

# **Määrittelydokumentti**

NJC2

Helsinki 16.2.2004

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	11.2.2004	Ensimmäinen versio
1.1	16.2.2004	Lisätty vaatimukset ja käyttötapauskaavio

# Sisältö

<b>1</b>	<b>Johdanto</b>	<b>1</b>
1.1	Dokumentin tarkoitus . . . . .	1
1.2	Dokumentin rakenne . . . . .	1
1.3	Käytettävät termit ja lyhenteet . . . . .	1
<b>2</b>	<b>Yleiskuvaus</b>	<b>2</b>
2.1	Toteutusympäristö . . . . .	2
2.2	Suoritusympäristö . . . . .	2
2.3	Käyttäjät . . . . .	2
2.3.1	Kirjoittajat . . . . .	2
2.3.2	Toimitus . . . . .	2
2.3.3	Asiantuntijat . . . . .	3
<b>3</b>	<b>Vaatimukset</b>	<b>3</b>
3.1	Vaatimusten prioriteetit . . . . .	3
3.2	Käyttjävaatimukset . . . . .	3
3.2.1	Toiminnalliset vaatimukset . . . . .	3
3.2.2	Ei-toiminnalliset vaatimukset . . . . .	4
3.3	Järjestelmävaatimukset . . . . .	4
3.4	Ympäristövaatimukset . . . . .	5
<b>4</b>	<b>Sidosryhmäkaavio</b>	<b>5</b>
<b>5</b>	<b>Kirjoittajan käyttötapaukset</b>	<b>6</b>
5.1	Rekisteröityminen . . . . .	6
5.2	Sisäänkirjautuminen . . . . .	6
5.3	Tietojen muokkaus . . . . .	6
5.4	Artikkelin lähettäminen . . . . .	6
5.5	Artikkelin tilanteen seuraaminen . . . . .	6
5.6	Lausunnon lukeminen . . . . .	6
5.7	Korjatun artikkelin lähettäminen . . . . .	7
5.8	Julkaistavan version lähettäminen . . . . .	7

<b>6 Toimituksen käyttötapaukset</b>	<b>7</b>
6.1 Sisäänkirjautuminen . . . . .	7
6.2 Saapuneiden artikkelien hallinta . . . . .	7
6.3 Asiantuntijoiden valinta . . . . .	7
6.4 Asiantuntijan lisäys . . . . .	7
6.5 Lausuntojen kokoaminen . . . . .	8
6.6 Artikkelien valinta . . . . .	8
6.7 Artikkelien julkaisu . . . . .	8
6.8 Korjattujen artikkelien hallinta . . . . .	8
6.9 Käyttäjien tietojen muokkaus . . . . .	8
6.10 Raporttien tulostus . . . . .	8
<b>7 Asiantuntijan käyttötapaukset</b>	<b>9</b>
7.1 Sisäänkirjautuminen . . . . .	9
7.2 Tietojen muokkaus . . . . .	9
7.3 Lausuntopyyntöön vastaaminen . . . . .	9
7.4 Lausunnon antaminen . . . . .	9
<b>8 Arkkitehtuurisuunnitelma</b>	<b>9</b>
8.1 Web-käyttöliittymät . . . . .	10
8.2 Datan hallinta ja käsittely . . . . .	10
8.3 Tietokanta . . . . .	10
<b>9 Testaus</b>	<b>10</b>
9.1 Yksikkötestaus . . . . .	11
9.2 Integraatiotestaus . . . . .	11
9.3 Validointitestaus . . . . .	11
9.4 Järjestelmätestaus . . . . .	11

## Liitteet

### 1 Käyttötapauskaavio

# 1 Johdanto

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

Projekti kuuluu Helsingin yliopiston tietojenkäsittelytieteen laitoksen Ohjelmistotuotantoprojekti-kurssiin. Järjestelmä tulee Nordic Journal of Computing -lehden toimituksen käyttöön. Ryhmän jäsenet ovat luovuttaneet projektin oikeudet Helsingin yliopistolle, joka julkaisee ohjelmiston GNU General Public License- tai GNU Lesser General Public License -lisenssin alaisuudessa.

