) alhaisen tason tiedon saannin logiikan erottamiseksi korkean tason logiikasta ja tässä ohjelmistossa käytettävän luokan nimi
GNU-lisenssi	General Public License, lisenssin alaiset ohjelmat ovat vapaasti levitettävissä ja muunneltavissa
JSP	JavaServer Pages, teknologia dynaamisten verkkosivujen tekoon
NJC	Nordic Journal of Computing -lehti, projektin asiakas
NJC2	Ryhmän nimi
PostgreSQL	Tietokantaohjelmisto joka tukee SQL-toimintoja
Servlet	Java-kielinen palvelimella suoritettava ohjelma
Tomcat	Servlet-alusta, servlettien hallitsemiseen

## 1.4 Yleiskatsaus dokumenttiin

Seuraavassa luvussa esitellään toteutettavan järjestelmän yleiskuvaus, lähtien projektin tarkoituksesta ja päätyen tarvittaviin ohjelmistoihin sekä asiakkaan kanssa sovittuihin toteutuksen reunaehtoihin. Luvussa 3 keskitytään ohjelmiston arkkitehtuurin yleiskuvaukseen siten, että lukija saisi käsityksen koko ohjelmiston rakenteesta. Rakenteen esittelyn jälkeen luvussa 4 kerrotaan yleisiä virheenkäsittelysääntöjä ja luvussa 5 kerrotaan yksityiskohtaisesti luokkien ja metodien tarkoitukset sekä niiden kommunikointirajapinnat.

Luvussa 6 esitellään lyhyesti järjestelmään sisältyvät JavaServer Pages (JSP) -sivut. Tämän jälkeen luvussa 7 esitellään ohjelmiston tietokantarakenne ensin yleiskuvauksena ja sitten yksityiskohtaisesti tietokantataulu kerrallaan. Luku 8 keskittyy käyttöliittymiin. Luvussa 9 käydään läpi järjestelmästä saatavia raportteja, ja luvussa 10 keskitytään erityisiin teknisiin ratkaisuihin.

## 2 Järjestelmän yleiskuvaus

Luvussa esitellään toteutettavan järjestelmän yleiskuvaus, johdatus asiakkaan ympäristöön ja sovellusalueeseen.

### 2.1 Sovellusalueen kuvaus

Projektin on tarkoitus toteuttaa Nordic Journal of Computing -lehden toimitukselle väline, jolla nopeutetaan ja helpotetaan lehden julkaisuprosessia. Tuotettavan ohjelmiston on tarkoitus automatisoida artikkelin tarkastuksen vaiheita ja helpottaa toimituksen kommunikoinnista sidosryhmien kanssa.

Projekti kuuluu Helsingin yliopiston tietojenkäsittelytieteen laitoksen Ohjelmistotuotantoprojekti- kurssiin. Järjestelmä tulee Nordic Journal of Computing -

lehden toimituksen käyttöön.

## 2.2 Järjestelmän liittyminen ympäristöönsä

Järjestelmä toimii itsenäisesti olematta minkään muun järjestelmän osana. Järjestelmä käyttää apunaan lähtevän sähköpostin palvelinta.

## 2.3 Laiteympäristö

Ohjelmistoa suoritetaan tietojenkäsittelytieteen laitoksen alkokrunni-palvelimessa, johon on asennettu tarpeelliset palvelin- ym. ohjelmistot. JSP- ja Servlet-sivuja suoritetaan Tomcat-ohjelmiston avulla, joka on myös asennettu valmiiksi alkokrunni-palvelimelle. Tietokantana käytetään PostgreSQL:ää.

## 2.4 Ohjelmistoympäristö

	Ohjelma	Versio
Käyttöjärjestelmä	Linux	CSL 2.113
Kääntäjä	Java	j2sdk 1.4.2
JSP ja JavaServlet ajoympäristö	Tomcat	4.1.18
WWW-palvelin	Apache	
Tietokantaohjelmisto	PostgreSQL	7.4
Lähtevän sähköpostin palvelin	Sendmail	
WWW-selain	Internet Explorer	5.0
	Mozilla	1.6
	Netscape	6
	Opera	6

## 2.5 Toteutuksen keskeiset reunaehdot

Tärkeimpiä vaatimuksia ovat:

- Sovelluksen on oltava tietoturvallinen
- Käyttöliittymä tehdään englannin kielellä
- Projektin oikeudet ovat Helsingin yliopistolla
- Ohjelmisto julkaistaan GNU-lisenssin alaisuudessa

## 2.6 Sopimukset ja standardit

### Koodausperiaatteet

- Koodi kirjoitetaan suomen kielellä
- Luokkien nimet kirjoitetaan isolla alkukirjaimella
- Metodit ja muuttujat kirjoitetaan pienelle alkukirjaimella
- Muuttujat nimetään samannimisiksi kuin niitä vastaavat atribuutit relaatiossa
- Sanat erotetaan isolla kirjaimella
- Final-tyyppiset muuttujat kirjoitetaan kokonaan isolla ja sanat erotetaan alaviivalla
- Kommentoinnit tehdään javadoc-tyylin mukaisesti ja suomen kielellä

### Käytettävät pakkaukset

Kuvaus	Nimi	pakkaus
Tiedostojen Upload-toiminto	Apache FileUpload 1.0	commons-fileupload-1.0.jar
Yhteys PostgreSQL-kantaan	JDBC	jdbc7.0-1.2.jar
Testaustyökalu	JUnit 3.8.1	junit.jar
Sähköpostin lähetys	JavaMail 1.3.1	mail.jar

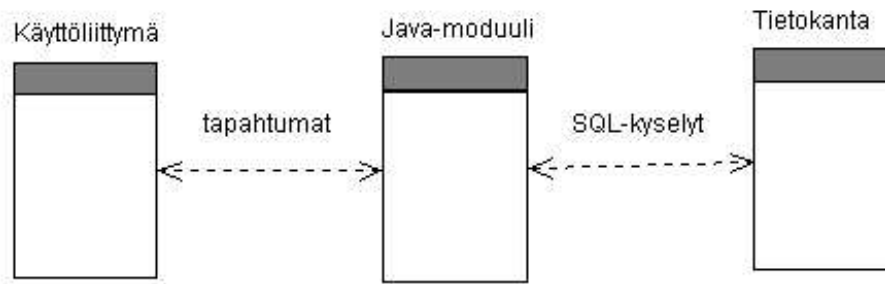
## 3 Arkkitehtuurin kuvaus

### 3.1 Arkkitehtuurin yleiskuvaus

Järjestelmä jaetaan kolmeen erilliseen osajärjestelmään: web-pohjaiseen käyttöliittymään, tietokannan ja käyttöliittymän välillä toimivaan Java-moduuliin sekä itse tietokantaan.

Käyttäjille näkyvä osa on selaimen kautta käytettävä käyttöliittymä, jonka avulla käyttäjät lähettävät artikkeliaan toimitukselle, joka puolestaan lähettää ne eteenpäin asiantuntijoille. Sekä kirjoittajat, asiantuntijat että toimittajat käyttävät järjestelmää web-käyttöliittymän kautta. Osa käyttöliittymästä on kaikille käyttäjäryhmille yhteistä, mutta pääasiassa jokainen ryhmä käyttää järjestelmää oman käyttöliittymänsä kautta.

Java-moduuliin sijoitettu servlet kuuntelee käyttöliittymäsivuilta lähetettäviä tapahtumia (event), käsittelee ne ja hakee kyselyiden avulla tietokannasta tarvitsemansa tiedot. Tällaisia tapahtumia ovat esimerkiksi käyttäjän sisäänkirjautuminen ja rekisteröityminen sekä uuden artikkelin lähettäminen. Tietokantakyselyiden tulosten perusteella luodaan käyttäjille päivittynyt käyttöliittymäsivu. Järjestelmän Java-osio siis käsittelee käyttöliittymän ja tietokannan välillä kulkevaa tietoa.



Kuva 1: Arkkitehtuurin yleiskuvaus

### 3.1.1 Suunnittelumallit

Järjestelmän arkkitehtuuri toteutetaan **ModelViewController (MVC)**<sup>1</sup> -suunnittelumallilla. Järjestelmä jaetaan kolmeen eri kerrokseen: Model, View ja Controller.

- **Model** huolehtii tietokantakyselyistä.
- **View** muokkaa modellin sisällön käyttökelpoiseen esitysmuotoon.
- **Controller** määrittelee järjestelmän toiminnallisuuden ja ottaa vastaan pyynnöt käyttäjältä.

Model on toteutettu käyttämällä **DAO (Data Access Object)**<sup>2</sup> suunnittelumallia. Ideana on, että yhteys tietoresurssiin toimii rajapinnan kautta yhdessä paikassa. Tämä mahdollistaa tietoresurssin mahdollisen vaihtamisen aiheuttamatta muutoksia muualle kuin DAO-luokkaan (esim. XML-muotoinen tieto tai jokin toinen tietokantatyyppejä). Lisäksi kaikki kyselyt on selkeästi keskitettynä yhteen paikkaan.

## 3.2 Ohjelmiston toiminta esimerkikäyttötapausten avulla

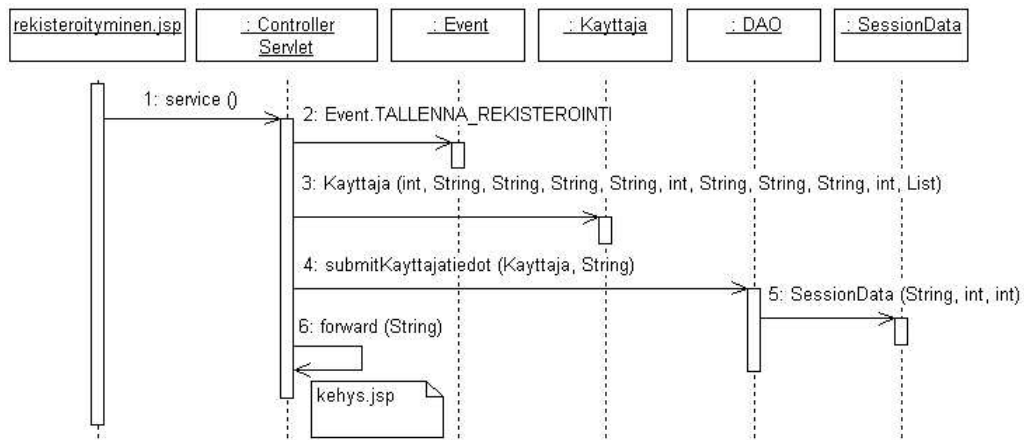
Tässä aliluvussa kuvataan luokkien välistä kommunikointia sekvenssikaavioiden avulla.

### 3.2.1 Rekisteröityminen

Kuvassa 2 kuvataan käyttäjän rekisteröitymistä järjestelmän kirjoittajaksi. Uuden käyttäjän tiedot annetaan Kayttaja-oliona salasanan kanssa parametriksi DAO:n metodille.

<sup>1</sup>[http://java.sun.com/blueprints/guidelines/designing\\_enterprise\\_applications\\_2e/app-arch/app-arch2.html#1105854](http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2e/app-arch/app-arch2.html#1105854)

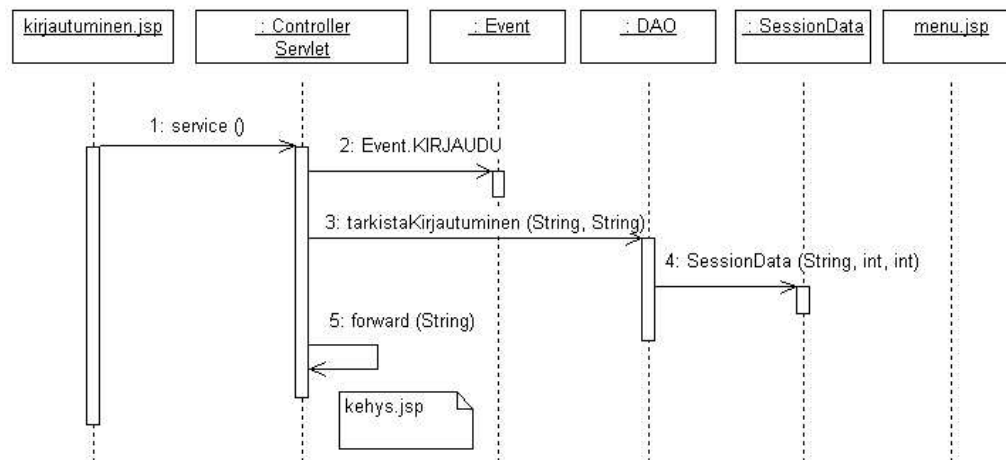
<sup>2</sup><http://java.sun.com/blueprints/corej2eepatterns/Patterns/DataAccessObject.html>



Kuva 2: Sekvenssikaavio, rekisteröityminen.

### 3.2.2 Sisäänkirjautuminen

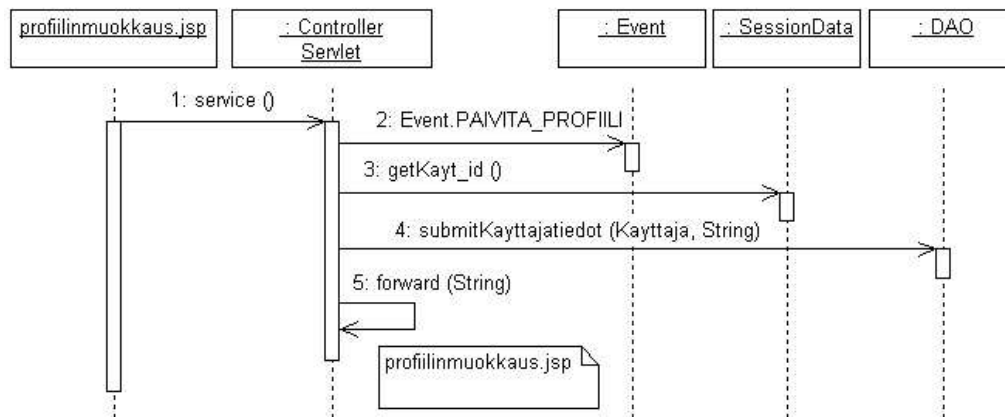
Sisäänkirjautuminen (kuva 3) tapahtuu kirjautuminen.jsp-sivulta, joka lähettää käyttäjän syöttämät tiedot ControllerServlet:ille. ControllerServlet kutsuu DAO:a, joka tarkistaa, että salasana ja käyttäjätunnus täsmäävät. Jos ne täsmäsivät, DAO luo uuden SessionData-olion, ja ControllerServlet lähettää tiedon sisäänkirjautumisen onnistumisesta menu.jsp:lle.



Kuva 3: Sekvenssikaavio, sisäänkirjautuminen.

### 3.2.3 Profilin muokkaus

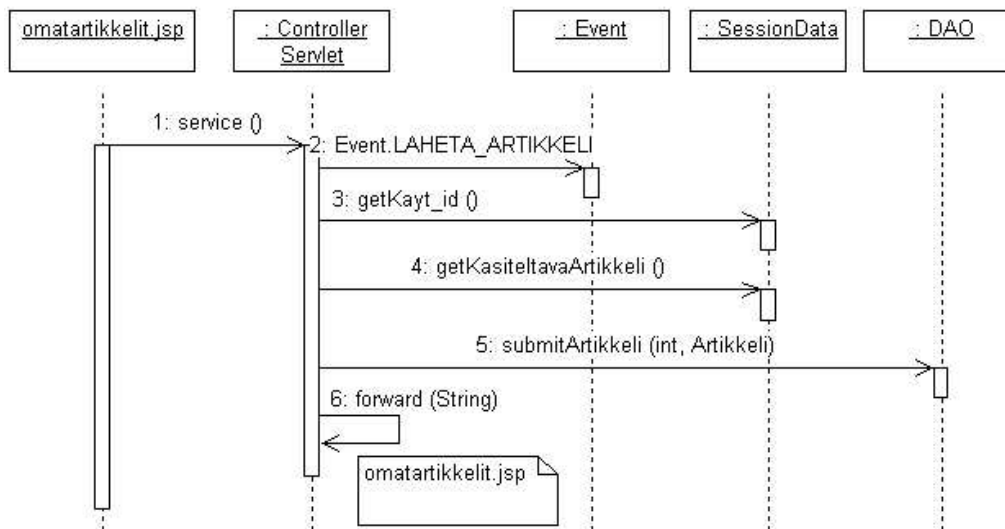
Kuvassa 4 kuvataan käyttäjän profilin päivitysoperaatio. ControllerServlet kutsuu DAO:a, joka päivittää tietokantaan käyttäjän uudet tiedot. Lopuksi ControllerServlet välittää tiedon päivityksen onnistumisesta takaisin käyttäjän nähtäväksi.



Kuva 4: Sekvenssikaavio, profiilin muokkaus.

### 3.2.4 Artikkelin lähettäminen

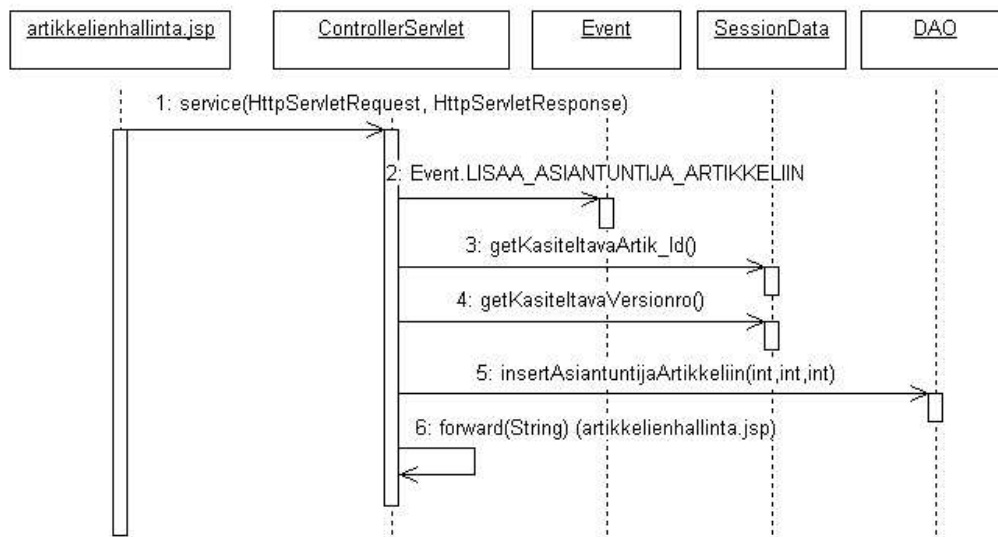
Kuvassa 5 kuvataan kirjoittajan artikkelin lähettämistä järjestelmään. Uuden artikkelin tiedot annetaan Artikkelio-oliona käyttäjätunnuksen kanssa parametriksi DAO:n metodille.



Kuva 5: Sekvenssikaavio, artikkelin lähettäminen.

### 3.2.5 Asiantuntijan valinta artikkeliin

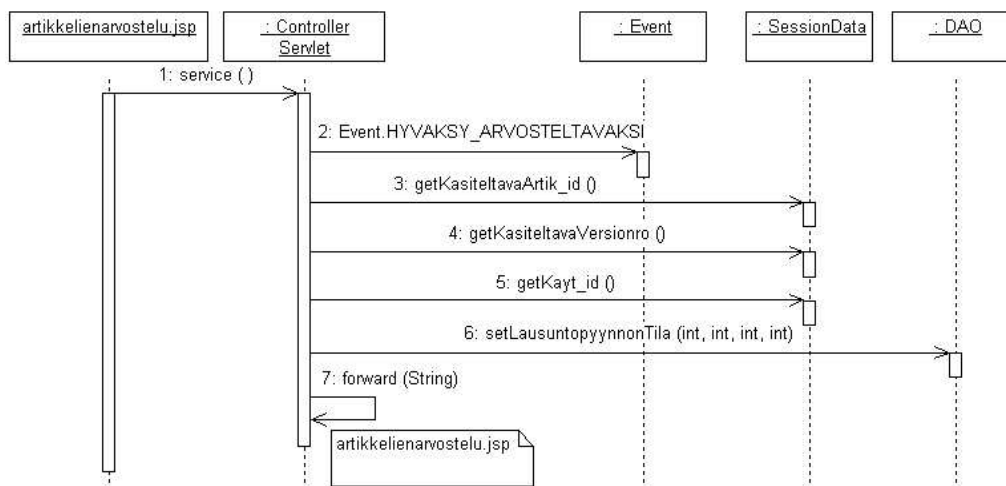
Kuvassa 6 kuvataan asiantuntijan valinta artikkeliin. Artikkelin tunnus ja versionumero haetaan SessionDatasta, jonka jälkeen ne annetaan parametreina DAO:n metodille.



Kuva 6: Sekvenssikaavio, asiantuntijan valinta artikkeliin

### 3.2.6 Lausuntopyyntöön vastaaminen

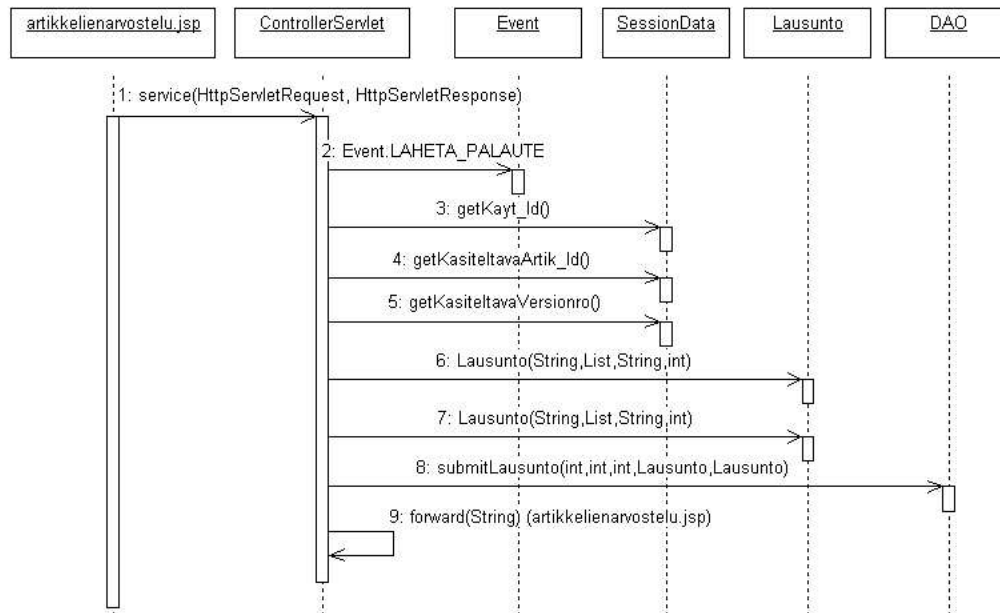
Kuvassa 7 kuvataan asiantuntijan vastaaminen (myöntävästi, mutta kielteinen vastaus tapahtuu samalla periaatteella) hänelle esitettyyn pyyntöön kirjoittaa arvio artikkelista. Artikkelin tunnus, versionumero, sekä käyttäjän tunnus haetaan SessionDatasta, jonka jälkeen ne annetaan parametreina DAO:n metodille.