## 1.1 Dokumentin tarkoitus

Määrittelydokumentin tarkoitus on kiinnittää asiakkaan vaatimukset ja määrittellä toteutettava ohjelmisto. Dokumentti laaditaan yhteistyössä asiakkaan kanssa, jotta se palvelee parhaalla mahdollisella tavalla sekä asiakasta että projektiryhmää. Vaatimusten lisäksi dokumentissa kuvataan järjestelmän tietosisältöä ja suunnittelun pääpiirteitä, mutta yksityiskohtainen toteutustapa määritellään suunnitteludokumentissa. Määrittelydokumentti toimii sopimuksena asiakkaan ja ryhmän välillä siitä, mitä tuotetaan, joten siinä pyritään määrittelemään mahdollisimman tarkasti tuotettava ohjelmisto.

Projektiryhmä ja asiakas käyvät dokumentin läpi 23.2. katselmoinnissa, jonka jälkeen asiakas esittää ryhmälle dokumenttiin tehtävät muutokset. Kun ryhmä on tehnyt muutokset, toimitetaan korjattu versio asiakkaalle. Asiakkaan hyväksyessä korjatun version dokumentti jäädytetään, jonka jälkeen asiakkaalla tai ryhmällä ei ole oikeutta muuttaa ohjelmiston vaatimuksia. Jos jäädytettyä dokumenttia joudutaan muuttamaan, tehdään tämä asiakkaan ja ryhmän yhteisellä päätöksellä ja muutokset eritellään dokumentin erilliseen lukuun.

## 1.2 Dokumentin rakenne

Luku 2 kuvaa ohjelmiston käyttäjät, suoritus- ja toteutusympäristön. Luvut 3-6 esittävät järjestelmän sidosryhmät ja heidän käyttötapaukset. Luku 7 kuvaa järjestelmän arkkitehtuurin ja viimeinen luku 8 ohjelmiston testauksen ja validoinnin.

## 1.3 Käytettävät termit ja lyhenteet

Termi/lyhenne	Merkitys
NJC	Nordic Journal of Computing -lehti, projektin asiakas
NJC2	Ryhmän nimi

## 2 Yleiskuvaus

Järjestelmän käyttö tapahtuu WWW-käyttöliittymän avulla, jonka avulla NJC-lehden kirjoittajat voivat jättää artikkeleitaan toimitukselle. Toimitus valitsee lähetetyille artikkeleille tyypillisesti kolme asiantuntijaa, jotka lukevat artikkelin, ja antavat mielipiteensä siitä. Tämän jälkeen toimitus voi päättää artikkelin julkaisusta, tai palauttaa sen takaisin kirjoittajalle korjattavaksi. Järjestelmä tarjoaa käyttöliittymän sekä kirjoittajille, arvioijille että toimitukselle.

### 2.1 Toteutusympäristö

Järjestelmä toteutetaan Java-versiolla J2SDK 1.4.2. Ohjelmisto saattaa toimia myös vanhemmilla versioilla, mutta sitä ei voida taata, eikä yhteensopivuuteen vanhempien versioiden kanssa kiinnitetä huomiota toteutusvaiheessa. Sovelluskehittimenä käytetään Eclipseä ja testauksissa käytetään JUnitia. Ohjelmiston komponentit yritetään tehdä järjestelmäriippumattomiksi, mutta sen toiminta tullaan varmistamaan vain Linuxissa. Versiohallintaan käytetään TKTL:n palvelimille asennettua CVS:ää. JSP-sivuja ja Servlettejä ajetaan Tomcat-ohjelmalla, josta käytetään 4.x versiota, joka tukee Servlet 2.3 ja JSP 1.2 -versiota.

### 2.2 Suoritusympäristö

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 Postgresiä.

### 2.3 Käyttäjät

#### 2.3.1 Kirjoittajat

Kirjoittajia ovat kaikki, jotka kirjoittavat lehteen artikkeleita julkaisua varten. Kirjoittajien rekisteröitymistä ei järjestelmän puolesta valvota millään tavalla: kuka tahansa voi rekisteröityä kirjoittajaksi. Toimitus voi kuitenkin seurata käyttäjien rekisteröitymistä ja valvoa artikkelien lähettämistä.

Järjestelmä tarjoaa kirjoittajalle mahdollisuuden lähettää artikkeleita ja seurata niiden käsittelyn etenemistä.

#### 2.3.2 Toimitus

Toimitukseen kuuluvat kaikki NJC-lehden toimittajat. Toimituksen tehtävänä on hallita lähetettyjen artikkeleiden elinkaarta ja koota niistä lopulta valmis lehti. Toimitusta avus-

tamassa on joukko asiantuntijoita, jotka toimitus valitsee.

Artikkeleiden hallinnan lisäksi järjestelmä tarjoaa toimitukselle mahdollisuuden pitää yllä asiantuntijatietokantaa.

### 2.3.3 Asiantuntijat

Asiantuntijat ovat toimituksen valitsemissa henkilöitä, joilla on erikoisosaamista tietyiltä aloilta. Asiantuntijat saavat toimitukselta omaan osaamisalaansa liittyviä artikkeleita koskien lausuntopyyntöjä, jotka asiantuntijalla on oikeus hyväksyä tai hylätä. Artikkelien julkaisu tai julkaisematta jättäminen päätetään toimituksessa asiantuntijoiden antamien lausuntojen perusteella.

Järjestelmä tarjoaa asiantuntijoille mahdollisuuden lukea heille lähetettyjä artikkeleita ja antaa niitä koskien lausuntoja.

## 3 Vaatimukset

### 3.1 Vaatimusten prioriteetit

Vaatimukset voidaan jakaa kolmeen ryhmään. Tärkeimpiä vaatimuksista ovat *kriittiset vaatimukset* (1), joita ilman ohjelmisto ei voi toimia. Jos jokin kriittinen vaatimus ei toteudu, projekti katsotaan epäonnistuneeksi.

*Välttämättömät* (2) vaatimukset ovat vaatimuksia jotka toteutetaan, mutta jotka eivät ole kriittisiä vaatimuksia. Koko projektin ei suoraan katsota epäonnistuneen vaikka jokin välttämätön vaatimus ei toteutuisikaan.

*Mahdolliset* (3) vaatimukset kuvaavat alhaisimman prioriteetin vaatimuksia ja niiden sisällöstä ja toteuttamisen aikataulusta sovitaan asiakkaan kanssa erikseen, mikäli kriittiset ja välttämättömät vaatimukset saadaan aikataulun puitteissa toteutettua.

### 3.2 Käyttäjävaatimukset

#### 3.2.1 Toiminnalliset vaatimukset

Koodi	Kuvaus	Prio
GUI.01	Käyttöliittymistä tehdään mahdollisimman optimoituja	2

#### Kirjoittajan käyttöliittymä

<b>Koodi</b>	<b>Kuvaus</b>	<b>Prio</b>
KIR.01	Kirjoittajan rekisteröityminen	1
KIR.02	Kirjoittajan sisäänkirjautuminen	1
KIR.03	Kirjoittajan artikkelin lähettäminen PS/PDF-muodossa	1
KIR.04	Omaa artikkelia koskevan lausunnon lukeminen	1
KIR.05	Korjatun artikkelin lähettäminen	1
KIR.06	Oman artikkelin tilanteen seuraaminen	2
KIR.07	Julkaisukelpoisen version lähettäminen LaTeX-muodossa	2

### Toimituksen käyttöliittymä

<b>Koodi</b>	<b>Kuvaus</b>	<b>Prio</b>
TOI.01	Saapuneiden artikkelien hallinta	1
TOI.02	Asiantuntijoiden valinta	1
TOI.03	Asiantuntijan lisäys (tiedot ja osaamisalueet)	1
TOI.04	Lausuntojen kokoaminen ja välitys kirjoittajalle	1
TOI.05	Artikkelien lopullinen julkaisu lehteen	1
TOI.06	Korjattujen artikkelien asiantuntijoiden vaihto	1
TOI.07	Artikkelien valinta haluttuun numeroon	2
TOI.08	Värikoodaus asiantuntijoiden kuormituksille	3