Kuva 7: Sekvenssikaavio, lausuntopyyntöön vastaaminen

### 3.2.7 Lausunnon lähettäminen

Kuvassa 8 kuvataan asiantuntijan artikkelin lausunnon lähettäminen. Artikkelin tunnus, versionumero sekä käyttäjän tunnus haetaan SessionDatasta, jonka jälkeen ne annetaan kahden Lausunto-olion kanssa parametreina DAO:n metodille.



Kuva 8: Sekvenssikaavio, lausunnon lähettäminen

### 3.2.8 Artikkelin poistaminen

Kuvassa 9 kuvataan artikkelin poistamista järjestelmästä. Artikkelin tunnus haetaan SessionDatasta, jonka jälkeen tiedot annetaan parametreina DAO:n metodille.

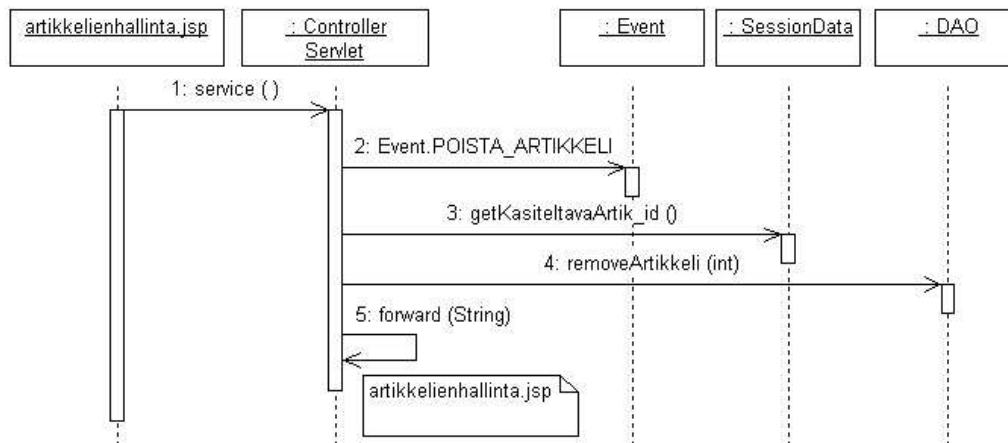
### 3.2.9 Käyttäjähallinta

Kuvassa 10 kuvataan toimituksen tekemää muutosta käyttäjän tietoihin. Alkutilanteessa valitun käyttäjän tiedot ovat näytöllä näkyvillä, ja muutokset tehtyään toimittaja on painanut "Save changes" -painiketta.

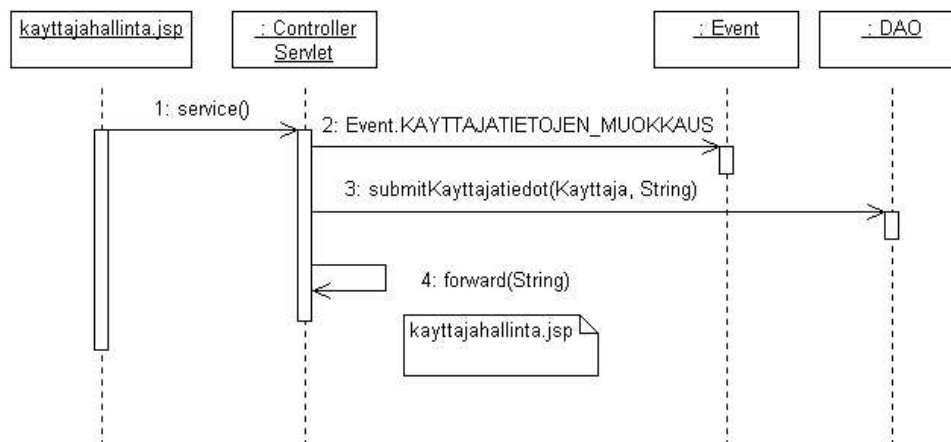
### 3.2.10 Artikkelin valinta lehteen

Kuvassa 11 kuvataan toimituksen päätöstä siirtää tarkasteltavana oleva artikkeli valittuna olevaan lehteen. Tarkasteltavan artikkelin tiedot annetaan parametriksi DAO:n metodille yhdessä lehden numeron ja vuosikerran kanssa.





Kuva 9: Sekvenssikaavio, artikkelin poistaminen



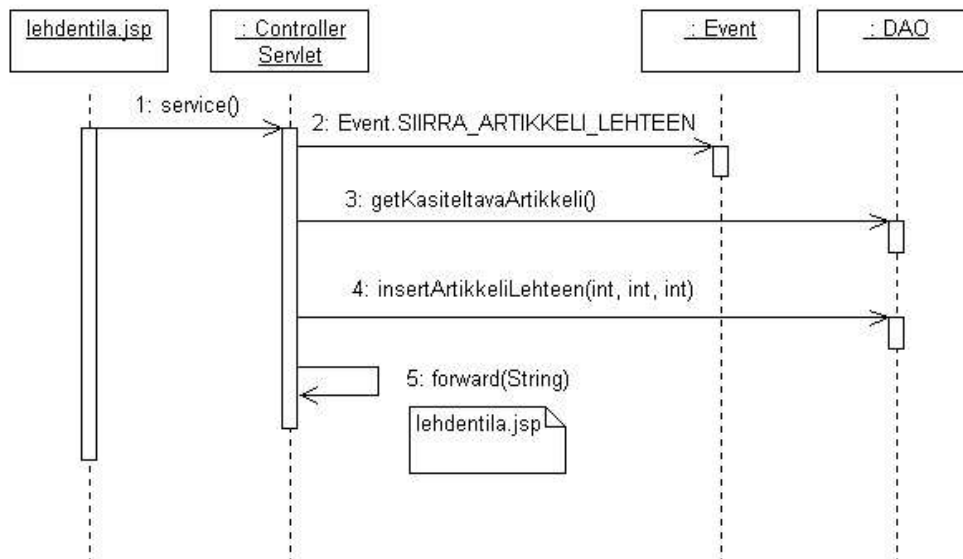
Kuva 10: Sekvenssikaavio, käyttäjähallinta

### 3.2.11 Viestin lähettäminen päätoimittajalle/toimittajalle

Kuvassa 12 kuvataan toimittajan viestin lähettämistä päätoimittajalle (tai päinvastoin). Tarkasteltavan artikkelin tiedot annetaan parametriksi DAO:n metodille yhdessä viestin ja status-tiedon (onko viesti toimittajalta vai päätoimittajalta) kanssa.

### 3.2.12 Päätöksen antaminen artikkelille

Kuvassa 13 kuvataan toimittajan päätöksen antamista artikkelille. Tarkasteltavan artikkelin tiedot ja päätös-tieto annetaan parametriksi DAO:n metodille.



Kuva 11: Sekvenssikaavio, artikkelin valinta lehteen.

### 3.2.13 Uloskirjautuminen

Kuvassa 14 käyttäjän uloskirjautumista järjestelmästä. Käyttäjän SessionData-olio tuhoaan ControllerServletin metodilla.

## 4 Virheen käsittely

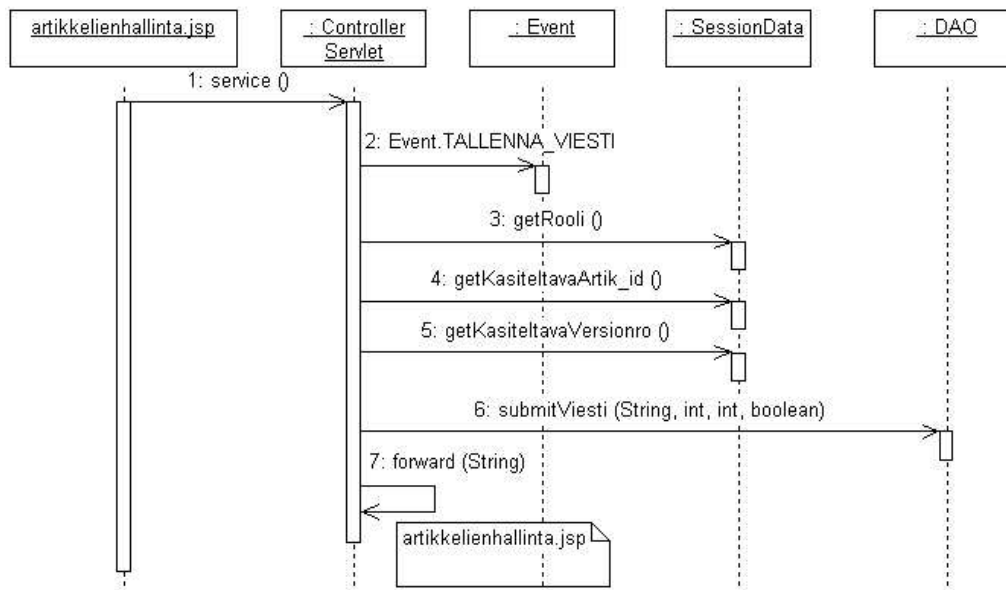
Tässä luvussa käsitellään yleisiä virheenkäsittelysääntöjä. Käyttöliittymiin liittyvät virhetilanteet löytyvät luvun 8 käyttöliittymäkuvausten yhteydestä.

### 4.1 Käyttäjän virheiden käsittely

Käyttäjän syötteistä johtuvien virheiden käsittely kattaa virheet, jotka saattavat häiritä ohjelman toimintaa. Tällaisia ovat esimerkiksi syötteen väärä muoto tai tyhjä syöte silloin, kun se ei ole sallittu.

Virheen käsittely toteutetaan Javalla, joka rajoittaa virheiden huomaamisajankohtaa. Javalla tehdyt virheet voidaan korjata vasta sitten, kun käyttäjä on lähettänyt virheet sisältävät tiedot palvelimelle. Virheen sattuessa virheen sisältävä sivu ladataan uudestaan näkyviin asiaankuuluvalla virheilmoituksella varustettuna, eikä mitään sivun sisältämiä muutoksia vielä tallenneta järjestelmään.

Virheilmoitukset tulostetaan aina mahdollisimman lähelle käyttäjän virheellistä syötettä. Virheilmoitus pyritään asettamaan siten, että se on virheen oikealla- tai alapuolella.



Kuva 12: Sekvenssikaavio, viestin lähettäminen päätoimittajalle/toimittajalle.

Virheilmoitukset merkitään CSS-komennolla (tag) `class="virhe"`, ja niille määritellään jokin yhteinen huomioväri.

## 4.2 Ohjelmistovirheiden käsittely

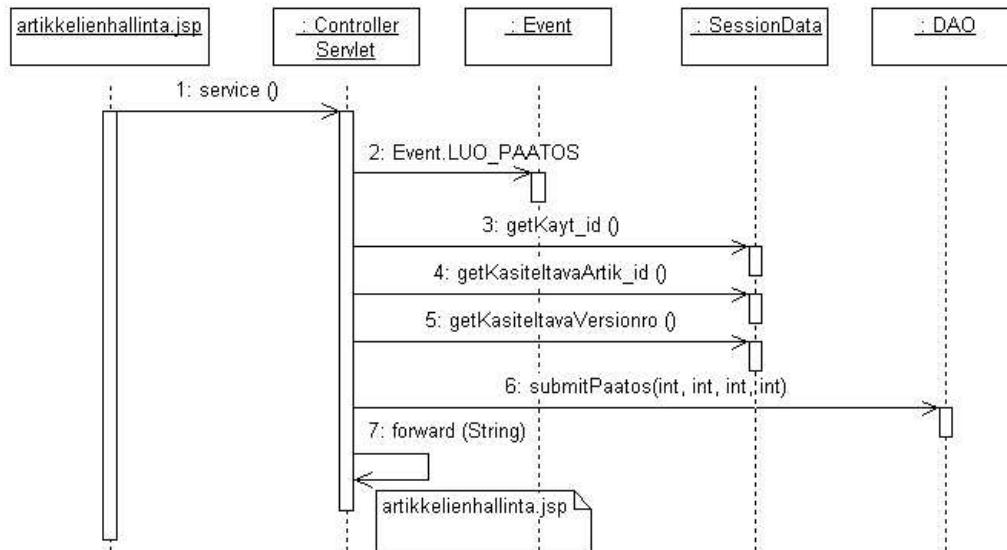
Ohjelmistossa pyritään siihen, että mikään yksittäinen ohjelmistovirhe ei kaada koko järjestelmää, vaan muiden osien käyttö on mahdollista.

Jos ohjelman try-lohkossa syntyy poikkeus, sitä vastaavassa catch-lohkossa pidetään aina huoli siitä, että lokiin jää merkintä poikkeuksesta sekä riittävästi dataa sen jäljittämiseksi. Käytännössä tämä tarkoittaa poikkeuksen tyyppin ja poikkeuksen pinolistauksen (stacktrace) tallentamista lokitiedostoon.

## 4.3 Laitteistovirheiden käsittely

Laitteistovirheet on pyritty ottamaan huomioon ensisijaisesti sitoutumiskäytäntöä suunniteltaessa. Järjestelmän kaatuessa laitteistovirheen johdosta käyttäjän syöttämää tietoa saattaa kadota, mutta tietokanta pysyy eheänä. Virheen sattuessa tai istunnon päättyessä esimerkiksi selaimen sulkemisen tai uloskirjautumisen vuoksi mahdolliset väliaikatie-dostot ja keskeneräiset työt menetetään. Tomcatin ja Postgresin käynnistettyä laitteisto-virheen jälkeen järjestelmä on jälleen normaalisti käytettävissä.

Tietokantaoperaatioiden toteutuksesta ja sitoutumiskäytännöstä on tarkempi kuvaus luvussa 7.



Kuva 13: Sekvenssikaavio, päätöksen antaminen artikkelille.

## 5 Luokat ja metodit

Tässä luvussa esitellään järjestelmän luokat. Yleiskuvauksen järjestelmän luokista saa kuvan 15 luokkakaaviosta.

### 5.1 ControllerServlet

Pääservlet, joka muodostuu switch-case rakenteesta, jossa jokainen case-haara vastaa joltain eventtiä. Casen sisällä kutsutaan luokan samannimistä metodia kuin itse event. Metodin sisällä tehdään tarvittavat operaatiot, minkä jälkeen ohjataan pyynnöt oikealle JSP-sivulle.

Esimerkiksi:

```
Event.TALLENNA_REKISTEROINTI => tallennaRekisterointi()
```

*Metodit*

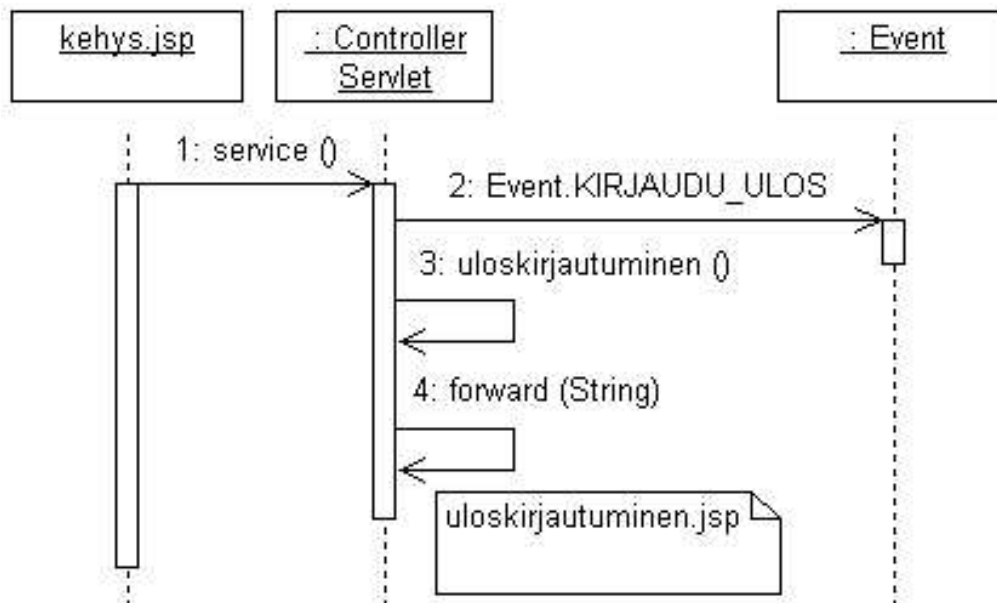
**private void forward(String sivu)**

Ohjaa parametrina annetulle sivulle. Asettaa sivun nimen myös SessionDatan muuttujaan setKasiteltavaSivu(String)-metodilla.

**private int getInt(String requestParameter)**

Palauttaa int-arvon olion request-muuttujasta parametrissa annetulle nimelle. Jos muunnos ei onnistu, niin palautetaan -1.

**private String getStr(String requestParameter)**



Kuva 14: Sekvenssikaavio, uloskirjautuminen.

Palauttaa String-arvon olion request-muuttujasta parametrissa annetulle nimelle. Jos muunnos ei onnistu, niin palautetaan tyhjä merkkijono

**public void service(HttpServletRequest req, HttpServletResponse res)**

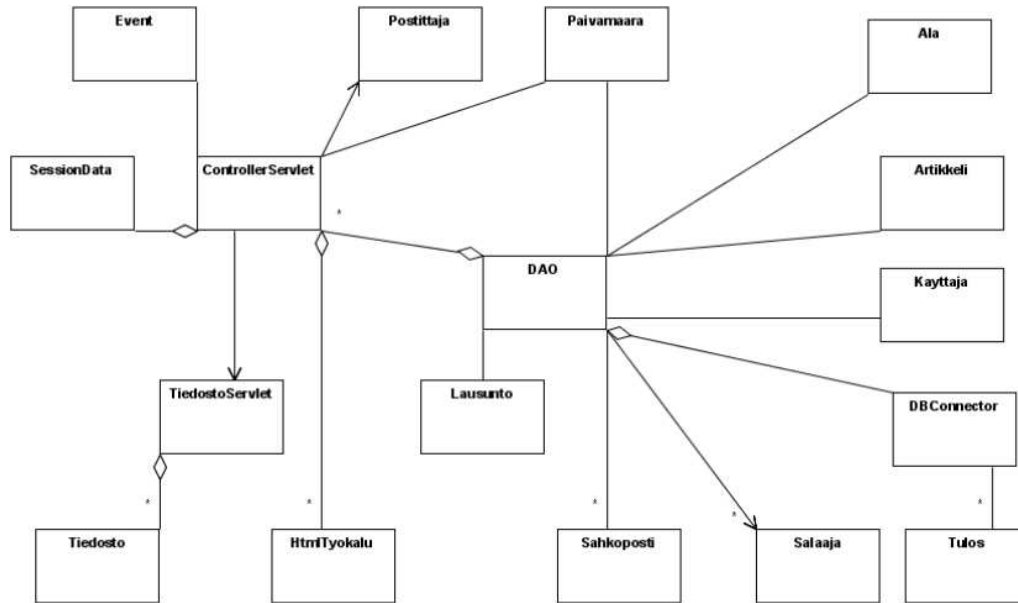
Päämetodi, jonka sisällä eventit tulkitaan.

**private void tallennaRekisterointi()**

Esimerkki tapahtumaa vastaavasta metodista, jonka koodi voisi esimerkiksi näyttää tältä:

```

String enimi=getStr(''enimi'');
...
Kayttaja kayt=new Kayttaja(enimi,...);
if(dao.submitKayttajatiedot(kayt,salasana)) {
    session.setAttribute(''sessionData'',
        new SessionData(kaytt_id,rooli));
    forward(''kehys.jsp?sivu=omatartikkelit.jsp'');
}
else {
    forward(''rekisteroituminen.jsp'');
}
  
```



Kuva 15: Luokkakaavio.

## 5.2 DAO

DAO-luokan avulla päästään käsiksi järjestelmän tietosisältöön. Luokan avulla voidaan lukea tai kirjoittaa tietokantaan.

Luokan metodien yhteydessä olevat tietokantakyselyiden viitteet viittaavaat testausuunitelmassa määriteltymiin SQL-kyselyihin.

### *Luokkavakiot*

```

int JARJESTA_ARTIKKELI_OTSIKKO = 1
int JARJESTA_ARTIKKELI_KIRJOITTAJA = 2
int JARJESTA_ARTIKKELI_PAIVAMAARA = 3
int JARJESTA_ARTIKKELI_AIHE = 4
int JARJESTA_ARTIKKELI_TILA = 5

```

```

int JARJESTA_ASiantuntija_NIMI = 10
int JARJESTA_ASiantuntija_ALA = 11

```

```

int JARJESTA_KAYTTAJA_NIMI = 20
int JARJESTA_KAYTTAJA_ROOLI = 21
int JARJESTA_KAYTTAJA_EMAIL = 22

```

### *Konstruktorit*

```

public DAO(DBConnector dbConnector)

```

Luo DAO-olion ja ottaa yhteyttä kantaa määriteltyyn DBConnector-olion avulla.

### *Metodit*

#### **public List getAlat()**

Palauttaa täydellisen listan tietokantaan tallennetuista aloista Ala-olioina.

Metodi käyttää tietokantakyselyä, joka hakee kaikki taulun Ala rivit (AS/TO-SQL-4).

#### **public Artikkelit getArtikkeli(int artik\_id, int versionro)**

Palauttaa Artikkelit-olion halutusta artikkelista.

Metodi käyttää tietokantakyselyä, joka liittää taulut Artikkelit ja Versio sekä hakee rivin tästä liitoksesta (KI/TO-SQL-3).

#### **public List getArtikkelit()**

Palauttaa kaikki artikkelit listana, joka sisältää Artikkelit-olioita.

Metodi käyttää tietokantakyselyä, joka liittää taulut Artikkelit ja Versio (TO-SQL-3).

#### **public List getArtikkelit(int jarjestysehto)**

Palauttaa kaikki artikkelit listana, joka sisältää Artikkelit-olioita. Lista on järjestetty järjestysehdon perusteella.

Metodi käyttää tietokantakyselyä, joka liittää taulut Artikkelit ja Versio (TO-SQL-3).

#### **public List getAsiantuntijanArtikkelit(int kayt\_id)**

Palauttaa listan Artikkelit-olioista, joissa kyseinen käyttäjä on arvostelijana.

Metodi käyttää tietokantakyselyä, joka hakee rivejä taulusta Lausuntopyynto (AS-SQL-1).

#### **public List getAsiantuntijat()**

Palauttaa Asiantuntija-listana kaikki asiantuntijat.

Metodi käyttää tietokantakyselyä, joka liittää taulut Kayttaja ja Asiantuntija ja hakee rivejä tästä liitoksesta (TO-SQL-4).

#### **public List getAsiantuntijat(int artik\_id, int versionro)**

Palauttaa haluttuun artikkeliin kiinnitetyt asiantuntijat.

Metodi käyttää tietokantakyselyä, joka liittää taulut Kayttaja, Asiantuntija ja Lausuntopyynto ja hakee rivejä tästä liitoksesta (TO-SQL-5).

#### **public List getAsiantuntijat(String hakusanat)**

Palauttaa Asiantuntija-oliolistan kaikista asiantuntijoista, joiden nimi- tai osaamisala-kenttä sisältää hakusanan.

Metodi käyttää tietokantakyselyä, joka liittää taulut Kayttaja ja Asiantuntija ja hakee rivejä tästä liitoksesta (TO-SQL-4).

**public List getKayttajat()**

Palauttaa listan Kayttaja-olioita kaikista järjestelmän käyttäjistä.

Metodi käyttää tietokantakyselyä, joka hakee kaikki taulun Kayttaja rivit (TO-SQL-1).

**public List getKayttajat(int jarjestysehto)**

Palauttaa listan Kayttaja-olioita järjestelmän käyttäjistä järjestettynä järjestysehdon perusteella.

Metodi käyttää tietokantakyselyä, joka hakee kaikki taulun Kayttaja rivit (TO-SQL-1).

**public Kayttaja getKayttajatiedot(int kayt\_id)**

Palauttaa Kayttaja-olion halutun käyttäjän tiedoista.

Metodi käyttää tietokantakyselyä, joka hakee rivin taulusta Kayttaja (KI/TO-SQL-1) tai käyttäjän ollessa asiantuntija tauluista Kayttaja ja Asiantuntija (AS/TO-SQL-1) sekä rivejä taulusta Osaamisala (AS/TO-SQL-2).

**public List getKirjoittajanArtikkelit(int kayt\_id)**

Palauttaa listan Artikkelio-olioita halutun käyttäjän artikkeleista.

Metodi käyttää tietokantakyselyä, joka liittää taulut Artikkelio ja Versio sekä hakee rivejä tästä liitoksesta (KI-SQL-2)

**public List getLausunnot(int artik\_id, int versionro, boolean kirjoittaja)**

Palauttaa artikkeliin liittyvät lausunnot Lausunto-listana. Jos boolean-muuttujan kirjoittaja arvo on true, palautetaan kirjoittajalle tarkoitetut lausunnot. Jos arvo on false, palautetaan toimitukselle tarkoitetut lausunnot.

Metodi käyttää tietokantakyselyitä, jotka hakevat rivejä tauluista Lausunto ja Liite (KI/TO-SQL-4).

**public Lausunto getLausunto(int kayt\_id, int artik\_id, int versionro, boolean kirjoittaja)**

Palauttaa tietyn asiantuntijan (kayt\_id) tietylle artikkelille antaman lausunnon Lausunto-oliona.

Metodi käyttää tietokantakyselyitä, jotka hakevat rivin taulusta Lausunto ja rivejä taulusta Liite (KI/TO-SQL5).

**public int getLausuntopyyntonTila(int artik\_id, int versionro, int kayt\_id)**

Palauttaa lausuntopyyntön tilan.

Metodi käyttää tietokantakyselyä, joka hakee rivin taulusta Lausuntopyynto (AS/TO-



SQL-5).

**public List getLehti(int nro, int vuosikerta)**

Palauttaa tiettyyn painokseen valitut artikkelit Artikkeli-listana

Metodi käyttää tietokantakyselyä, joka liittyy taulut Lehti ja Artikkeli ja hakee rivejä tästä liitoksesta (TO-SQL-7).

**public int getPaatos(int artik\_id, int versionro)**

Palauttaa artikkelin version päätöksen.

Metodi käyttää tietokantakyselyä, joka hakee rivin taulusta Paatos (KI/TO-SQL-7).

**public boolean insertAla(String alanNimi)**

Lisää uuden alaluokituksen.

Metodi käyttää tietokantaoperaatiota, joka lisää rivin tauluun Ala (TO-SQL-6).

**public void insertArtikkeliLehteen(int artik\_id, int nro, int vuosikerta)**

Lisää artikkelin lehteen.

Metodi käyttää tietokantaoperaatiota, joka päivittää taulun Artikkeli riviä (TO-SQL-11).

**public boolean insertAsiantuntija(Kayttaja uusikayttaja)**

Lisää järjestelmään uuden asiantuntijan.

Metodi käyttää tietokantaoperaatiota, joka lisää rivin tauluihin Kayttaja, Asiantuntija ja Sposti sekä rivejä tauluun Osaamisala(TO-SQL-2).

**public void insertAsiantuntijaArtikkeliin(int artik\_id, int versionro, int kayt\_id)**

Lisää artikkelille asiantuntijan.

Metodi käyttää tietokantaoperaatiota, joka päivittää taulun Lausuntopyynto riviä (TO-SQL-8).

**public boolean insertToimittaja(Kayttaja uusikayttaja)**

Lisää järjestelmään uuden toimittajan.

Metodi käyttää tietokantaoperaatiota, joka lisää rivin tauluihin Kayttaja, Asiantuntija ja Sposti(TO-SQL-15).

**public void lahetaMuistutus(int artik\_id, int versionro, int kayt\_id)**

Lähetää asiantuntijalle sähköpostitse muistutuksen lausuntopyyntöstä.

Metodi käyttää tietokantaoperaatiota, joka päivittää taulun Lausuntopyynto riviä (TO-SQL-13).

**public void removeAla(int id)**

Poistaa valitun erikoisalan ala-listasta.

Metodi käyttää tietokantaoperaatiota, joka poistaa rivin taulusta Ala (TO-SQL-18).

**public boolean removeArtikkeli(int artik\_id)**

Poistaa valitun artikkelin ja palauttaa tiedon onnistumisesta.

Metodi käyttää tietokantaoperaatiota, joka poistaa rivin taulusta Artikkeli ja rivin/rivejä taulusta Versio (TO-SQL-20).

**public void removeArtikkeliLehdesta(int artik\_id, int nro, int vuosikerta)**

Poistaa artikkelin lehdestä.

Metodi käyttää tietokantaoperaatiota, joka päivittää taulun Artikkeli riviä (To-SQL-12).

**public void removeAsiantuntijaArtikkelista(int artik\_id, int versionro, int kayt\_id)**

Poistaa artikkelilta asiantuntijan.

Metodi käyttää tietokantaoperaatiota, joka päivittää taulun Lausuntopyyntö riviä (TO-SQL-9).

**public void setLausuntoLukittu(int artik\_id, int versionro, int kayt\_id, boolean lukittu)**

Asettaa lausunnon lukituksi tai poistaa lukituksen.

Metodi käyttää tietokantaoperaatiota, joka päivittää riviä taulussa Lausunto (TO-SQL-17).

**public void setLausuntopyyntöTila(int artik\_id, int versionro, int kayt\_id, int tila)**

Asettaa lausuntopyyntö tilan artikkelille.

Metodi käyttää tietokantaoperaatiota, joka päivittää taulun Lausuntopyyntö riviä (AS-SQL-3, AS-SQL-4, AS-SQL-5 ja TO-SQL-10).

**public void setVersioLukittu(int artik\_id, int versionro, boolean lukittu)**

Asettaa artikkelin version lukituksi tai poistaa lukituksen.

Metodi käyttää tietokantaoperaatiota, joka päivittää riviä taulussa Versio (TO-SQL-16).

**public boolean submitArtikkeli(int kayt\_id, Artikkelin artikkeli)**

Päivittää artikkelin muuttuneet tiedot. Jos artikkelia ei löydy tietokannasta, lisätään se sinne.

Metodi käyttää tietokantaoperaatioita, jotka lisäävät rivin tauluun Artikkelin ja Versio (KI-SQL-3), tai artikkelin ollessa uudempi versio vanhasta artikkelista lisätään rivi tauluun Versio (KI-SQL4).

**public boolean submitKayttajatiedot(Kayttaja uusiKayttaja, String salasana)**

Päivittää tietokannassa olevaa käyttäjätietoriviä. Jos riviä ei löydy, sellainen luodaan. Jos salasana-String on alustettu, asetetaan se käyttäjän uudeksi salasanaksi. Jos salasana on null, siihen ei tehdä muutoksia.

Käyttäjän ollessa uusi metodi käyttää tietokantaoperaatiota, joka lisää rivin tauluihin Kayttaja ja Sposti (KI-SQL-1) tai käyttäjän ollessa asiantuntija tauluihin Kayttaja, Asiantuntija ja Sposti sekä rivejä tauluun Osaamisala. Jos käyttäjä on jo olemassa päivitetään edelläänmainituissa tauluissa olevia rivejä (KI/TO-SQL-2 ja AS/TO-SQL-3).

**public boolean submitLausunto(int kayt\_id, int artik\_id, int versionro, Lausunto kirjoittajaLausunto, Lausunto toimitusLausunto)**

Päivittää lausunnon muuttuneet tiedot. Jos lausuntoa ei löydy tietokannasta, lisätään se sinne.

Metodi käyttää tietokantaoperaatiota, joka lisää rivin tauluun Lausunto, tai lausuntoa muokatessa päivittää riviä taulussa Lausunto (AS-SQL-2).

**public void submitPaatos(int kayt\_id, int artik\_id, int versionro, int paatos)**

Lähetää päätöksen.

Metodi käyttää tietokantaoperaatiota, joka lisää rivin tauluun Paatos (TO-SQL-21).

**public void submitViesti(String viesti, int artik\_id, int versionro, boolean toimittajalta)**

Lähetää viestin toimittajalta päätoimittajalle tai päätoimittajalta toimittajalle.

Metodi käyttää tietokantaoperaatiota, joka päivittää taulun Versio riviä (TO-SQL-14).

**public boolean tarkistaKirjautuminen(String tunnus, String salasana)**

Tarkistaa käyttäjätunnuksen ja salasanan oikeellisuuden.

Metodi käyttää tietokantakyselyä, joka hakee taulusta Sposti sähköpostiosoitetta vastaavan salasanan(SQL-1).

**public boolean tarkistaSahkoposti(String sahkoposti)**

Palauttaa tiedon löytyykö parametrina saatu sähköpostiosoite jo järjestelmästä.

Metodi käyttää tietokantakyselyä, joka hakee rivin taulusta Sposti (SQL-2).

**public void updateAla(int id, String uusiNimi)**

Vaihtaa halutun alan nimeä.

Metodi käyttää tietokantaoperaatiota, joka päivittää riviä taulussa Ala (TO-SQL-19).

## 5.3 SessionData

Jokaisen sisäänkirjautumisen yhteydessä luotava olio, joka sisältää käyttäjän istuntokohtaiset muuttujat. Luokan avulla voidaan varmistaa käyttöoikeudet eri operaatioita tehdessä. Lisäksi luokan apumuuttujien avulla voidaan vähentää tietokantahakuja. Olio on olemassa koko istunnon ajan sessiossa, josta se saadaan käyttöön joka sivulle automaattisesti.

*Konstruktorit*

**public SessionData(String sessionId, int kayt\_id, int rooli)**

Luo ilmentymän SessionData-luokasta ja sijoittaa parametrit kenttien arvoksi.

*Metodit*

**public DBConnector getDbConnector()**

Palauttaa dbConnector-muuttujan arvon.

**public int getKasiteltavaArtik\_id()**

Palauttaa kasiteltavaArtik\_id-muuttujan arvon.

**public int getKasiteltavaSivu()**

Palauttaa kasiteltavaSivu-muuttujan arvon.

**public int getKasiteltavaVersionro()**

Palauttaa kasiteltavaVersionro-muuttujan arvon.

**public int getKayt\_id()**

Palauttaa kayt\_id-muuttujan arvon.

**public int getLisattavaArtikkeli()**

Palauttaa lisattavaArtikkeli-muuttujan arvon.

**public int getRooli()**

Palauttaa rooli-muuttujan arvon.

**public String getSessionId()**

Palauttaa sessionId-muuttujan arvon.

**public void setDbConnector(DBConnector dbConnector)**

Asettaa dbConnector-muuttujalle uuden arvon

**public void setKasiteltavaVersionro(int kasiteltavaVersionro)**

Asettaa kasiteltavaVersionro-muuttujalle uuden arvon

**public void setKayttajaTiedot(Kayttaja kayttajaTiedot)**

Asettaa kayttajaTiedot-muuttujalle uuden arvon

**public void setLisattavaArtikkeli(Artikkeli lisattavaArtikkeli)**

Asettaa lisattavaArtikkeli-muuttujalle uuden arvon

## 5.4 Event

Luokka sisältää joukon int-tyyppisiä luokkavakioita. Vakiot kuvaavat jsp-sivuilla aiheutettuja operaatioita. Kaikki luokan sisältämät vakiot on kuvattu litteessä 10.4.

## 5.5 HtmlTyokalu

Luokan metodien avulla saadaan usein toistuvista HTML-elementeistä (artikkelilistat, käyttäjälisat jne.) valmis HTML-esitys. Tarvittaessa metodille annetaan parametrina List-olio generoitavasta tietosisällöstä.

*Metodit*

**public String getArtikkelit(List artikkelit)**

Palauttaa artikkelilistan HTML-esityksen.

**public String getKirjoittajat(List kirjoittajat)**

Palauttaa kirjoittajalistan HTML-esityksen.

## 5.6 TiedostoServlet

Servlet-olio, jonka avulla sivulla voidaan näyttää tiedostoja, joihin käyttäjällä ei ole suoraa pääsyä. Lisäksi luokan avulla voidaan rajoittaa esimerkiksi tiedostoon kohdistuvia lukemiskertoja. Sivu saa parametrit kayt\_id, artik\_id, versionro ja tiedostonimi. Näillä parametreillä löydetään oikea tiedosto. Servlet palauttaa lopulta binääriesityksen tiedostosta.

## 5.7 Tiedosto

TiedostoServletin apuluokka, jonka avulla voidaan ladata tiedostoja ei-julkisista hakemis-  
toista ohjelman käyttöön.

*Metodit*

**public static BufferedOutputStream naytaTiedosto(String polku)**  
Palauttaa tiedoston bittivirtana.

**public static boolean poistaTiedosto(String polku)**  
Poistaa halutun tiedoston.

**public static boolean tallennaTiedosto(String nimi, String polku, HttpRequest request)**  
Tallettaa HTML-lomakkeelta tulevan tiedoston haluttuun paikkaan.

## 5.8 Postittaja

Luokan avulla voidaan lähettää käyttäjille sähköpostia.

*Metodit*

**public static boolean laheta(Sahkoposti lahetettava)**  
Lähettää viestin.

## 5.9 Salaja

Luokan avulla salataan salasanoja ja muuta salattavaa tietoa. Salasanojen salaamiseen  
käytetään Javan security-pakkauksesta löytyvää MessageDigest-luokkaa.

*Metodit*

**public static String salaa(String teksti)**  
Palauttaa parametrilla saadun tekstin MD5-algoritmilla salatussa muodossa.

## 5.10 DBConnector

Luokan avulla otetaan yhteyttä tietokantaan ja suoritetaan tietokantakyselyjä.

*Luokkavakiot*

String AJURI = käytettävän tietokanta-ajurin tyyppi  
String SALASANA = tietokannan salasana  
String TUNNUS = tietokannan käyttäjätunnus  
String URL = tietokannan osoite

#### *Konstruktorit*

##### **public DBConnector()**

Ottaa yhteyden tietokantaan luokkavakioissa määritellyillä tunnuksilla.

#### *Metodit*

##### **public boolean close()**

Sulkee yhteyden tietokantaan.

##### **public Tulos execute(String kysely)**

Tekee tietokantakyselyn ja palauttaa tuloksen Tulos-oliona.

##### **public int getId(String sekvenssi)**

Palauttaa pyydetyn sekvenssin seuraavan arvon.

## **5.11 Tulos**

Oma tietorakenne tietokantakyselyiden tulosten tallentamiseen. Sisältää ResultSet-luokan metodeita vastaavat metodit, mutta luokan avulla voidaan käyttää aikaisempia tuloksia hyväksi tehtäessä uusia kyselyjä.

#### *Konstruktorit*

##### **public Tulos(ResultSet resultSet)**

Luo ResultSet-oliota vastaavan Tulos-olion.

#### *Metodit*

##### **public Date getDate(String sarake)**

Palauttaa käsiteltävän rivin ”sarake”-nimisen sarakkeen arvon Date-oliona.

##### **public int getInt(String sarake)**

Palauttaa käsiteltävän rivin ”sarake”-nimisen sarakkeen arvon int-arvona.

##### **public int getRivit()**

Palauttaa rivien lukumäärän.

**public String getString(String sarake)**

Palauttaa käsiteltävän rivin ”sarake”-nimisen sarakkeen arvon String-oliona.

**public boolean hasNext()**

Kertoo onko Tulos-oliassa vielä rivejä.

**public Object next()**

Siirtyy Tulos-olion seuraavalle riville ja palauttaa sen arvon.

## 5.12 Sahkoposti

*Konstruktorit***Sahkoposti()**

Luo uuden Sahkoposti-olion.

**Sahkoposti(String lahettaja, String vastaanottaja, String otsikko, String teksti)**

Luo uuden sähköposti-olion.

*Metodit***public String getLahettaja()**

Palauttaa lahettaja-muuttujan arvon.

**public String getOtsikko()**

Palauttaa otsikko-muuttujan arvon.

**public String getTeksti()**

Palauttaa teksti-muuttujan arvon.

**public String getVastaanottaja()**

Palauttaa vastaanottaja-muuttujan arvon.

## 5.13 Paivamaara

Luokka päivämäärien käsittelyyn.

*Konstruktorit***public Paivamaara()**



Luo uuden Paivamaara-olion, jonka kenttien arvoksi tulee olion luontihetken päivämäärä.

**public Paivamaara(Date pvm)**

Luo Date-oliota pvm vastaavan Paivamaara-olion.

**public Paivamaara(String pvm)**

Luo String-oliota pvm vastaavan Paivamaara-olion.

**public Paivamaara(String paiva, String kuukausi, String vuosi)**

Luo Paivamaara-olion annetuilla arvoilla.

*Metodit*

**public String getKuukausi()**

Palauttaa kuukausi-muuttujan arvon.

**public String getPaiva()**

Palauttaa paiva-muuttujan arvon.

**public String getPvmKantaan()**

Palauttaa päivämäärän muodossa, joka voidaan tallentaa tietokantaan.

**public String getPvmSivulle()**

Palauttaa päivämäärän muodossa, joka voidaan näyttää käyttöliittymässä.

**public String getVuosi()**

Palauttaa vuosi-muuttujan arvon.

## 5.14 Lausunto

*Luokkavakiot*

int HYLATTY = 1

int HYVAKSYTTY\_SUURIN\_MUUTOKSIN = 2

int HYVAKSYTTY\_PIEININ\_MUUTOKSIN = 3

int HYVAKSYTTY = 4

int PYYNTO\_VALITTU = 1

int PYYNTO\_EL\_VASTANNUT = 2

int PYYNTO\_HYLANNYT = 3

int PYYNTO\_HYVAKSYNYT = 4

#### *Konstruktorit*

**public Lausunto(String asiantuntija, List tiedostot, String teksti, int paatos)**

Luo uuden Lausunto-olion.

#### *Metodit*

**public String getAsiantuntija()**

Palauttaa asiantuntija-muuttujan arvon.

**public int getPaatos()**

Palauttaa paatos-muuttujan arvon.

**public String getTeksti()**

Palauttaa teksti-muuttujan arvon.

**public List getTiedostot()**

Palauttaa tiedostot-muuttujan arvon.

## 5.15 Artikkelit

#### *Konstruktorit*

**public Artikkelit(int artik\_id, String otsikko, Paivamaara pvm, String kirjoittaja, int tila, Url teksti, int versio, List asiantuntijat)**

Luo uuden Artikkelit-olion ja alustaa sen kentät parametrien arvoilla. Asiantuntijat ovat Javan List-rakenteessa olevia Kayttaja-olioita.

#### *Metodit*

**public int getArtik\_id()**

Palauttaa artik\_id-muuttujan arvon.

**public List getAsiantuntijat()**

Palauttaa asiantuntijat-muuttujan arvon.

**public String getKirjoittaja()**

Palauttaa kirjoittaja-muuttujan arvon.

**public String getLausunto()**

Palauttaa lausunto-muuttujan arvon.