### Asiantuntijan käyttöliittymä

<b>Koodi</b>	<b>Kuvaus</b>	<b>Prio</b>
ASI.01	Sähköpostitiedotus asiantuntijalle tarkastettavasta artikkelista	1
ASI.02	Asiantuntijan sisäänkirjautuminen	1
ASI.03	Artikkeleiden toimitus asiantuntijalle sähköpostilla, jossa linkki	1
ASI.04	Asiantuntijan lausunnon antaminen: hyväksyty, hyväksyty pienin tai suurin muutoksin, hylätty	1

### 3.2.2 Ei-toiminnalliset vaatimukset

<b>Koodi</b>	<b>Kuvaus</b>	<b>Prio</b>
DOK.01	Dokumentointi ja koodi kirjoitetaan suomeksi	1
DOK.02	Käyttöliittymä kirjoitetaan englanniksi	1
DOK.03	Käyttöliittymästä kirjoitetaan myös suomenkielinen versio	3

## 3.3 Järjestelmävaatimukset

Järjestelmää varten tarvitaan seuraavat palvelimet:

- Tomcat Java-luokkia, Java Servlettejä sekä JSP-sivuja varten

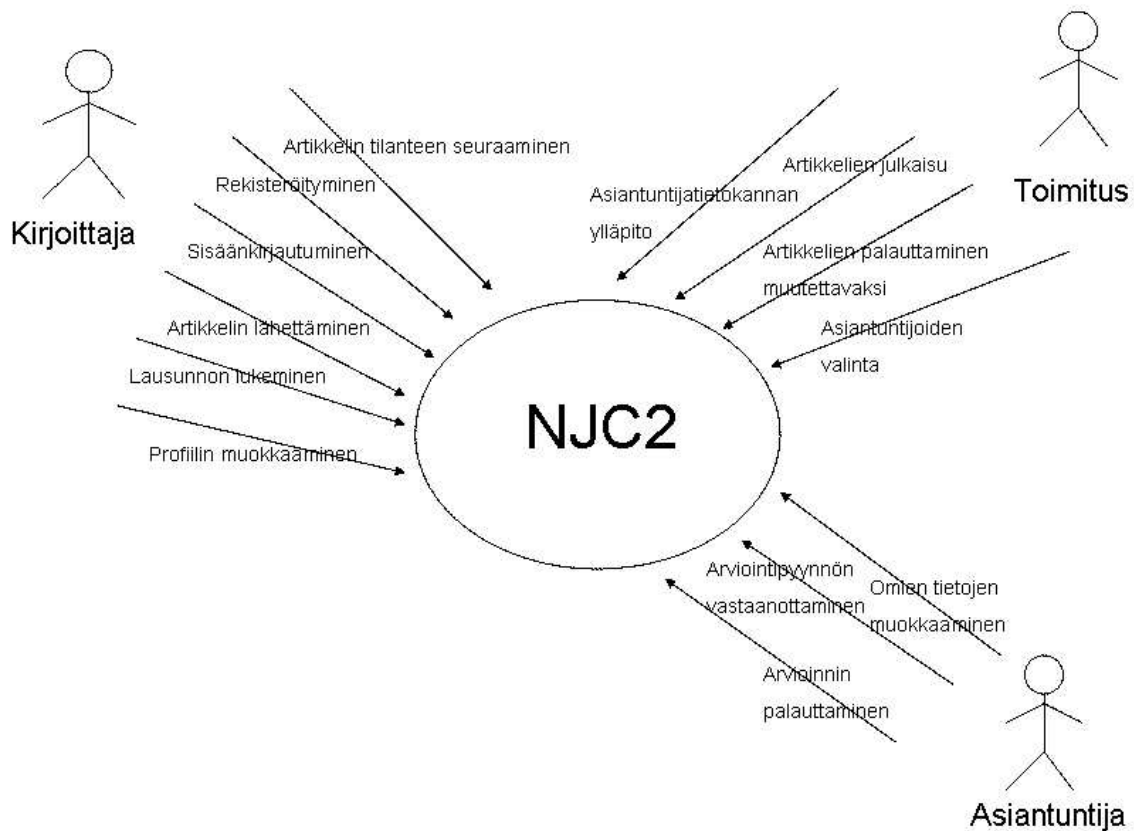


- PostgreSQL tietokantaa varten
- Apache WWW-server staattisia sivuja varten (HTML, css, binääritiedostot)