**public String getOtsikko()**

Palauttaa otsikko-muuttujan arvon.

**public Paivamaara getPvm()**

Palauttaa pvm-muuttujan arvon.

**public Url getTeksti()**

Palauttaa teksti-muuttujan arvon.

**public String getTiivistelma()**

Palauttaa tiivistelma-muuttujan arvon.

**public int getTila()**

Palauttaa tila-muuttujan arvon.

**public int getVersio()**

Palauttaa versio-muuttujan arvon.

## 5.16 Kayttaja

*Luokkavakiot*

int KAYTTAJA = 1

int ASIAANTUNTIJA = 2

int TOIMITTAJA = 3

int PAATOIMITTAJA = 4

*Konstruktorit*

**public Kayttaja(int kayt\_id, String etunimi, String sukunimi, String katuosoite, String kaupunki, int postinumero, String maa, String osavaltio, String sahkoposti, int rooli, List erikoisalat)**

Luo uuden Kayttaja-olion ja alustaa sen kentät parametrien arvoilla.

*Metodit*

**public String getAktiivinenEmail()**

Palauttaa aktiivinenEmail-muuttujan arvon.

**public String getEmailLista()**

Palauttaa emailLista-muuttujan arvon.

**public List getErikoisala()**

Palauttaa erikoisala-muuttujan arvon.

**public String getEtunimi()**

Palauttaa etunimi-muuttujan arvon.

**public String getKatuosoite()**

Palauttaa katuosoite-muuttujan arvon.

**public String getKaupunki()**

Palauttaa kaupunki-muuttujan arvon.

**public int getKayt\_id()**

Palauttaa kayt\_id-muuttujan arvon.

**public String getMaa()**

Palauttaa maa-muuttujan arvon.

**public String getOsavaltio()**

Palauttaa osavaltio-muuttujan arvon.

**public int getPostinumero()**

Palauttaa postinumero-muuttujan arvon.

**public int getRooli()**

Palauttaa rooli-muuttujan arvon.

**public String getSukunimi()**

Palauttaa sukunimi-muuttujan arvon.

## 5.17 Ala

*Konstruktorit*

**public Ala(int ala\_id, String nimi)**

Luo uuden Ala-olion ja asettaa kenttien alkuarvot.

*Metodit*

**public int getAla\_id()**

Palauttaa ala\_id-muuttujan arvon.

**public String getNimi()**

Palauttaa nimi-muuttujan arvon.

## **6 JSP-sivut**

Jokaisen sivun alussa tarkistetaan, että sessio on voimassa. Jos ei, käyttäjä ohjataan sisäänkirjautumissivulle virheilmoituksen kera.

### **6.1 Kaikkien käyttäjien sivut**

#### **6.1.1 header.jsp**

Jokaisen sivun alkuun ladattava JSP-sivu. Tarkistaa onko istunto auki, riittävätkö käyttäjän oikeudet sivulle sekä lataa tarvittavat HTML-tagit sivun alkuun.

#### **6.1.2 footer.jsp**

Jokaisen sivun loppuun ladattava JSP-sivu. Tulostaa tarvittavat HTML:n lopetustagit sivulle.

#### **6.1.3 kehys.jsp**

Pääsivu, joka määrittelee sivun kehyksien asettelut. Sisältää kaksi kehystä: sivun vasempaan reunaan navigointivalikon (menu.jsp) ja pääkehysten. Pääkehykseen ladataan aluksi käyttäjän roolin mukaan omatartikkelit.jsp (kirjoittaja), artikkelienarvostelu.jsp (asiantuntija), artikkelienhallinta.jsp (toimittaja) tai paatoimittajanakyma.jsp (päätoimittaja).

#### **6.1.4 menu.jsp**

Sivun vasemmassa laidassa oleva navigointivalikko. Valikon linkit johtavat suoraan eri JSP-sivuille, ja se mukautuu käyttäjän roolin mukaan. Kirjoittajalle näytetään kirjoittajan JSP-sivut, asiantuntijalle kirjoittajan sekä asiantuntijan sivut ja toimitukselle kaikki.

### 6.1.5 kirjautuminen.jsp

Ensimmäisenä aukeava sivu. Sisältää e-mail- ja salasana kentät. Kenttien tiedot tarkistetaan ”Login”-painikkeen painamisen jälkeen. Onnistuneen kirjautumisen jälkeen käyttäjä ohjataan oikealle JSP-sivulle käyttäjäryhmänsä mukaan.

### 6.1.6 profiilinmuokkaus.jsp

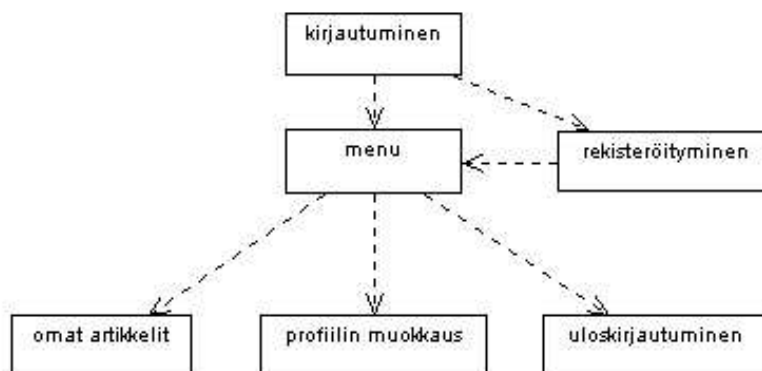
Sivulle ladataan lomake, joka on täytetty valmiiksi käyttäjän tiedoilla. Käyttäjän tehtyä muutokset ja painettua ”Update profile” -painiketta muuttuneet tiedot viedään tietokantaan ja käyttäjä saa ilmoituksen päivityksen onnistumisesta.

Lomakkeen ulkoasu muuttuu käyttäjän ryhmän mukaan, esimerkiksi asiantuntija voi muokata omia erikoisalojaan sivun kautta.

### 6.1.7 uloskirjautuminen.jsp

Suljetaan käyttäjän istunto ja poistetaan istuntoon liittyvät muuttujat muistista. Tulostetaan käyttäjälle hyvästelyviesti.

## 6.2 Kirjoittajan sivut



Kuva 16: Kirjoittajan navigointi.

### 6.2.1 rekisteroityminen.jsp

Lomake, jonka avulla käyttäjä voi rekisteröityä kirjoittajaksi järjestelmään. Lomakkeen täyttämisen jälkeen tarkistetaan kenttien oikeellisuus ja tallennetaan tiedot tietokantaan. Tämän jälkeen käyttäjä ohjataan sisäänkirjautumissivulle.

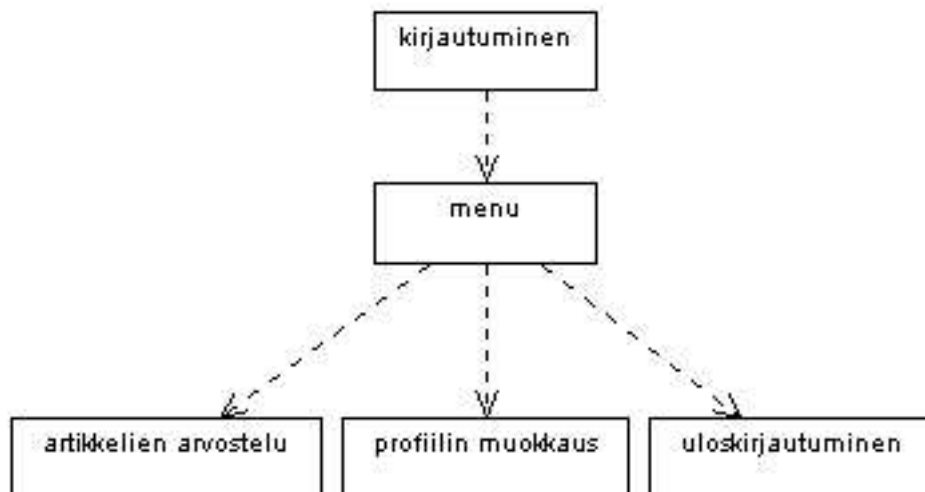
### 6.2.2 omatartikkelit.jsp

Omat artikkelit -sivu sisältää HTMLTyokalulla tuotetun artikkelilistan. Jos parametriä artik\_id ei ole määritelty, tulostetaan lomake, jolla voidaan lähettää uusi artikkeli. Mikäli artik id on määritelty, näytetään kyseisen artikkelin tiedot.

Uusi artikkeli pitää voida lähettää myös PS- tai PDF-muodossa. Tätä varten sivulla on tekstikenttä, jonka avulla voidaan lähettää binäärimuotoisia tiedostoja. Lähetettyään uuden artikkelin käyttäjä saa ilmoituksen toimenpiteen onnistumisesta, ja artikkeli vietään tietokantaan.

Kirjoittaja voi järjestää artikkelit haluamaansa järjestykseen painamalla listan otsikkorivillä olevia linkkejä (otsikko, aihepiiri, lähetyspäivä).

## 6.3 Asiantuntijoiden sivut



Kuva 17: Asiantuntijan navigointi.

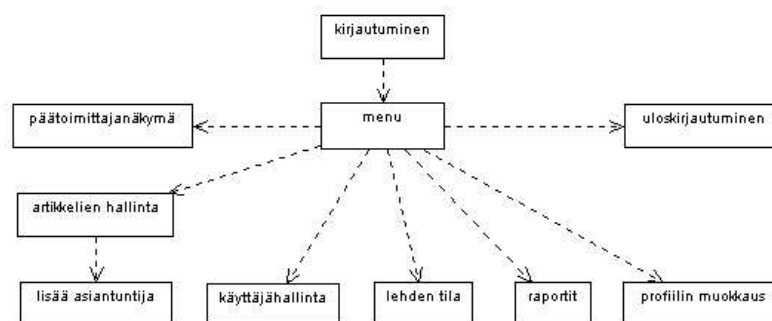
### 6.3.1 artikkelienarvostelu.jsp

Asiantuntija näkee listan artikkeleista, joihin hänet on merkattu arvostelijaksi. Klikkaamalla artikkelin otsikkoa, avautuu artikkelin informaatio sivu alakehykseen. Jos artikkeli on uusi, kysytään asiantuntijalta haluaako hän toimia kyseisen artikkelin arvostelijana. Kun asiantuntija painaa ”Yes”-painiketta, jää siitä jälki tietokantaan, ja artikkeli siirtyy tilaan, jossa asiantuntijalta odotetaan palautetta.

Jos asiantuntija on hyväksynyt artikkelin arvosteltavaksi ja painaa linkkiä kyseisen artikkelin kohdalta, saa hän arvostelulomakkeen täytettäväksi. Painettuaan ”send review”-painiketta lausunto siirtyy tietokantaan ja artikkeli muuttuu arvostelluksi asiantuntijan osalta. Painaessaan linkkiä arvostellun artikkelin kohdalta asiantuntija näkee artikkelin tiedon ja antamansa palautteen.

Asiantuntija voi järjestää artikkelit haluamaansa järjestykseen painamalla listan otsikko-rivillä olevia linkkejä (otsikko, aihepiiri, lähetyspäivä).

## 6.4 Toimituksen sivut



Kuva 18: Toimituksen navigointi.

### 6.4.1 artikkelienhallinta.jsp

Sivulla näkyy oletuksena luettelo artikkeleista, jotka odottavat toimituksen toimenpiteitä. Luettelon yhteydessä on myös linkki, jota painamalla saadaan luettelon kaikista artikkeleista.

Kun artikkelin linkkiä painetaan, avautuu alempaan kehykseen artikkelin tiedot. Sivulla sisältää artikkelin perustiedot (sekä linkin varsinaiseen artikkeliin), asiantuntijoiden valintalomakkeen sekä viestilaatikon päätoimittajalle.



Toimittaja voi valita artikkelille asiantuntijat valintalistassa. Kun toimittaja valitsee vasemmalla puolella olevasta luettelosta haluamansa asiantuntijan artikkelille ja painaa ”add referee to article” -linkkiä, siirtyy hänen valitseman asiantuntija oikeanpuoleen luetteloon, jossa on lueteltu artikkelille valitut asiantuntijat. Vastaavasti toimittaja voi poistaa artikkelin asiantuntijoita artikkelista ”remove referee from article” -linkistä. Jos asiantuntija puuttuu luettelosta, voidaan uusi lisätä samalta sivulta löytyvällä lomakkeella. Toimittajan täydennettyä asiantuntijan tiedot ja painettua ”add referee” -nappia, arvot viedään tietokantaan ja asiantuntija ilmestyy asiantuntijaluetteloon valittavaksi.

Jos toimittaja haluaa tiedustella päätoimittajalta artikkeliin liittyvää asiaa, voi hän kirjoittaa viestin sille varattuun laatikkoon.

Jos artikkeli on käynyt arvostelukierroksen läpi, avautuu alempaan kehykseen asiantuntijoiden palautteet. Toimittaja näkee sivulta tulleet palautteet ja hän voi tarvittaessa muokata niitä. Lisäksi toimittaja voi hallita liitetiedostoja sivun avulla. Painaessaan ”save and send information to writer” -nappia ohjataan kirjoittajalle osoitetut palautteet ja liitetiedostot Sähköposti-oliona sähköposti.jsp-sivulle.

#### **6.4.2 lisaasiantuntija.jsp**

Artikkelinhallinnasta avautuva lomake, jolla voidaan lisätä uusia asiantuntijoita järjestelmään.

#### **6.4.3 lehdentila.jsp**

Lehden tila -sivulla toimittajalle aukeaa luettelo artikkeleista, joita ei ole vielä lisätty mihinkään lehteen, mutta joihin on saatu asiantuntijoilta arviot. Sivulla on pudotusvalikko lehden numeroa varten. Kun käyttäjä valitsee jonkun numeron, näytetään kyseiseen lehteen valitut artikkelit sivun alalaidassa olevassa luettelossa.

Sivun kautta voidaan lähettää myös muistutuksia arvostelijoille Painamalla ”send reminder” -painiketta. Kun muistutusviesti on lähetetty, tallennetaan tietokantaan aikaleima muistutuksen lähetyksestä.

#### **6.4.4 kayttajahallinta.jsp**

Sivun yläreunaan avautuu järjestettävissä oleva luettelo (joka saadaan HTMLTyokaluluokalta) käyttäjistä. Painamalla käyttäjän nimeä avautuu alakehykseen käyttäjän tiedoilla täydennetty lomake, johon toimittaja voi tehdä muutoksia. ”Update fields” -painiketta painamalla tiedot siirtyy tietokantaan.

#### **6.4.5 raportit.jsp**

Raportit-sivu on staattinen sivu, johon on kerätty erilaisia tietokantakyselyitä.

### 6.4.6 sahkoposti.jsp

Sähköposti-sivulla toimittaja voi lähettää sähköpostia järjestelmän sidosryhmille. Oletuksena toimittajalle aukeaa tyhjä sivu, johon hän voi täyttää tarvittavat tiedot. Painaessaan ”send”-painiketta kentän tiedot siirtyvät Sähköposti-oliona Postittaja-luokalle, joka lähettää viestin eteenpäin. Sivua voidaan kutsua myös Sähköposti-muotoisella parametrilla, jolloin lomake täytetään kyseisen olion sisältämällä arvoilla.

### 6.4.7 päätoimittajanäkymä.jsp

Päätoimittajan näkymä vastaa muun toimituksen näkymää artikkeleiden hallintaan, mutta päätoimittajalle lähetettävän viestikentän sijaan päätoimittajalla on mahdollisuus vastata hänelle lähetettyihin viesteihin. Päätoimittajan artikkeliluettelossa näytetään oletuksena vain ne artikkelit, jotka odottavat päätoimittajan kommentointia ja asiantuntijaehdotuksia.

## 7 Tietokantakuvaus

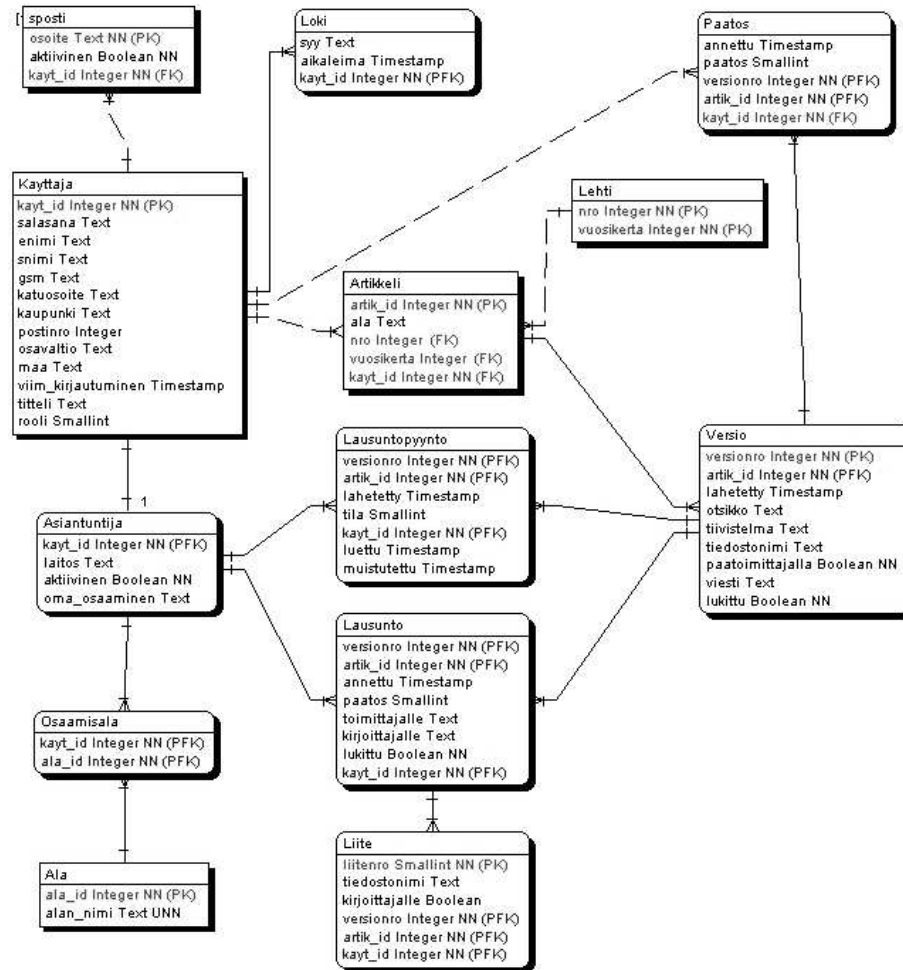
Tässä luvussa esitellään tietokannan suunnitteluun liittyvän yleisen kuvauksen lisäksi tietokantataulujen keskinäiset suhteet (kuva 19) ja luokkien yksityiskohtaiset luokkakuvaukset.

### 7.1 Tietokannan rakenne

Tietokannan rakenne täyttää seuraavat hyvän relaatiokaavion ominaisuudet:

- Kaavion merkitys on helposti selitettävissä
- Kaaviossa ei esiinny tarpeetonta toisteisuutta
- Tyhjiä arvoja esiintyy vain satunnaisesti
- Kaaviossa ei esiinny päivitysanomalia
- Avainattribuuttien arvot ovat pysyviä

Tietokannan kaikki relaatiot ovat Boyce-Codd -normaalimuodossa, jolloin relaatioissa ei esiinny lainkaan toisteisuutta. Minkään monikon mitään attribuuttiarvoa ei voida päätellä relaation muista monikoista kaavion funktionaalisia riippuvuuksia käyttäen. Itse asiassa ainoat funktionaaliset riippuvuudet ovat avainriippuvuuksia; tällaiset relaatiot ovat aina BCNF-muodossa. Väliaikaisia tyhjiä arvoja esiintyy lähinnä aikaleimojen yhteydessä.



Kuva 19: Tietokantakaavio.

## 7.2 Tietokantakyselyt ja transaktioiden sitoutuminen

Tietokantaa ja sen käyttöä suunniteltaessa lähtökohtana on ollut tietokannan eheyden säilyttäminen virhetilanteissa. Järjestelmän kaatuessa esimerkiksi laitteistovirheen vuoksi tietoa saatetaan menettää, sillä tietokantaan kohdistuvat päivitysoperaatiot pidetään mahdollisimman atomisina, ts. ne joko suoritetaan kokonaan tai ei ollenkaan.

Käyttäjän tekemät päivitykset tehdään tietokantaan vasta siinä vaiheessa, kun kaikki tarvittavat tiedot on saatu. Käytännössä siis käyttäjän käyttöliittymäsivuilla syöttämät tiedot tallennetaan tietokantaan syötteiden oikeellisuuden tarkistamisen jälkeen, kun käyttäjä on lähettänyt lomakkeen tiedot järjestelmälle. Virheellisten syötteiden tapauksessa käyttöliittymäsivu asianmukaisine virheilmoituksineen näytetään käyttäjälle uudestaan, mutta päivityksen tuottava transaktio ei ala eikä sitoudu ennen korjauksia. Transaktiot pidetään siis mahdollisimman lyhyinä, jolloin suurin osa likaisista luvuista ja kirjoituksista saadaan estettyä.

Joissakin tapauksissa anomalioita saattaa kuitenkin esiintyä. Käyttöliittymäsivujen generoimiseen voidaan nimittäin käyttää kahden eri tietokantakyselyn tulosjoukkoa, jolloin kyselyiden välissä tapahtuvat muutokset eivät näy jälkimmäistä kyselyä suoritettaessa; operointi tapahtuu siis vanhentuneilla monikoiden arvoilla. Tämä mahdollistaa eräänlaiset toistokelvottomat luvut, mutta tietokanta pysyy silti eheässä tilassa. Esimerkiksi kirjoittajan artikkeleita haettaessa artikkelilista luodaan sisäkkäisten kyselyiden avulla: ulommaisessa silmukassa haetaan tiettyyn kirjoittajaan liittyvät artikkelit ja sisemmässä kyseiseen artikkeliin liittyvät lausunnot.

Koska yhdellä käyttäjällä käytössä olevien tietojen lukitseminen muilta käyttäjiltä ei pidetä varteenotettavana ratkaisuna, on mahdollista että tietoa häviää päivitysten yhteydessä. Kirjoittaja ja toimittaja saattavat esimerkiksi samaan aikaan päivittää kirjoittajan tietoja, jolloin vain myöhemmin sitoutuneen transaktion aiheuttamat muutokset jäävät voimaan.

Selainpohjaisen käyttöliittymän vuoksi tällainen vanhentuneen tiedon käyttäminen päivityksiin on väistämättä mahdollista, sillä näytöllä näkyvän tiedon ajankohtaisuus on kiinni käyttäjästä itsestään. Yllämainitut seikat on kuitenkin pyritty ottamaan suunnittelussa huomioon tietokannan eheyden takaamiseksi.

### 7.3 Käyttäjä

Attribuutti	Arvojoukko	Kuvaus
kayt_id	luku	käyttäjän tunniste
salasana	teksti	käyttäjän salasana
enimi	teksti	etunimi
snimi	teksti	sukunimi
gsm	teksti	puhelinnumero
katuosoite	teksti	katuosoite
kaupunki	teksti	postitoimipaikka
postinro	luku	postinumero
maa	teksti	maa
osavaltio	teksti	osavaltio
arvo	teksti	arvonimi tai titteli
rooli	luku	kirjoittaja (1), asiantuntija (2), toimittaja (3), päätoimittaja(4)

Kirjoittaja-taulu sisältää järjestelmään rekisteröityneiden ja toimituksen lisäämien käyttäjien tiedot. Kaikilla rekisteröidyillä käyttäjillä on siis oikeus lähettää artikkeleita toimituksen ja asiantuntijoiden arvioitavaksi. Yhdellä kirjoittajalla voi olla yksi tai useampia sähköpostiosoitteita käytössään. Käyttäjä identifioidaan yksikäsitteisen käyttäjätunnisteen avulla.

## 7.4 Asiantuntija

Attribuutti	Arvojoukko	Kuvaus
kayt_id	viite	käyttäjän tunniste
laitos	teksti	työpaikka (yliopisto tai muu oppilaitos)
oma_osaaminen	teksti	vapaamuotoinen kuvaus osaamisaloista
aktiivinen	boolean	aktiivinen?

Asiantuntija on tavallisen käyttäjän, eli kirjoittajan erikoistapaus. Toimituksen lisäämällä asiantuntijalla on jokaiseen kirjoittajaan liittyvien tietojen ja identifioivan tunnuksen lisäksi laitos, jossa asiantuntija tällä hetkellä toimii, sekä rajoittamaton joukko osaamisaloja. Asiantuntijalla on hänelle arvioitavaksi lähetettyihin artikkeleihin liittyen joukko lausuntopyyntöjä niihin edelleen liittyviä lausuntoja. Asiantuntija voi valita, ettei halua toistaiseksi ottaa vastaan lausuntopyyntöjä, jolloin asiantuntijan tila muuttuu aktiivisesta poissaolevaksi.

## 7.5 Osaamisala

Attribuutti	Arvojoukko	Kuvaus
kayt_id	viite	käyttäjän tunniste
ala_id	viite	osaamisalan tunniste

Osaamisala-taulu sisältää asiantuntijoiden osaamisalat.

## 7.6 Ala

Attribuutti	Arvojoukko	Kuvaus
ala_id	luku	alan tunniste
alan_nimi	teksti	tieteenhaaran nimi

Ala-taulu sisältää kaikki tällä hetkellä valittavana olevat tieteenalat, joista asiantuntijat valitsevat omaa osaamistaan parhaiten kuvaavat alat.

## 7.7 Sähköposti

Attribuutti	Arvojoukko	Kuvaus
kayt_id	viite	käyttäjän tunniste
osoite	teksti	sähköpostiosoite
aktiivinen	boolean	aktiivinen tai passiivinen

Käyttäjällä voi olla useita sähköpostiosoitteita, joista käyttäjän kulloinkin valitsema toimii senhetkisenä aktiivisena osoitteena. Käyttäjää koskevat tiedotukset käsittelyn etenemisestä, uusista lausuntopyyntöistä ym. lähetetään aktiiviseen osoitteeseen.

## 7.8 Lehti

Attribuutti	Arvojoukko	Kuvaus
nro	luku	lehden numero
vuosikerta	luku	vuosikerta

Lehti koostuu määrittelemättömän monesta artikkelista ja se identifioidaan volyymin, eli lehden numeron ja vuosikerran, avulla.

## 7.9 Artikkelit

Attribuutti	Arvojoukko	Kuvaus
artik_id	luku	artikkelin tunniste
kayt_id	luku	kirjoittajan tunniste
ala	teksti	artikkeliin liittyvä tieteenhaara
nro	viite	lehden numero
vuosikerta	viite	lehden vuosikerta

Uudesta artikkelista luodaan rivi Artikkelit- ja Versio-tauluihin myöhempää versionhallintaa varten, ja se identifioidaan yksikäsitteisellä artikkelitunnuksella. Artikkelisiin liittyy yleensä useita uusintakierrosten ja korjausten perusteella syntyneitä versioita. Yhteistä kaikille artikkelin versioille on niihin liittyvä tieteenala.

## 7.10 Versio

Attribuutti	Arvojoukko	Kuvaus
artik_id	viite	artikkelin tunniste
versionro	luku	version numero
lahetetty	aikaleima	lähettämishetki
otsikko	teksti	otsikko
tiivistelma	teksti	raakamuotoinen tiivistelmä
tiedostonimi	teksti	viittaus artikkelitiedostoon
lukittu	boolean	uudelleenlähetyksen lukitus
paatoimittajalla	boolean	päätoimittajakierroksella tai toimituksella
viesti	teksti	toimituksen/päätoimittajan viesti päätoimittajalle/toimitukselle

Jokaisella artikkelin versiolla on oma versionumeronsa, jonka avulla versio identifioidaan. Uudelleenlähetysten yhteydessä sisällön lisäksi artikkelin otsikko ja raakatekstimuotoinen tiivistelmä saattavat muuttua. Yhden artikkelin tiettyyn versioon liittyy omat lausuntopyyntönsä ja lausuntonsa, ja eri saman artikkelin eri versioita voi olla arvioimassa eri asiantuntijoita.

## 7.11 Lausuntopyyntö

Attribuutti	Arvojoukko	Kuvaus
artik_id	viite	artikkelin tunniste
kayt_id	viite	asiantuntijan tunniste
versionro	viite	version numero
lahetetty	aikaleima	pyynnön lähettämishetki
tila	teksti	asiantuntija valittu (1), odottaa vastausta (2), hylätty (3), hyväksytty (4), poistettu (<0)
luettu	aikaleima	pyynnön lukemishetki
muistutettu	aikaleima	muistutuspyynnön lähettämishetki

Lausuntopyyntö koskee tiettyä artikkelin versiota, ja se identifioidaan artikkelin ja asiantuntijan tunnuksilla sekä versionumerolla. Lausuntopyynnöllä on kolme mahdollista tilaa: hyväksytty, hylätty tai odottaa vastausta. Asiantuntijan lukiessa hänelle lähetetyn lausuntopyynnön ensimmäistä kertaa lukemishetkesti otetaan aikaleima tilastointia ja käsittelytilanteen seuraamista varten.

## 7.12 Lausunto

Attribuutti	Arvojoukko	Kuvaus
kayt_id	viite	asiantuntijan tunniste
artik_id	viite	artikkelin tunniste
versionro	viite	version numero
annettu	aikaleima	lausunnon antamishetki
toimitukselle	teksti	toimitukselle menevä lausunto
kirjoittajalle	teksti	kirjoittajalle menevä lausunto
lukittu	boolean	uusintalausunnon lukitus
paatos	numero	hyväksytty (1), hyväksytty suurin muutoksin (2), hyväksytty pienin muutoksin (3), hylätty (4)

Lausunto koskee tiettyä artikkelin versiota, ja se identifioidaan lausuntopyynnön tavoin. Lausunnolla on neljä mahdollista tilaa: hyväksytty, hyväksytty pienin muutoksin, hyväksytty suurin muutoksin tai hylätty. Lausuntoon liittyy määrittelemättömän monta liitettä, jotka voivat olla joko kuvia tai tekstiä. Lausunnon antamishetkestä otetaan aikaleima tilastointia ja käsittelyn seuraamista varten.

## 7.13 Liite

Attribuutti	Arvojoukko	Kuvaus
kayt_id	viite	asiantuntijan tunniste
artik_id	viite	artikkelin tunniste
versionro	viite	version numero
liitenro	viite	liitteen numero
tiedostonimi	teksti	viittaus liitetiedostoon
luokitus	boolean	sekä kirjoittajalle että toimitukselle?

Liite liittyy yhteen lausuntoon, ja se identifioidaan artikkelin ja asiantuntijan tunnusten sekä versio- ja liitenumeron avulla. Asiantuntija määrittelee jokaiselle liitteelle näkyvyysluokituksen; osa liitteistä on tarkoitettu vain toimituksen käyttöön.



## 7.14 Päätös

Attribuutti	Arvojoukko	Kuvaus
artik_id	viite	artikkelin tunniste
kayt_id	viite	toimittajan tunniste
versionro	viite	versionumero
annettu	aikaleima	päätöksen antamishetki
paatos	numero	hyväksytty (0), hyväksytty suurin muutoksin (1), hyväksytty pienin muutoksin (2), hylätty (3)

## 7.15 Loki

Attribuutti	Arvojoukko	Kuvaus
kayt_id	viite	käyttäjän tunniste
aikaleima	aikaleima	kirjattavan toiminnon tapahtumishetki
syy	teksti	kirjattava toiminto

Loki-tauluun kirjataan sekä kirjoittajien että asiantuntijoiden toimintoja. Raporteissa useasti käytetyt aikaleimat pidetään kyselyiden helpottamiseksi niihin liittyvissä tauluissa, ja lokiin kirjataan kyseisten aikaleimojen lisäksi kaikki toimitusta mahdollisesti kiinnostavat tapahtumat. Tällaisia ovat esimerkiksi asiantuntijoiden sisäänkirjautumiset, lausuntopyyntöjen selailut ja pyyntöihin vastaamiset.

## 7.16 Järjestelmään liittyvien tiedostojen tallennus

Järjestelmässä tallennetaan tietokannan lisäksi dataa myös tiedostoihin. Jokaiselle järjestelmän asiantuntijalle luodaan hakemistoon *liitteet* alihakemisto, jonka nimi on *ID(kayt id)*.

Asiantuntijoiden lausuntoihin liittyvät liitetiedostot tallennetaan asiantuntijoiden henkilökohtaisiin hakemistoihin seuraavasti:

A(artik id)V(versionro)L(liitenro).(tiedostopäätte)

Siis esimerkiksi hakemistossa *liitteet/ID84/* olevasta tiedostosta *A110V3L2.pdf* voitaisiin päätellä, että se on asiantuntijan numero 84 PDF-muotoinen liite numero 2 ja se liittyy artikkelin numero 110 kolmannen version lausuntoon. Liitetiedoston muoto ei ole rajattu, ja se voi PDF-muodon lisäksi olla esimerkiksi JPEG, PS tai GIF.

Artikkelin eri versiot tallennetaan tiedostoon *versiot/kayt id/* seuraavassa muodossa:

A(artik id)V(versionro).(tiedostopääte)

Esimerkiksi *versiot/ID56/A2341V4.ps* tarkoittaa artikkelin numero 2341 neljättä PS-muotoista versiota, jonka on kirjoittanut käyttäjä numero 56.

Kun artikkelien versio tai liitetiedosto kopioidaan palvelimeen, tiedostot tallennetaan väliaikaisesti hakemistoon *temp* edellä mainitun nimisinä. Tiedostojen siirtäminen oikeisiin hakemistoihin tapahtuu lähetyispainikkeen painamisen yhteydessä. Jos lähetyispainiketta ei paineta, tyhjennetään temp-hakemisto uloskirjautumisen yhteydessä.

Erillinen lähetyispainike tarvitaan, jotta asiantuntija voi ladata useita liitetiedostoja. Lähetyispainike on myös tarpeellinen silloin, kun hitaan Internet-yhteyden takia tiedoston lataaminen palvelimelle kestää kauan.

## 8 Käyttöliittymä

Järjestelmällä on kolme käyttäjäryhmää: kirjoittajat, asiantuntijat ja toimitus, ja siten myös kolme erillistä käyttöliittymää. Jokainen asiantuntija on myös kirjoittaja, ja jokainen toimittaja on myös asiantuntija sekä kirjoittaja, joten asiantuntijalle tarjotaan myös kirjoittajan käyttöliittymä, ja toimittajalle asiantuntijan ja kirjoittajan käyttöliittymät.

Järjestelmän käyttöliittymähahmotelmat on kuvina liitteessä 10.4.

### 8.1 Kaikki käyttäjäryhmät

#### 8.1.1 Kirjautuminen

Järjestelmään rekisteröitynyt käyttäjä syöttää sähköpostiosoitteensa ja salasanansa niille varattuihin kenttiin. Painamalla ”Sign in” -painiketta käyttäjä kirjautuu sisään järjestelmään. Käyttäjän ollessa 1) kirjoittaja, hän siirtyy ”omat artikkelit” -näkömään, 2) asiantuntija, hän siirtyy ”artikkelien arvostelu” -näkömään, 3) toimittaja, hän siirtyy ”artikkelien hallinta” -näkömään tai 4) päätoimittaja, hän siirtyy ”päätoimittaja” -näkömään. Näkömään vasemmassa reunassa on kaikilla käyttäjillä valikko, jonka avulla siirrytään käyttäjän eri näkömien välillä.

Uusi käyttäjä voi rekisteröityä järjestelmän kirjoittajaksi ”Register new profile” -painiketta painamalla, jolloin hän siirtyy ”rekisteröityminen” -näkömään. Uusien asiantuntijoiden ja toimittajien rekisteröinnin hoitaa lehden toimitus.

#### Virheen käsittely

- Jos sähköpostiosoitetta ei löydy tietokannasta, tai jos salasana ei vastaa sähköpostiosoitteen salasanaa, annetaan virheilmoitus ”Login failed: wrong email or password.”

## 8.2 Kirjoittaja

Kirjoittajan näkymän vasemman reunan valikossa on linkit näkyviin ”omat artikkelit” ja ”profiilin muokkaus” sekä linkki ”uloskirjautuminen”.

### 8.2.1 Rekisteröityminen

Käyttäjä syöttää kenttiin etunimen, sukunimen, titelin, katuosoitteen, kaupungin, postinumeron, osavaltion, maan, puhelinnumeron, sähköpostiosoitteen sekä salasanan kahteen kertaan niille varattuihin kenttiin. Painamalla ”Create profile” -painiketta hän rekisteröityy kirjoittajaksi ja sisäänkirjautuu automaattisesti.

#### Virheenkäsittely

- Jos etunimi, sukunimi, sähköpostiosoite tai salasana on tyhjä, annetaan virheilmoitus ”You must specify first name / last name / e-mail / password”.
- Jos sähköpostiosoite ei ole oikeaa muotoa (@-merkki puuttuu), annetaan virheilmoitus ”Must have @-symbol in e-mail address”.
- Jos sähköpostiosoite löytyy jo tietokannasta, annetaan virheilmoitus ”E-mail address already registered”.
- Jos syötetty salasana ei ole tarpeeksi pitkä, annetaan virheilmoitus ”Password must be at least 5 characters”.
- Jos syötetty salasana ei vastaa uudelleensyötettyä salasanaa, annetaan virheilmoitus ”Passwords don’t match”.

### 8.2.2 Omat artikkelit

Näkymän yläreunassa on lueteltu kaikki kyseisen kirjoittajan järjestelmään lähettämät artikkelit. Luettelossa näkyy artikkelin otsikko, aihe, saapumispäivämäärä ja tila.

Artikkeliluettelon alapuolella on lomake, jolla kirjoittaja voi lähettää uuden artikkelin. Kirjoittaja syöttää artikkelin otsikon, aiheen, tiivistelmän ja artikkelin PS/PDF-muodossa niille varattuihin kenttiin ja painamalla ”Send article” -painiketta lähettää artikkelin. Uusi artikkeli siirtyy näkymän artikkeliluettelo, ja käyttäjä saa kuittauksen lähetyksen onnistumisesta.

Lausuntokierrokselta valmistuneen artikkelin otsikkoa painamalla kirjoittaja voi katsoa millaista palautetta artikkeli on saanut. Palautteisiin kuuluvat tekstit ja liitetiedostot luetellaan asiantuntijoin (anonyymisti), ja tiedoston nimeä painamalla kirjoittaja voi avata kyseisen tiedoston. Jos artikkeli on hyväksytty pienin tai suurin muutoksin tai siitä on pyydetty uusi versio, kirjoittaja voi lähettää korjatun version artikkelista palautteen alapuolella olevalla lomakkeella. Kirjoittaja voi muokata artikkelin otsikkoa ja tiivistelmää niille varatuissa kentissä sekä syöttää artikkelin korjatun version PS/PDF-muodossa. Kirjoittajan painettua ”Send article” -painiketta korjattu versio siirtyy näkymän artikkeliluetteloon, ja kirjoittaja saa kiittauksen lähetyksen onnistumisesta. Jos artikkeli on hyväksytty ja siitä on pyydetty julkaistava versio, kirjoittaja voi syöttää artikkelin L<sup>A</sup>T<sub>E</sub>X-muodossa sille varattuun kenttään ja lähettää sen ”Send article” -painiketta painamalla. Artikkelin tila muuttuu näkymän artikkeliluettelossa, ja kirjoittaja saa kiittauksen lähetyksen onnistumisesta.

### Virheenkäsittely

- Jos tiedostonlataamiskenttään on syötetty virheellinen polku, annetaan virheilmoitus ”Unknown path”.

### 8.2.3 Profilin muokkaus

Kirjoittaja voi muokata seuraavia tietojaan: etunimeä, sukunimeä, titteliä, katuosoitetta, kaupunkia, postinumeroa, osavaltiota, maata, puhelinnumeroa, sähköpostiosoitteita, aktiivista sähköpostiosoitetta ja salasanaansa niitä vastaavilla kentillä. ”Update profile” -painiketta painamalla muutokset tulevat voimaan, ja kirjoittaja siirtyy takaisin ”omat artikkelit” -näkyeseen.

### Virheenkäsittely

- Jos etunimi, sukunimi, sähköpostiosoite tai salasana on tyhjä, annetaan virheilmoitus ”You must specify first name / last name / e-mail address / password”.
- Jos lisättävä sähköpostiosoite on väärää muotoa (@-merkki puuttuu), annetaan virheilmoitus ”Must have @-symbol in e-mail address”.
- Jos lisättävä sähköpostiosoite löytyy jo tietokannasta, annetaan virheilmoitus ”E-mail address already registered”.
- Jos käyttäjän ainoa sähköpostiosoite yritetään poistaa, annetaan virheilmoitus ”You must have at least one e-mail address”.
- Jos syötetty salasana ei ole tarpeeksi pitkä, annetaan virheilmoitus ”Password must be at least 5 characters”.
- Jos syötetty salasana ei vastaa uudelleensyötettyä salasanaa, annetaan virheilmoitus ”Passwords don’t match”.

## 8.3 Asiantuntija

Asiantuntijan näkymän vasemman reunan valikossa on linkit näkyisiin ”artikkelien arvostelu” ja ”profiilin muokkaus” sekä linkki ”uloskirjautuminen”.

### 8.3.1 Artikkelien arvostelu

Näkymän yläreunassa on lueteltu kaikki artikkelit, joihin kyseinen asiantuntija on valittu lausunnonantajaksi. Oletuksena luettelosta on valittu usein saapunut artikkeli. Luettelossa näkyy artikkelien otsikko, kirjoittaja, saapumispäivämäärä, aihe sekä tila. Artikkelin otsikkoa painamalla asiantuntija voi tutkia artikkelin tiivistelmää ja itse artikkelia.

Asiantuntija voi hyväksyä uuden lausuntopyynnön ensin valittuaan kyseisen artikkelin (artikkelin tila: ”New article”) ja sitten painamalla ”Yes”-painiketta, jolloin artikkelin tila muuttuu (”Not yet reviewed”), tai hylätä pyynnön painamalla ”No”-painiketta, jolloin artikkeli poistuu asiantuntijan näkymän artikkeliluettelosta.

Asiantuntija voi arvostella artikkelin ensin valittuaan kyseisen artikkelin ja sitten valitsemalla sopivan vaihtoehdoista ”Accepted”, ”Accepted with minor revision”, ”Accepted with major revision” tai ”Rejected” sekä lisäämällä toimitukselle ja kirjoittajalle omat palautteensa niille varattuihin tekstilaatikoihin sekä palautteen liitteet tiedostoina niille varattuihin kenttiin ja painamalla ”Send review” -painiketta. Artikkelin tila päivittyy näkymän artikkeliluetteloon, ja asiantuntija saa kiittauksen palautteen lähetyksen onnistumisesta. Asiantuntija voi muokata jo antamaansa palautetta, jos toimitus on pyytänyt lausunnon uudelleenlähetyksen (esim. asiantuntijan pyynnöstä).

#### Virheenkäsittely

- Jos tiedostonlataamiskenttään on syötetty virheellinen polku, annetaan virheilmoitus ”Unknown path”.

### 8.3.2 Profiilin muokkaus

Asiantuntija voi muokata seuraavia tietojaan: etunimeä, sukunimeä, titteliä, katuosoitetta, kaupunkia, postinumeroa, osavaltiota, maata, laitosta, puhelinnumeroa, sähköpostiosoitteita, aktiivista sähköpostiosoitetta, erikoisaloja, vapaamuotoista kuvausta itsestään ja salasanaansa niitä vastaavilla kentillä. Asiantuntija voi myös vaihtaa statustaan (aktiivinen/passiivinen). ”Update profile” -painiketta painamalla muutokset tulevat voimaan, ja asiantuntija siirtyy takaisin ”artikkelien arvostelu” -näkökseen.

#### Virheenkäsittely

- Jos etunimi, sukunimi, sähköpostiosoite tai salasana on tyhjä, annetaan virheilmoitus ”You must specify first name / last name / e-mail address / password”.

- Jos lisättävä sähköpostiosoite on väärää muotoa (@-merkki puuttuu), annetaan virheilmoitus ”Must have @-symbol in e-mail address”.
- Jos lisättävä sähköpostiosoite löytyy jo tietokannasta, annetaan virheilmoitus ”E-mail address already registered”.
- Jos käyttäjän ainoa sähköpostiosoite yritetään poistaa, annetaan virheilmoitus ”You must have at least one e-mail address”.
- Jos syötetty salasana ei ole tarpeeksi pitkä, annetaan virheilmoitus ”Password must be at least 5 characters”.
- Jos syötetty salasana ei vastaa uudelleensyötettyä salasanaa, annetaan virheilmoitus ”Passwords don’t match”.

## 8.4 Toimitus

Toimittajan näkymän vasemman reunan valikossa on linkit näkymiin ”artikkelien hallinta”, ”lehden tila”, ”käyttäjähallinta”, ”raportit” ja ”profiilin muokkaus” sekä linkki ”uloskirjautuminen”. Päätoimittajalla on näiden lisäksi linkki näkymään ”päätoimittajanäkymä”.

### 8.4.1 Artikkelien hallinta

Näkymän yläreunassa on lueteltu kaikki järjestelmän artikkelit, jotka vaativat toimituksen toimia. Tällaisia artikkeleita ovat 1) uudet artikkelit, ts. artikkelit, joihin ei ole vielä valittu asiantuntijoita arvostelijoiksi, 2) artikkelit, jotka ovat juuri palanneet päätoimittajakierrokselta, 3) artikkelit, joiden kaikki lausunnot ovat tulleet, mutta toimitus ei ole vielä antanut lopullista päätöstä. Oletuksena luettelosta on valittu uusin saapunut artikkeli. Luettelossa näkyy artikkelien otsikko, kirjoittaja, aihe, saapumispäivämäärä sekä tila asiantuntijoittain. Luettelon yläpuolella olevaa ”Show all articles” -linkkiä painamalla toimittaja saa artikkeliluetteloon kaikki järjestelmän artikkelit. Artikkelin otsikkoa artikkeliluettelosta painamalla toimittaja pääsee käsittelemään kyseistä artikkelia. Toimittaja voi tutustua artikkelin tekstilaatikossa näkyvään tiivistelmään ja koko artikkeliin ”Show article” -painiketta painamalla, pyytää uudelleenlähetystä ”Request resending article” -painikkeella, hylätä artikkelin ”Reject article” -painikkella tai poistaa artikkelin järjestelmästä ”Delete article” -painikkeella.

Toimittaja voi valita uudelle artikkelille asiantuntijoita arvostelijoiksi. Artikkelin tietojen alapuolella on luettelo, jossa on lueteltu kaikki järjestelmässä olevat asiantuntijat sekä heidän erikoisalansa. Tästä luettelosta toimittaja voi lisätä asiantuntijoita arvostelemaan valittua artikkelia ”Add”-painiketta halutun asiantuntijan kohdalla painamalla, jolloin valittu asiantuntija siirtyy kyseisen artikkelin arvostelijoiden luetteloon. Arvostelijoita voi poistaa kyseiseltä artikkelilta ”Remove”-painiketta painamalla. Toimittaja voi hakea toimittajia hakusanalla kirjoittamalla hakusanan ”Search referee” -kenttään ja painamalla

”Search referee” -painiketta. Toimittaja voi lisätä järjestelmään uuden asiantuntijan painamalla ”Add new referee” -linkkiä, jolloin hänelle avautuu uusi ikkuna, jossa toimittaja täyttää asiantuntijan etunimen, sukunimen, erikoisalat ja sähköpostiosoitteen niille varattuihin kenttiin ja painaa ”Add referee” -painiketta. ”Notify selected” -painiketta painamalla artikkelin lausuntopyynnöt lähtevät valituille asiantuntijoille ja artikkelin tila päivittyy näkymän artikkeliluetteloon. Toimittaja voi lähettää viestin päätoimittajalle valittua artikkelia koskien kirjoittamalla viestin sille varattuun kenttään ja painamalla ”Send message” -painiketta.

Lausuntokierrokselta valmistuneen artikkelin saamat palautteet ja niiden liitetiedostot on lueteltu asiantuntijoittain ryhmiteltynä. Toimittaja voi muokata kirjoittajalle tarkoitettuja palautteita, avata liitetiedostoja tiedostonimiä painamalla sekä valitsemalla valintaruuduilla mitkä liitetiedostot lähetetään kirjoittajalle. Toimittaja voi pyytää lausunnotta uudelleenlähetyksen painamalla ”Request resending review” -painiketta. Toimittaja voi asiantuntijoiden palautteiden perusteella valita sopivan vaihtoehdoista ”Accepted”, ”Accepted with minor revision”, ”Accepted with major revision” tai ”Rejected”. ”Save and send information to writer” -painiketta painamalla artikkelin tila päivittyy näkymän artikkeliluetteloon ja toimittaja saa kiittauksen lähetyksen onnistumisesta.

Toimittaja voi navigoida artikkelin ”lausunto”- ja ”asiantuntijoiden valinta” -näkyvien välillä painamalla linkkejä ”View reviews” ja ”View article organizing”.

### Virheenkäsittely

- Jos lisättävän asiantuntijan etunimi, sukunimi tai sähköpostiosoite on tyhjä, annetaan virheilmoitus ”You must specify referee’s first name / last name / e-mail address”.
- Jos lisättävän asiantuntija sähköpostiosoite on väärää muotoa (@-merkki puuttuu), annetaan virheilmoitus ”Must have @-symbol in e-mail address”.
- Jos lisättävän asiantuntijan sähköpostiosoite löytyy jo tietokannasta, annetaan virheilmoitus ”E-mail address already registered”.

### 8.4.2 Lehden tila

Näkymän yläreunassa on lueteltu järjestelmän artikkelit, joita ei ole vielä sijoitettu mihinkään lehteen, ja alareunassa lehden valittuun numeroon valitut artikkelit. Toimittaja voi siirtyä lehden muihin numeroihin artikkeliluettelon alapuolella olevasta lehtiluettelosta halutun lehden vuosikerran ja numero valitsemalla. Artikkeleista näkyy luettelossa otsikko, kirjoittaja, aihe, saapumispäivämäärä sekä tila. Artikkelin otsikkoa artikkeliluettelossa painettuaan toimittaja voi tutustua artikkelin tiivistelmään ”Show abstract” -painiketta painamalla, koko artikkeliin ”Show article” -painikkeella, artikkeliin  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -muodossa ”Show Latex” -painikkeella tai lähettää artikkelin lausunnonantajille muistutuksen ”Send reminder” -painiketta painamalla halutun asiantuntijan nimen kohdalla.

”Send reminder” -painikkeiden oikealla puolella on kenttä, jossa näkyy viimeisin ajankohta, jolloin kyseiselle asiantuntijalle on lähetetty muistutus kyseisen artikkelin arvostelusta. ”Publish this article” -painiketta painamalla valittu artikkeli siirtyy valittuun lehden numeroon julkaistavaksi. Lehden numeroon sijoitetun artikkelin voi poistaa kyseisestä numerosta painamalla artikkelin kohdalla ”Remove from selected” -painiketta.

### 8.4.3 Käyttäjähallinta

Näkymän yläreunassa on lueteltu kaikki järjestelmän käyttäjät. Luettelossa näkyy käyttäjän nimi, rooli (kirjoittaja/asiantuntija/toimittaja/päätoimittaja) ja sähköpostiosoite. Käyttäjän nimeä painamalla toimittaja pääsee muokkaamaan kyseisen käyttäjän tietoja käyttäjäluektion alapuolella olevalla lomakkeella. Lomakkeessa on kentät etunimelle, sukunimelle, tittelille, katuosoittele, kaupungille, postinumerolle, osavaltiolle, maalle, laitokselle, puhelinnumerolle, sähköpostiosoittele, roolille, erikoisaloille sekä vapaalle kuvaukselle valitun käyttäjän roolin mukaan. Toimittaja voi lisätä järjestelmään ja valitulle asiantuntijalle uuden alan täyttämällä ”Add new expertise” -kentän ja painamalla ”Add” -painiketta. ”Update fields” -painiketta painamalla kyseisen käyttäjän päivitetty tiedot astuvat voimaan ja päivittyvät käyttäjäluektion. Toimittaja saa kiittauksen tietojen päivituksen onnistumisesta. Toimittaja voi lähettää käyttäjälle uuden järjestelmän generoiman salasanan painamalla ”Generate and send new password” -painiketta.

Toimittaja voi lisätä järjestelmään uuden toimittajan näkymän alareunan lomakkeella täyttämällä uuden toimittajan etunimen, sukunimen ja sähköpostiosoitteen vastaaviin kenttiin ja painamalla ”Add editor” -painiketta.

Toimittaja voi lisätä järjestelmään uuden asiantuntijan näkymän alareunan lomakkeella täyttämällä uuden asiantuntijan etunimen, sukunimen, erikoisalat ja sähköpostiosoitteen vastaaviin kenttiin ja painamalla ”Add referee” -painiketta.

Toimittaja voi muokata järjestelmän erikoisaloja näkymän alareunassa. Erikoisalat on lueteltu aakkosjärjestyksessä, ja niitä voi poistaa sekä muokata vastaavilla painikkeilla. Uusia erikoisaloja voi lisätä ”Add new expertise” -painikkeella.

### Virheenkäsittely

- Jos muokattu etunimi tai sukunimi on tyhjä, annetaan virheilmoitus ”You must specify user’s first name / last name”.
- Jos lisättävä sähköpostiosoite on väärää muotoa (@-merkki puuttuu), annetaan virheilmoitus ”Must have @-symbol in e-mail address”.
- Jos lisättävä sähköpostiosoite löytyy jo tietokannasta, annetaan virheilmoitus ”E-mail address already registered”.
- Jos käyttäjän ainoa sähköpostiosoite yritetään poistaa, annetaan virheilmoitus ”User must have at least one e-mail address”.



- Jos yritetään antaa päätoimittaja-status useammalle kuin yhdelle toimittajalle, annetaan virheilmoitus ”Editor in chief already defined: name”.
- Jos lisättävän toimittajan etunimi, sukunimi tai sähköpostiosoite on tyhjä, annetaan virheilmoitus ”You must specify editor’s first name / last name / e-mail address”.
- Jos lisättävän toimittajan sähköpostiosoite löytyy jo tietokannasta, annetaan virheilmoitus ”E-mail address already registered”.
- Jos lisättävä ala löytyy jo tietokannasta, annetaan virheilmoitus ”Expertise already defined.”.

#### **8.4.4 Raportit**

Näkymässä näkyy järjestelmän tilastotietoja, joihin liittyviä attribuutteja on esitelty luvussa 9.

#### **8.4.5 Profilin muokkaus**

Toimittaja voi muokata seuraavia tietojaan: etunimeä, sukunimeä, titteliä, katuosoitetta, kaupunkia, postinumeroa, osavaltiota, maata, laitosta, puhelinnumeroa, sähköpostiosoitteita, aktiivista sähköpostiosoitetta, erikoisaloja, vapaamuotoista kuvausta itsestään ja salasanaan niitä vastaavilla kentillä. Toimittaja voi myös vaihtaa asiantuntija-statustaan (aktiivinen/passiivinen). ”Update profile” -painiketta painamalla muutokset tulevat voimaan, ja toimittaja siirtyy takaisin näkymään, jossa oli ennen profilin muokkausta.

#### **Virheenkäsittely**

- Jos etunimi, sukunimi, sähköpostiosoite tai salasana on tyhjä, annetaan virheilmoitus ”You must specify first name / last name / e-mail address / password”.
- Jos muokattu sähköpostiosoite on väärää muotoa (@-merkki puuttuu), annetaan virheilmoitus ”Must have @-symbol in e-mail address”.
- Jos muokattu sähköpostiosoite löytyy jo tietokannasta, annetaan virheilmoitus ”E-mail address already registered”.
- Jos syötetty salasana ei ole tarpeeksi pitkä, annetaan virheilmoitus ”Password must be at least 5 characters”.
- Jos syötetty salasana ei vastaa uudelleensyötettyä salasanaa, annetaan virheilmoitus ”Passwords don’t match”.

### 8.4.6 Päätoimittajanäkymä

Päätoimittajanäkymän yläreunassa on lueteltu artikkelit, joita koskien muu toimitus on lähettänyt viestejä päätoimittajalle. Luettelon yläpuolella olevaa ”Show all articles” -linkkiä painamalla päätoimittaja saa artikkeliluetteloon kaikki järjestelmän artikkelit. Artikkelin otsikkoa artikkeliluettelosta painamalla päätoimittaja näkee mahdollisen toimituksesta tulleen viestin artikkelia koskien sekä käsittelemään artikkelia samanlaisen näkymän kautta kuin muutkin toimittajat. Päätoimittaja voi vastata toimituksen viestiin kirjoittamalla viestinsä samaan kenttään ja painamalla ”Reply”-painiketta.

Päätoimittajanäkymän vasemman reunan valikko on identtinen tavallisen toimittajan valikon kanssa ”päätoimittajanäkymä”-linkkiä lukuunottamatta, joten päätoimittajalle tarjotaan samat toiminnallisuudet kuin muillekin toimittajille.

## 9 Järjestelmästä saatavat raportit

Lehden toimituksella on mahdollisuus tulostaa erilaisia raportteja artikkeleihin ja asiantuntijoihin liittyen. Seuraavassa luetellaan raporttien kannalta oleelliset attribuutit ja niiden merkitykset sekä tieto jonka mukaan kyseinen attribuutti voidaan järjestää. Asiantuntijoiden tietoja voidaan tulostaa seuraavasti:

Attribuutti	Merkitys	Järjestäminen
Sukunimi	Asiantuntijan sukunimi	Aakkosjärjestys
Etunimi	Asiantuntijan etunimi	(Aakkosjärjestys)
Ala	Asiantuntijan erikoistumisalat	Aakkosjärjestys
Arvioitavaa	Asiantuntijalla on yhden tai useamman artikkelin arviointi kesken	kesken/ei kesken
Kieltäytymisiä	Kuinka monta kieltäytymistä artikkelin tarkastuksesta/tarkastuspyynnöt	Kieltäytymisten lkm
Lausunnon kesto	Kuinka kauan lausunnon antaminen on keskimäärin kestänyt	Keston mukaan
Arviot	Kuinka monta arviota kussakin ryhmässä: hyväksytty / hyväksytty pienin muutoksin / hyväksytty suurin muutoksin / hylätty /	Ei voida järjestää

Lisäksi seuraavista attribuuteista on ”yhteensä”-kenttä: arvioitavaa, kieltäytymisiä, lausunnon kesto ja arviot. Artikkelien tietoja voidaan rajata tietyille aikavälille ja tulostaa seuraavasti:

Attribuutti	Merkitys	Järjestäminen
Nimi	Artikkelin nimi	Aakkosjärjestys
Tila	Artikkelin tila: hylätty, vastaanotettu, 1. kierros, 2. kierros, ..., hyväksytty muutoksin, hyväksytty, julkaistu. Jokaisessa päivämäärä.	Tilan ja tilan sisällä ajan mukaisessa järjestyksessä

Lisäksi seuraavista tiloista on ”yhteensä”-kenttä: hylätty, vastaanotettu, 1.kierros, 2.kierros, ..., hyväksytty, julkaistu.

## 10 Erityiset tekniset ratkaisut

### 10.1 Salasanan tallennus

Kaikki salasanat ovat tietokannassa Salaja-luokan avulla salatussa muodossa. Turvallisuussyistä salasanoja käsitellään selväkielisenä vain, kun salasana lähetetään kirjautumissivulta järjestelmään. Kirjautumissivulta saatu salasana salataan Salaja-luokan avulla ja verrataan salattua versiota tunnusta vastaavaan salattuun arvoon tietokannassa. Jos salasana oli oikein, luodaan käyttäjää varten istuntotunnus (sessionID), joka tallennetaan istuntomuuttujiin. Käyttöoikeuksia tarkistettaessa verrataan tunnus+istuntotunnus-yhdistelmää vastaavaan tauluun tietokannassa.

### 10.2 Sähköpostin lähetys

Sähköpostin lähetykseen käytämme Javan standardilaajennusta JavaMail API 1.3.1.<sup>3</sup> JavaMail-kirjasto käyttää hyväkseen JavaBeans Activation Framework -kirjastoa. Molemmat löytyvät Sunin sivuilta<sup>4</sup> ja tulevat myös JBuilder 9:n mukana. JBuilderissa nämä kaksi kirjastoa saadaan käyttöön projektin luonnin yhteydessä valitsemalla ne tarvittaviksi kirjastoiksi (Required Libraries). Sunin sivuilta löytyy myös artikkeli komentokirjaston (Tag Library) käytöstä JavaMailin kanssa<sup>5</sup>.

### 10.3 Tiedostojen käsittely

Binääritiedostot, joille halutaan rajattu näkyvyys, säilytetään aina hakemistossa, johon on oikeus vain njc2-tunnuksen haltijalla. Näin varmistetaan, etteivät ulkopuoliset pääse lukemaan tiedostoja esimerkiksi URL-osoitetta arvaamalla.

<sup>3</sup><http://java.sun.com/products/javamail/javadoc/index.html>

<sup>4</sup><http://java.sun.com/products/javamail/>

<sup>5</sup><http://java.sun.com/developer/technicalArticles/javaserverpages/emailapps/>

Tiedostot näytetään käyttäjälle TiedostoServlet-luokan avulla. Näin voidaan pitää huoli siitä, että käyttäjällä on oikeus lukea kyseinen tiedosto, ja että näyttökertoja voidaan tarvittaessa rajata.

Myös järjestelmään lähetetyt tiedostot tallennetaan hakemistoon, johon ei ole suoraa pääsyä Internetistä.

## **10.4 JSP-sivujen header**

Jokaisen JSP-sivun alkuun importataan header.jsp. Tiedosto saa parametrina ryhmä, jonka arvo voi olla joko 1, 2, 3 tai 4. Tiedosto tulostaa HTML-sivun tarvittavat aloitustagit, esimerkiksi viitteen tyylitiedostoon. Lisäksi tiedostossa tarkistetaan session voimassaolo katsomalla löytyykö sessiosta SessionData-olio. Jos ei, niin ohjataan login-sivulle virheilmoituksen kera.

## **Liite 1. Event-luokan vakiot**

Event.AVAA\_ARTIKKELI = 1  
Event.HAE\_ASiantuntija = 2  
Event.HYLKAA\_ARTIKKELI = 3  
Event.HYLKAA\_ARVOSTELUPYYNTO = 4  
Event.HYVAKSY\_ARVOSTELTAVAKSI = 5  
Event.JARJESTA\_EMAIL = 6  
Event.JARJESTA\_EXPERTISE = 7  
Event.JARJESTA\_NAME = 8  
Event.JARJESTA\_STATE = 9  
Event.JARJESTA\_SUBMIT = 10  
Event.JARJESTA\_TITLE = 11  
Event.JARJESTA\_TOPIC = 12  
Event.JARJESTA\_ROLE = 13  
Event.JARJESTA\_WRITER = 14  
Event.JULKAISE\_ARTIKKELI = 15  
Event.KIRJAUDU = 16  
Event.KIRJAUDU\_ULOS = 17  
Event.LAHETA\_ARTIKKELI = 18  
Event.LAHETA\_LATEX = 19  
Event.LAHETA\_MUISTUTUS = 20  
Event.LAHETA\_PALAUTE = 21  
Event.LAHETA\_TIEDOTUS = 22  
Event.LATAA = 23  
Event.LATAA\_LIITE = 24  
Event.LAUSUNNON\_UUDELLEENPYYNTO = 25  
Event.LISAA\_ASiantuntija\_ARTIKKELIIN = 26  
Event.LISAA\_SPOSTI = 27  
Event.LISAA\_TOIMITTAJA = 28  
Event.LISAA\_OMA\_ALA = 29  
Event.LISAA\_UUSI\_ALA = 30  
Event.LISAA\_ASiantuntija = 31  
Event.LUO\_PAATOS = 32  
Event.MUOKKAA\_ALA = 33  
Event.NAYTA\_KAIKKI = 34  
Event.PAIVITA\_PROFILI = 35  
Event.POISTA\_ALA = 36  
Event.POISTA\_ARTIKKELI = 37  
Event.POISTA\_ASiantuntija\_ARTIKKELISTA = 38  
Event.POISTA\_LEHDESTA = 39  
Event.POISTA\_LIITE = 40  
Event.POISTA\_OMA\_ALA = 41  
Event.POISTA\_SPOSTI = 42  
Event.PYYDA\_UUDELLEENLAHETYS = 43

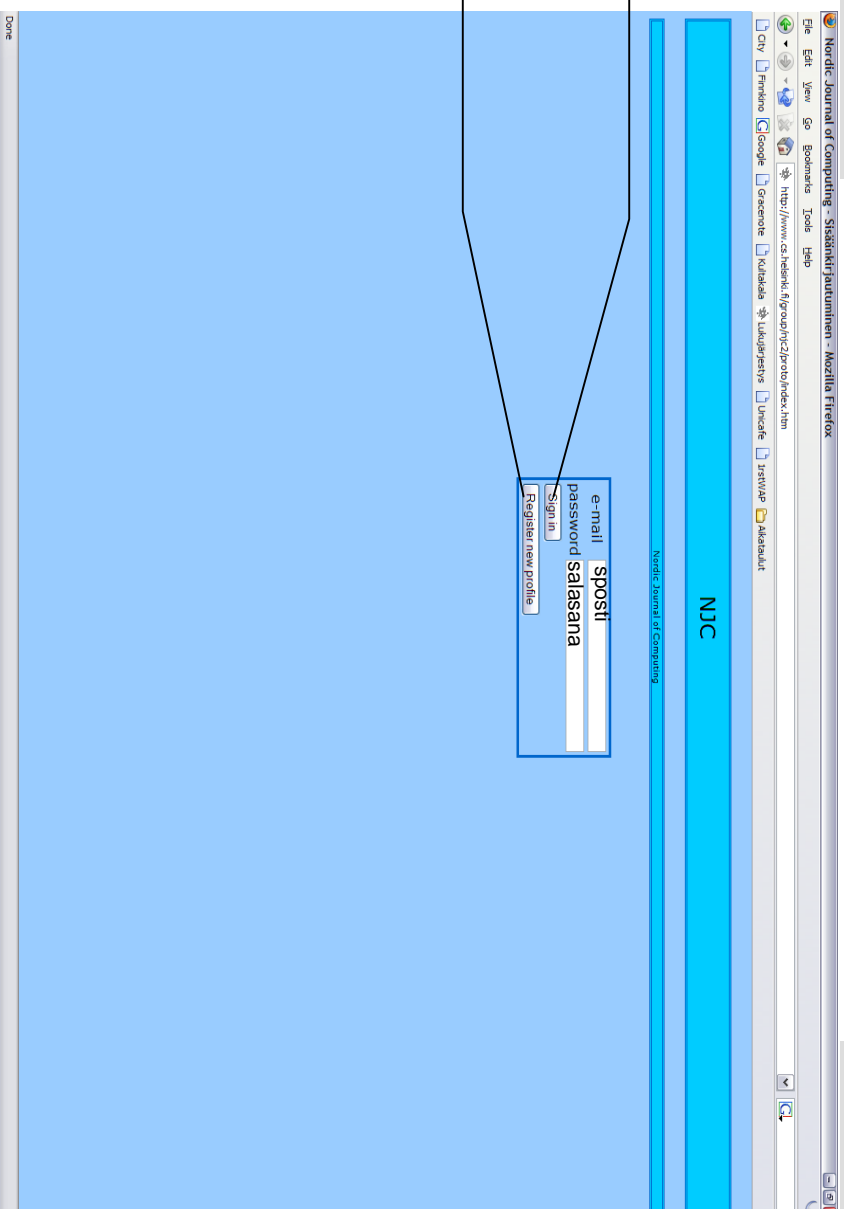
Event.REKISTEROIDY = 44  
Event.SIIRRY\_VALITTUUN\_LEHTEEN = 45  
Event.TALLENNA\_REKISTEROINTI = 46  
Event.TALLENNA\_VIESTI = 47

## **Liite 2. Käyttöliittymähahmotelmat**

Saatavat parametrin sposti (virheellisen kirjautumisen seurauksena)

Metodeita

SessionData tarkistaKirjautuminen(sposti, salasana)



Event.KIRJAUDU

Event.REKISTEROIDY



### Saatavat parametrit

enimi, snimi, titteili, katuosoite, kaupunki,postinro, osavaltio, maa, sposti, salasana, salasana2

Virheet:

enimi\_tark 1:puuttuu  
snimi\_tark 1:puuttuu  
sposti\_tark 1:puuttuu, 2:ei validi, 3:käytössä  
salasana\_tark 1:puuttuu, 2:ei täsmää, 3:pituus<5

### Metodeita

boolean tarkistaaSahkoposti(String sahkoposti)  
submittKaytajaratedot (Kaytajar, salasana )

Field	Value
First name	enimi
Last name	snimi
Title	titteili
Address	katuosoite
City	kaupunki
Zip code	postinro
State	osavaltio
Country	maa
E-mail	sposti
Password	salasana
Retype password	salasana2

Event: TALLENNA\_REKISTEROINTI

### Saatavat parametrit

jarjesta\_1:title, 2:topic, 3:submit  
artik\_id  
versiono  
otsikko  
ala  
tiivitelma

Event.JARJESTA\_TITLE

Event.JARJESTA\_TOPIC

### Metodetta

List getKirjoittajanArtikkelit(kayt\_id)  
submitArtikkeli(Artikkeli)  
tallennaTiedosto(nimi)

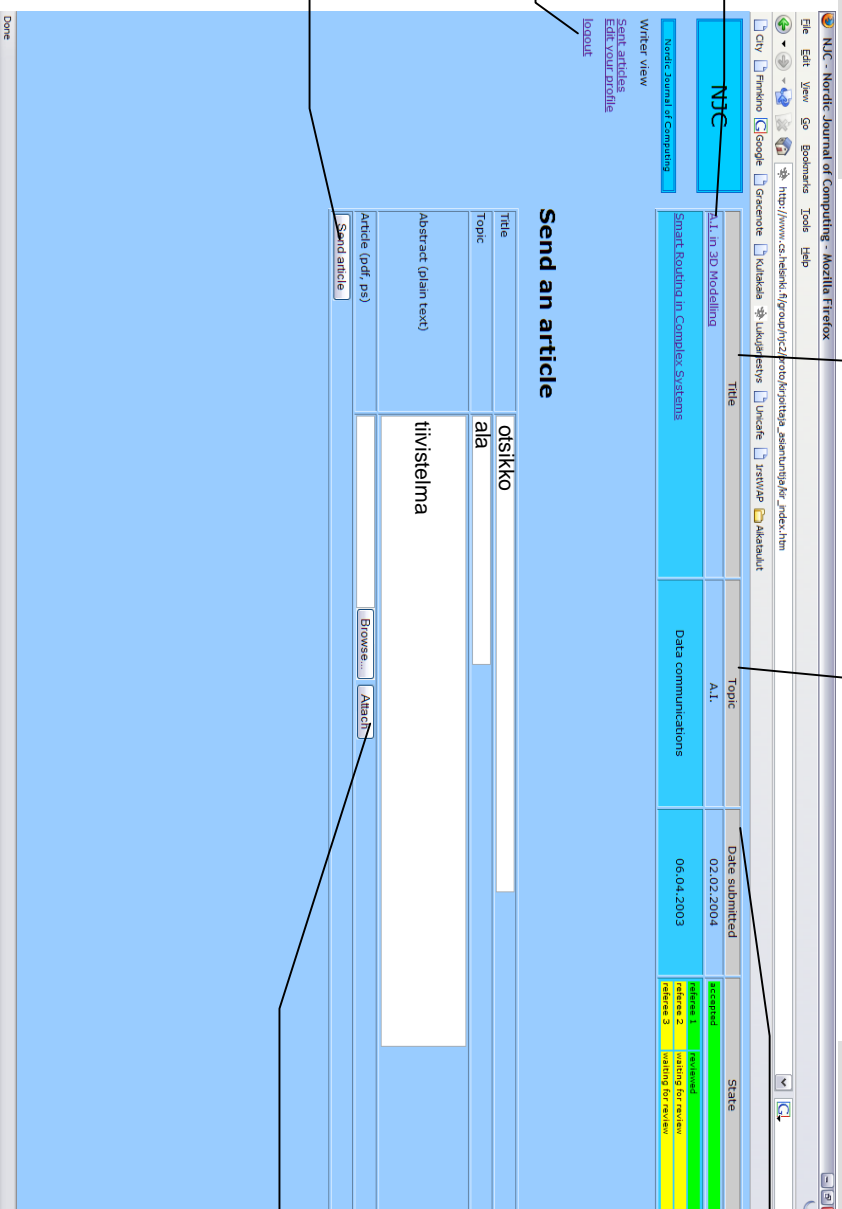
Event.JARJESTA\_SUBMIT

Event.AVAA\_ARTIKKELI  
kayt\_id  
artik\_id  
versiono

Event.KIRJAUDU\_ULOS

Event.LAHETA\_ARTIKKELI

Tiedoston nimeäminen:  
x/kayt\_id/artik\_id/versionoTiedosto  
Tarkistetaan:  
kenttien tyhjyys, tiedoston muoto ps/pdf



Event.LATAA

NJC - Nordic Journal of Computing - Mozilla Firefox

File Edit View Go Bookmarks Tools Help  
[http://www.cs.helsinki.fi/group/njc2/proto/finpictaja\\_asetukset/fin\\_index.htm](http://www.cs.helsinki.fi/group/njc2/proto/finpictaja_asetukset/fin_index.htm)

**NJC**  
 Nordic Journal of Computing

[Sent articles](#)  
[Edit your profile](#)  
[logout](#)

[Writer view](#)  
[Smart articles](#)  
[Smart articles](#)  
[Smart articles](#)

Title	Topic	Date submitted	State
<a href="#">A.I. in 3D Modelling</a>	A.I.	2.2.2004	accepted
<a href="#">Uusikas menit</a>	A.I.	5.1.2004	resand requested
<a href="#">Smart Routing in Complex Systems</a>	Data communications	6.4.2003	referees 1 reviewed referees 2 waiting for review referees 3 waiting for review

**State of review**

Title	Topic	Date submitted
<a href="#">Smart Routing in Complex Systems</a>	A.I.	2.2.2004

referee 1	reviewed
referee 2	waiting for review
referee 3	waiting for review

Abstract:  
 Tässä on artikkelin tiivistelmä.

[Show article](#)

Done

The screenshot shows the NJC website interface. At the top, there is a navigation menu with 'File', 'Edit', 'View', 'Go', 'Bookmarks', 'Tools', and 'Help'. The address bar shows the URL: [http://www.cs.helsinki.fi/group/nc2/prod/ohjeltoista\\_sisartuntable/index.htm](http://www.cs.helsinki.fi/group/nc2/prod/ohjeltoista_sisartuntable/index.htm). The main content area features a table of articles:

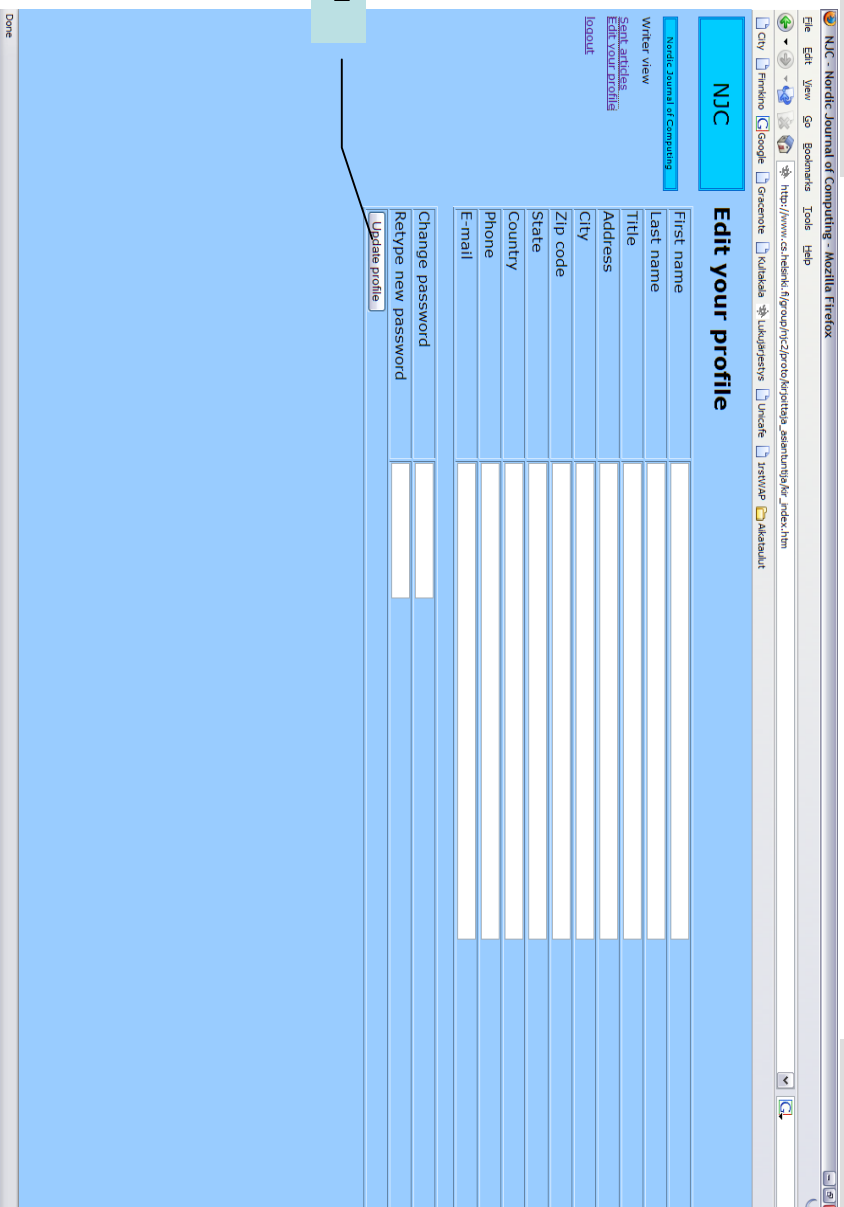
Title	Topic	Date submitted	State
A.I. in 3D Modelling	A.I.	2.2.2004	accepted
Uusien men	A.I.	5.1.2004	resend requested
Smart Routing in Complex Systems	Data communications	6.4.2003	reference 1 reviewed reference 2 waiting for review reference 3 waiting for review

Below the table, there is a section titled 'Feedback for the article' for the article 'A.I. in 3D Modelling'. It includes a 'Title' field with 'A.I.' and a 'Date submitted' field with '2.2.2004'. A green bar indicates 'Article accepted'. There are two 'Feedback from referees' sections:

- referee 1**: [file.pdf](#)  
Tässä on palaute ensimmäiseltä arvostelijalta.
- referee 2**: [files5\\_files4](#)  
Tässä on palaute toiselta arvostelijalta.

At the bottom, there is a 'Send article in Latex form' button. A callout box points to this button with the text 'Event: LAHETA\_LATEX'.

*Katso Create new profile*



Event: TALLENNA\_REKISTEROINTI

SessionData submitKayttajatiedot

- id hidden-kenttänä

NJC - Nordic Journal of Computing - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

Http://www.cs.helsinki.fi/group/njc2/prood/krp/krp/ta\_ssa/vuorokkaiw\_index.htm

**NJC**

Nordic Journal of Computing

Writer view

Send articles  
Edit your profile  
logout

Title	Topic	Date submitted	State
A.I. in 3D Modelling	A.I.	2.2.2004	accepted
Juustik's menü	A.I.	5.1.2004	resend requested
Smart Routing in Complex Systems	Data communications	6.4.2003	referee 1 reviewed referee 2 waiting for review referee 3 waiting for review

**Resend an article**

Uusiksi menü      A.I.      Topic      5.1.2004      Date submitted

**Resend requested**

Title

Topic

Abstract (plain text)

Article (pdf, ps)      Browse...      Attach

Send article

Done

The screenshot shows the NJC website interface. At the top, there is a navigation menu with links like 'Home', 'About', 'Contact', etc. Below the menu, there is a list of articles with columns for Title, Author, Date submitted, Topic, and State. The article 'jotain muuta' by Author 5 is highlighted. Below the list, there is a 'Request for reviewing an article' form. The form includes the article title, author, and topic. It also has an 'Abstract' section and a question 'Would you like to review this article?' with 'Yes' and 'No' buttons. The page also shows a sidebar with 'Event.HYVAKSY\_ARVOSTELLTAVAKSI' and 'Event.HYLKAA\_ARVOSTELLUPYYNTO'.

Title	Writer	Date submitted	Topic	State
<a href="#">jotain muuta</a>	Author 3	2.4.2003	Databases	Reviewed (Rejected)
<a href="#">yabian tihä otstika</a>	Author 9	15.4.2003	Databases	Reviewed (Accepted with minor changes)
<a href="#">erilaisia</a>	Author 12	14.11.2003	Data mining	Not yet reviewed
<a href="#">yielä otain</a>	Author 5	13.12.2003	Databases	New article

**Request for reviewing an article**

**Title** jotain muuta **Writer** Author 5 **Topic** Data communications

**Abstract:**  
Tässä on artikkelin tiivistelmä.

**tiivistelmä**

[Show article](#)

Would you like to review this article?

Event.HYVAKSY\_ARVOSTELLTAVAKSI

Event.HYLKAA\_ARVOSTELLUPYYNTO

artikk\_id  
versionro  
setLausuntopyyntonTila(artikk\_id,versionro,kayt\_id, tila)

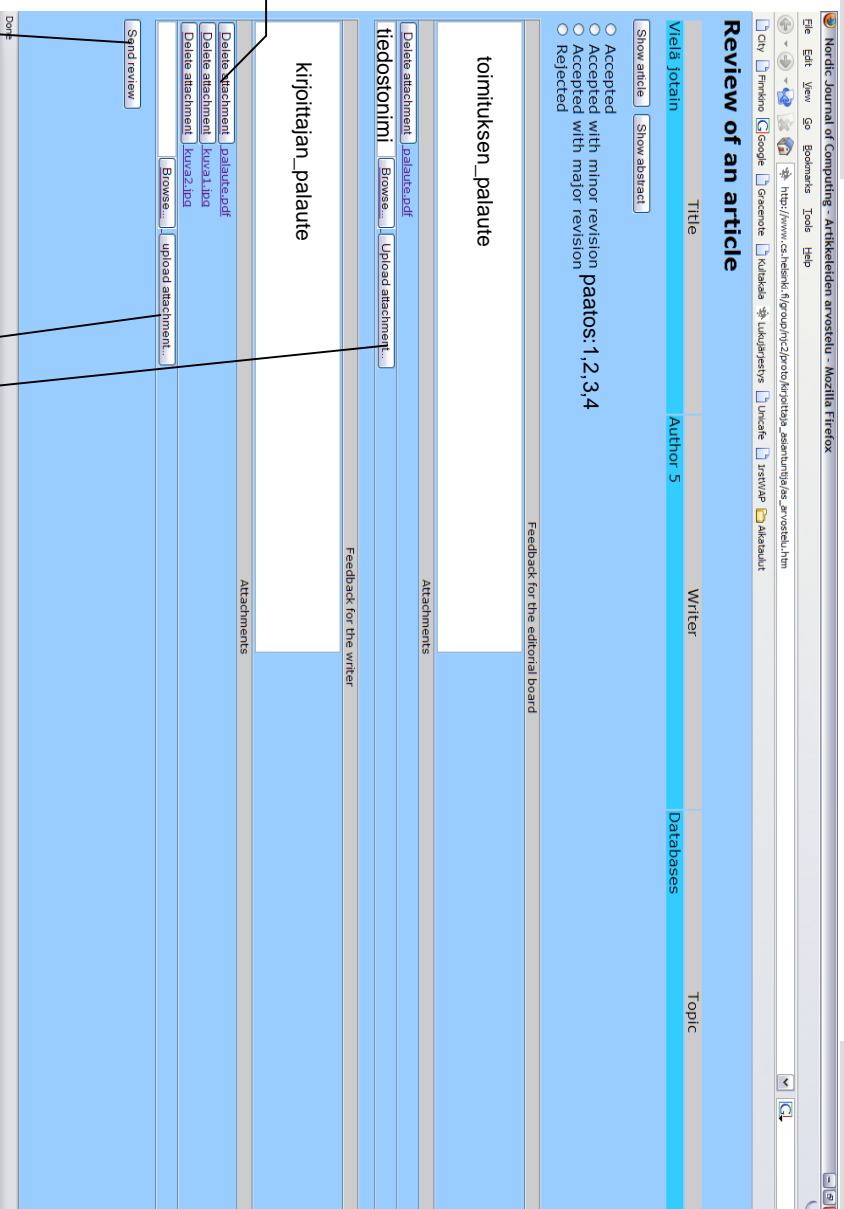
artikk\_id  
versionro  
setLausuntopyyntonTila(artikk\_id,versionro,kayt\_id, tila)

Saatavat parametrit

artik\_id  
versionro

Metodeita

Artikkeli: getArtikkeli(artik\_id,versionro)  
Lausunto: getLausunto(artik\_id,kayt\_id,versionro,  
boolean kirjoittaja)



Event: POISTA\_LIITE

artik\_id  
versionro  
liiteno

Event: LAHETA\_PALAUTE

submitLausunto(int artik\_id,int versionro, int kayt\_id, Lausunto toim, Lausunto kirj)

Event: LATAA\_LIITE

hiddent Kenttänä luokitus

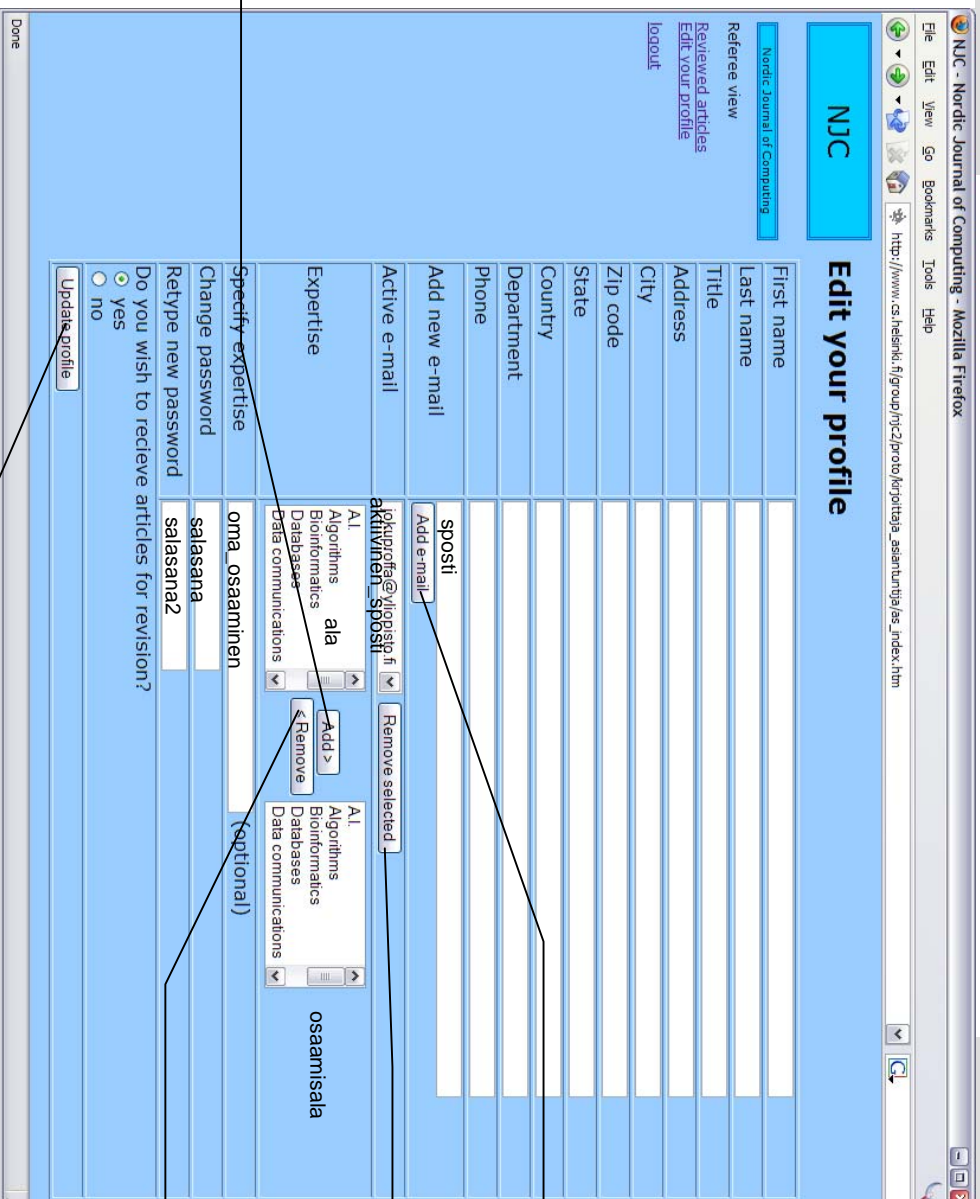


Saatavat parametrit

kayt\_id

Metodeita

Kayttaja getKayttajatiedot(kayt\_id)  
List getAlat()



Event:LISAA\_OMA\_ALA

alan\_nimi  
kayt\_id

Event:PAIVITA\_PROFILLI

submitKayttajatiedot(Kayttaja kayttaja,String salasana)

Event:LISAA\_SPOSTI

spositi, kayt\_id

Event:POISTA\_SPOSTI

spositi, kayt\_id

Event:POISTA\_OMA\_ALA

Event.JARJESTA\_TITLE

nayta\_kaikki

Event.JARJESTA\_WRITER

nayta\_kaikki

Event.JARJESTA\_TOPIC

nayta\_kaikki

Event.JARJESTA\_SUBMIT

Event.JARJESTA\_STATE

Event.AVAA\_ARTIKKELI

artik\_id  
versionno

Event.NAYTA\_KAIKKI

The screenshot shows the NJC (Nordic Journal of Computing) website. At the top, there are navigation links: "Organizing articles", "Organizing articles (re)", "State of magazine", "Organizing users", "Reports", "Edit your profile", and "logout".

The main content area is titled "Organizing articles" and contains a table with the following columns: Title, Author, Topic, Date submitted, and State. The table lists several articles, including "Jotain kuva oteliko", "Yhden tilin oteliko", "Siika samana", "Yhden melhentuntoksempaa", "Erlaisia", and "Jotain muuta".

Below the table, there is a section for "Jotain muuta" (Something new) with an abstract: "Tässä on artikkelin tiivistelmä." (This is a summary of the article.)

There are also search and submission options: "Search referee by expertise", "Show article", "Resend article", "Add referees to article", and "add referee to article >".

At the bottom, there is a section for "Referrees on this article" with a table listing referees and their expertise.

NJC - Nordic Journal of Computing - Mozilla Firefox  
<http://www.cs.helsinki.fi/group/njc/2003/omnis/index.htm>

NJC

Organizing articles

Title	Author	Topic	Date submitted	State
Jotain kuva oteliko	Author 1	Databases	2.5.2003	reference 1: not found reference 2: waiting for review reference 3: waiting for review
Yhden tilin oteliko	Author 2	Bioinformatics	3.6.2003	reference 1: not found reference 2: not found reference 3: not found
Siika samana	Author 3	Software engineering	15.7.2003	reference 1: waiting for review reference 2: waiting for review
Yhden melhentuntoksempaa	Author 4	A.I.	20.9.2003	reference 1: not found reference 2: not found reference 3: waiting for review
Erlaisia	Author 5	Databases	10.10.2003	reference 1: not found reference 2: waiting for review reference 3: not found
Jotain muuta	Author 6	Data communications	7.1.2004	reference 1: not found reference 2: waiting for review reference 3: not found reference 4: references not yet selected

**Jotain muuta** Author 6  
**Topic** Data communications

**Abstract:**  
 Tässä on artikkelin tiivistelmä.

**Name** Expertise E-mail  
 Referree 1 Databases Data communications hopohopo@helsinki.fi  
 Referree 2 A.I. Algorithms profa@uni.se

**Name** Expertise  
 Referree 1 Databases Data communication

**Name** Expertise  
 Referree 1 Databases Data communication

**Name** Expertise  
 Referree 1 Databases Data communication

**Name** Expertise  
 Referree 1 Databases Data communication

### Saatavat parametrit

artik\_id  
versionno  
String hakuehto  
String viesti  
jarjestä 1:name, 2:expertise

### Metodeita

```
List getAsiantuntijat(String hakuehto)  
List getAsiantuntijat(artik_id, versionno)  
int getAusuntopyyntöTia(kayt_id, artik_id, versionno)
```

The screenshot shows the 'Organizing articles' page in the Medica Journal of Computing. The page is divided into several sections:

- Organizing articles:** A table listing articles with columns: Title, Writer, Topic, Date submitted, and State. The table contains several rows of data, including articles like 'Jokain muuta', 'tivistelmä', and 'viesti'.
- Abstract:** A section for the selected article, showing the title 'tivistelmä' and a brief description.
- Article:** A section for the selected article, showing the title 'viesti' and a brief description.
- Referred articles:** A section for the selected article, showing a table of referred articles with columns: Name, Expertise, and Add reference. The table contains several rows of data, including articles like 'Referrae 1', 'Referrae 2', 'Referrae 3', 'Referrae 4', 'Referrae 5', and 'Referrae 6'.
- Add new referee:** A section for adding a new referee, showing a form with fields for Name, Expertise, and Notify.

Event: TALLENNA\_VIESTI

submitViesti(viesti, artik\_id, versionno, toimittajakayt)

Event: POISTA\_ARTIKKELI

Event: HYLKAA\_ARTIKKELI

Event: PYYDA\_UUDELEENLAHETYS

Event: HAE\_ASIAANTUNTUTIA

Event: JARJESTA\_NAME

Event: JARJESTA\_EXPERTISE

Event: LAHETA\_TIEDOTUS

tarkistetaan onko kyseessä  
1. lausuntopyyntö  
jos on, generoidaan salasana  
lähetetään sposti automaattisesti

Event: POISTA\_ASIAANTUNTUTIA\_ARTIKKELILISTA

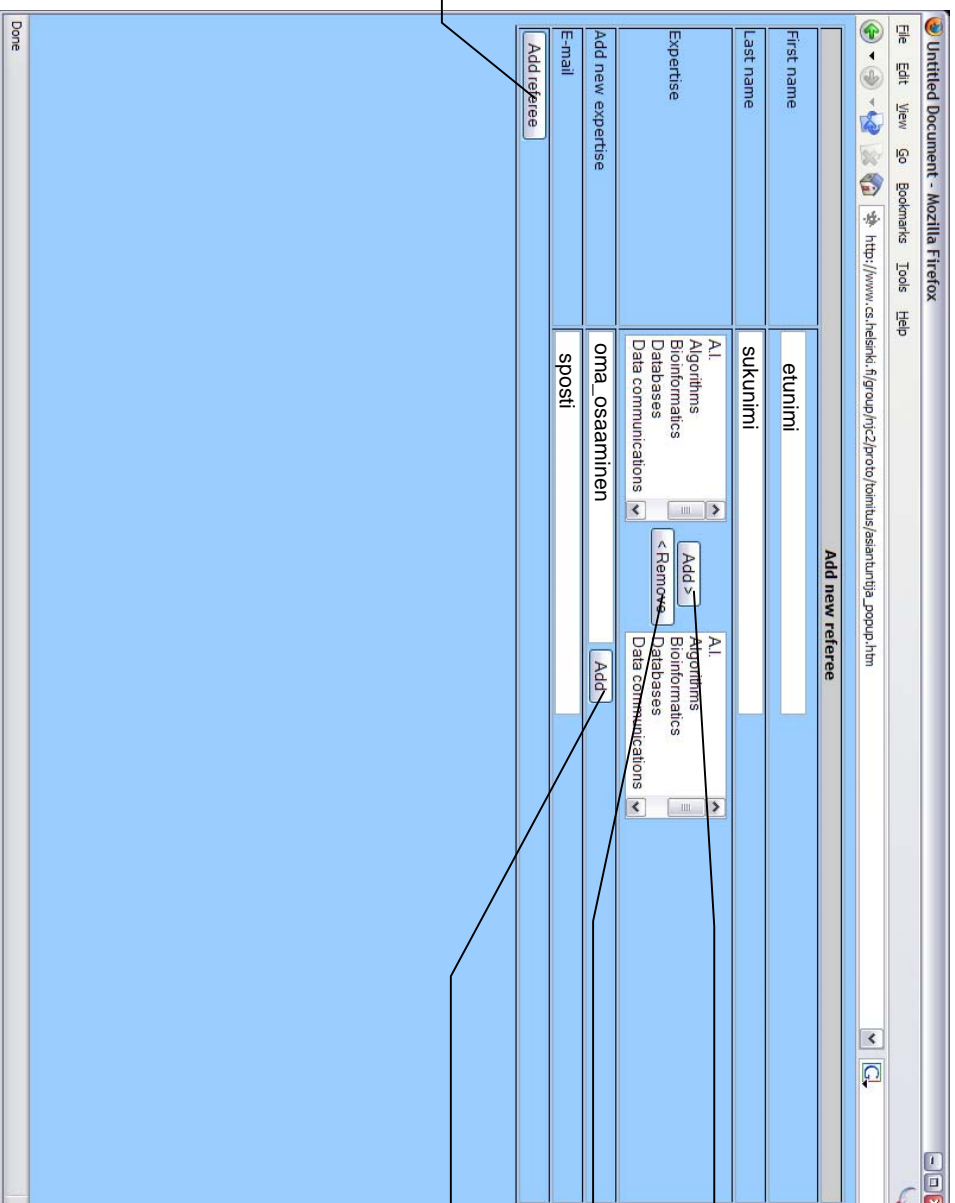
removeAsiantuntijaArtikkelista (artik\_id, versionno, kayt\_id)

Add new referee  
lisaasaasantuntija.jsp

Event: LISAA\_ASIAANTUNTUTIA\_ARTIKKELIIN

insertAsiantuntijaArtikkeliin ( artik\_id, versionno, kayt\_id)

submitAsiantuntija(Kayttaja)  
insertAla(alanNimi)



Event:LISAA\_ASIANTUNTJUA

Event:LISAA\_UUSI\_ALA

lisää oikeaan listaan, jos uusi,  
muuten näyttää valittuna olevaa  
vasemmassa listassa

Event:LISAA\_OMA\_ALA

Event:POISTA\_OMA\_ALA

Microsoft Internet Explorer  
 Address: http://www.cs.helsinki.fi/group/nc2/jproot/kommu/lausunnot.htm

**Title** Vähän tyvisä otsikko  
**Author** Author 2  
**Writer** Author 2  
**Topic** Data communications

**Accepted** 20.3.2004 Toimittaja 1

**Abstract:**  
 Thesis on architecture visualization.

Feedback for the writer		Feedback for the editorial staff	
referee 1 Accepted	Included attachments file.pdf	referee 1 täsmä palautte toimittajalle	Included attachments file.pdf
referee 2 Accepted	Included attachments file.doc	referee 2 täsmä palautte toimittajalle	Included attachments file.pdf
referee 3 Accepted with minor revision	Included attachments file.pdf	referee 3 täsmä palautte toimittajalle	Included attachments file.pdf

paatos: 1, 2, 3, 4

Event.LUO\_PAATOS

Event.LAUSUNNON\_UDELLEENPYNTTO

artik\_id  
versionno

getLhti(nro, vuosikerta)  
getLasiantuntijat(artik\_id, versionno)

NJC - Nordic Journal of Computing - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://www.as.helsinki.fi/group/njc2/proc/bannus/index.htm

**NJC**  
Nordic Journal of Computing

Organizing articles  
Organizing articles (ce)  
Database parameters  
Check for updates  
Check for updates  
Reports  
Edit your profile  
Logout

Show urgent articles Show all articles

Title	Writer	Topic	Date submitted	State
Jotain kiva otsikko	Author 1	Databases	2.5.2003	reference 1 reference 2 reference 3 accepted waiting for review waiting for review
Vähän tylsä otsikko	Author 2	Bioinformatics	3.6.2003	reference 1 reference 2 reference 3 accepted with minor revision accepted accepted
Sikä samaa	Author 3	Software engineering	15.7.2003	reference 1 reference 2 reference 3 waiting for review waiting for review waiting for review
Vähän mielenkiintoisempaa	Author 4	A.I.	20.9.2003	reference 1 reference 2 reference 3 waiting for review rejected waiting for review

**State of the journal**

Title	Author	Writer	Topic
Jotain muuta	Author 6		Data communications

Current issue: 2/5  
Move to: issue 1 volume 1

Select issue  
Show article  
Show abstract  
Publish this article

Referee	Remind referee	Last reminded
referee 1	Send reminder	12.2.2004
referee 2	Send reminder	never
referee 3	Send reminder	never

Title	Author	Topic	Date submitted	State	Remove
Vähän tylsä otsikko	author 2	Bioinformatics	3.6.2003	reference 1 reference 2 reference 3 reviewed reviewed reviewed	Remove from selected
Erläistä	author 5	Databases	10.10.2003	reference 1 reference 2 reference 3 reference 4 reviewed reviewed reviewed reviewed	Remove from selected
Jotain muuta	author 6	Data communications	7.1.2004	reference 1 reference 2 reference 3 reviewed reviewed waiting for review	Remove from selected

Event: JULKAISE\_ARTIKKELI

artik\_id  
versionno  
nro  
vuosikerta  
insertArtikkeilLehteen(Artikkeli, nro, vuosikerta)

Event: SIIRRY\_VALITTUUN\_LEHTEEN

nro  
vuosikerta

Event: LAHETA\_MUISTUTUS

kaytt\_id  
artik\_id  
versionno

Event: POISTA\_LEHDESTA

artik\_id  
versionno  
removeArtikkeilLehdesta(Artikkeli, nro, vuosikerta)

NJC - Nordic Journal of Computing - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

Http://www.cs.helsinki.fi/group/njc/product/komitas/index.htm

### NJC

Nordic Journal of Computing

- Editorial view
- Organizing articles
- Latest articles (62)
- Search articles
- Organization users
- Reports
- Edit your profile
- Reference view
- Reviewed articles
- Writer view
- Submit articles
- Logout

## Edit your profile

First name																					
Last name																					
Title																					
Address																					
City																					
Zip code																					
State																					
Country																					
Department																					
Phone																					
Add new e-mail																					
Active e-mail	jokuprotta@yliojoista.fi <input type="button" value="Remove selected"/>																				
Expertise	<table><tr><td>AI</td><td><input type="checkbox"/></td><td>Remove selected</td><td>AI</td><td><input type="checkbox"/></td></tr><tr><td>Domains</td><td><input type="checkbox"/></td><td></td><td>Domains</td><td><input type="checkbox"/></td></tr><tr><td>Databases</td><td><input type="checkbox"/></td><td></td><td>Databases</td><td><input type="checkbox"/></td></tr><tr><td>Data communications</td><td><input type="checkbox"/></td><td></td><td>Data communications</td><td><input type="checkbox"/></td></tr></table> <p><b>Katsastamattomia profiilitiivokkaus</b></p> <p>(optional)</p>	AI	<input type="checkbox"/>	Remove selected	AI	<input type="checkbox"/>	Domains	<input type="checkbox"/>		Domains	<input type="checkbox"/>	Databases	<input type="checkbox"/>		Databases	<input type="checkbox"/>	Data communications	<input type="checkbox"/>		Data communications	<input type="checkbox"/>
AI	<input type="checkbox"/>	Remove selected	AI	<input type="checkbox"/>																	
Domains	<input type="checkbox"/>		Domains	<input type="checkbox"/>																	
Databases	<input type="checkbox"/>		Databases	<input type="checkbox"/>																	
Data communications	<input type="checkbox"/>		Data communications	<input type="checkbox"/>																	
Specify expertise																					
Change password																					
Retype new password																					
Do you wish to receive articles for revision?	<input type="radio"/> yes <input type="radio"/> no																				
Update profile																					

Done

Saatavat parametrit

Event.JARJESTA\_NAME

Metodit  
getKayttajat()  
getKayttajat(jarjestysho)

Event.JARJESTA\_ROLE

Event.JARJESTA\_EMAIL

Event.POISTA\_ALA

Name	Role	Email
Author 1	writer	joni.gissan@i.fi
Author 2	writer	johanna.gissan@i.fi
Author 3	writer	prof@i.fi

### Organizing users

edit users profile

Katso asiantuntijan/kirjoittajan/toimittajan profiilin muokkaus

Event.MUOKKAA\_ALA

Event.LISAA\_UUSI\_ALA

Event.LISAA\_TOIMITTAJA

edit info

edit ala

Uusi ala

Add new expert

Add new editor

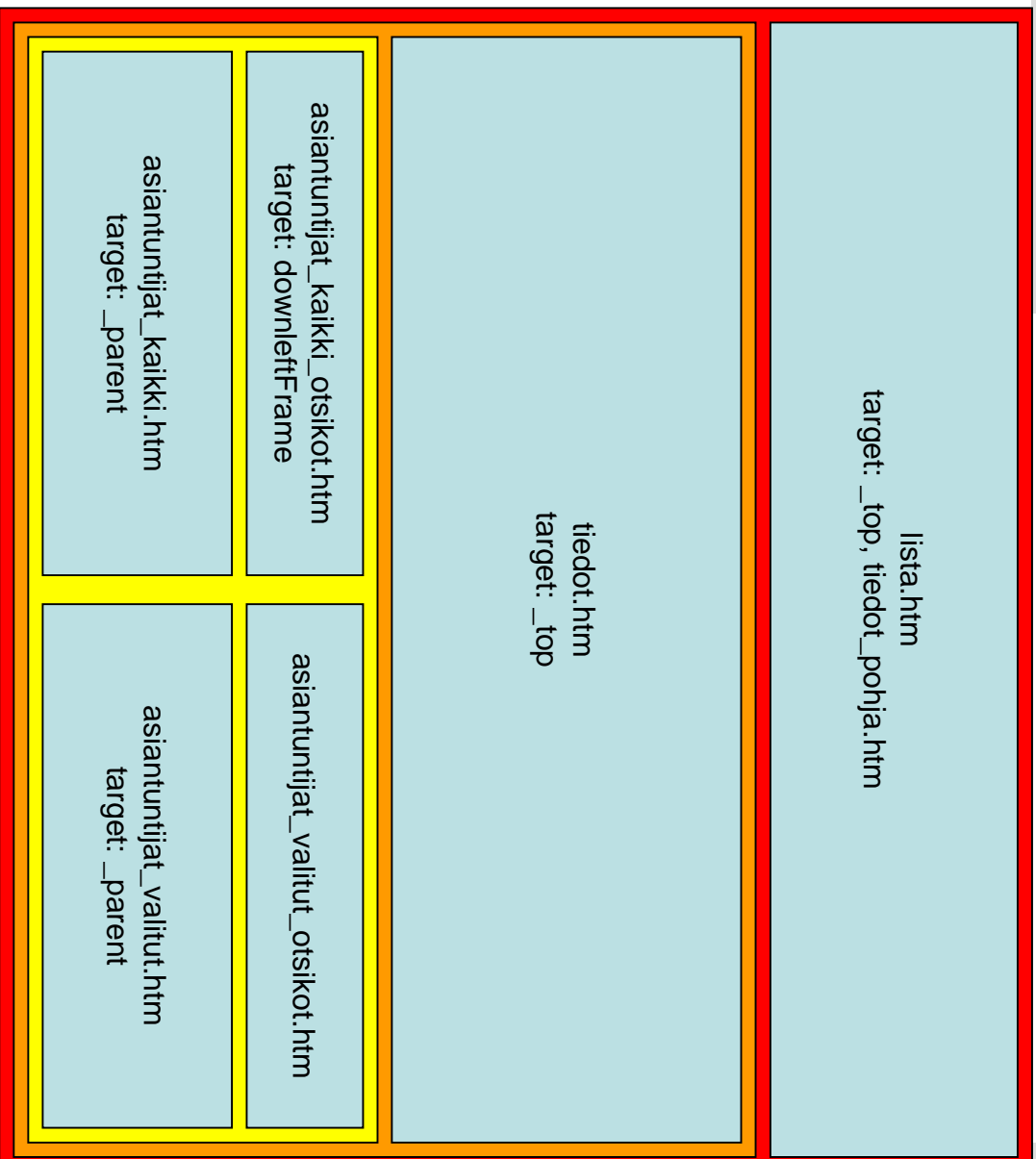
Add new referee

Katso asiantuntijan lisääminen (popup)

Done



# Toimittajan näkymä/ artikkelin tiedot



tiedot\_pohja.htm  
- topFrame  
- lowerFrame

tiedot\_alaosa.htm  
- centerFrame  
- bottomFrame

asiatuntijat.htm  
- upleftFrame  
- uprightFrame  
- downleftFrame  
- downrighFrame

## Toimittajan näkymä/ lausunnot

lista.htm  
target: \_top

lausunnot.htm  
target: \_self



tiedot\_pohja.htm  
- topFrame  
- lowerFrame

# Toimittajan näkymä/ lehden tila

lista.htm  
target: \_top

lehtivalikko.htm  
target: \_top

lehti\_lista.htm  
target: \_top



lehden\_tila.htm  
- topFrame  
- centerFrame  
- bottomFrame