### 3.4 Ympäristövaatimukset

Järjestelmän käyttöä varten tarvitaan WWW-selain, joka tukee kehyksiä (frameset), javascriptiä sekä css-tyylitiedostoja (esim. Internet Explorer 6.0, Mozilla 1.6). Näytön resoluution on oltava vähintään 800x600.

## 4 Sidosryhmäkaavio



Kuva 1: Sidosryhmäkaavio

## **5 Kirjoittajan käyttötapaukset**

### **5.1 Rekisteröityminen**

Uusi käyttäjä haluaa lähettää tuoreen artikkelinsa lehden toimitukselle. Hän rekisteröityy järjestelmään syöttämällä nimensä, sähköpostiosoitteensa, postiosoitteensa ja valitseman sa salasanan lomakkeeseen. Rekisteröitymisen jälkeen järjestelmä kirjaa käyttäjän sisään automaattisesti.

### **5.2 Sisäänkirjautuminen**

Aikaisemmin rekisteröitynyt käyttäjä syöttää sähköpostiosoitteensa ja salasanasensa niille osoitettuihin kenttiin ja kirjautuu sisään järjestelmään.

### **5.3 Tietojen muokkaus**

Käyttäjän postiosoite on muuttunut. Hän kirjautuu sisään ja syöttää muuttuneet tietonsa muokkauslomakkeeseen.

### **5.4 Artikkelin lähettäminen**

Sisäänkirjautunut käyttäjä päättää lähettää artikkelinsa toimitukselle. Hän syöttää lomakkeeseen artikkelin otsikon, tiivistelmän ja itse artikkelin PS/PDF-muodossa. Lähetettyään tiedot toimitukselle käyttäjä näkee artikkelin siirtyneen kantaan, ja sen käsittelytilanteen.

### **5.5 Artikkelin tilanteen seuraaminen**

Käyttäjä haluaa nähdä, missä vaiheessa hänen kaksi viikkoa sitten lähettämänsä artikkelin käsittely on. Kirjaututtuaan sisään järjestelmään käyttäjä näkee kaikkien lähettamiensä artikkeleiden käsittelytilanteen. Järjestelmä kertoo kyseisen artikkelin odottavan asiantuntijalausuntoja.

### **5.6 Lausunnon lukeminen**

Käyttäjä saa sähköpostilla ilmoituksen asiantuntijalausuntojen valmistumisesta. Kirjaututtuaan sisään järjestelmään käyttäjä valitsee lähettämiensä artikkeleiden joukosta kyseistä artikkelia koskevan lausunnon ja lukee sen. Artikkelin on hyväksytty julkaistavaksi pienin muutoksin.

## 5.7 Korjatun artikkelin lähettäminen

Käyttäjä on tehnyt artikkeliin asiantuntijoiden ja toimituksen haluamat muutokset ja haluaa lähettää sen uudestaan julkaisua varten. Hän etsii lähettämiensä artikkeleiden joukosta oikean artikkelin ja valitsee uudelleenlähetyksen. Syötettyään osittain valmiiksi täytettyyn lomakkeeseen itse PS/PDF-muotoisen artikkelin, käyttäjä lähettää tiedot toimitukselle. Korjattu versio ilmestyy lähetettyjen artikkeleiden listaan.

## 5.8 Julkaistavan version lähettäminen

Toimitus ja asiantuntijat ovat hyväksyneet artikkelin julkaistavaksi ja tarvitsevat sen käsiteltävämässä muodossa. Kirjoittaja saa tiedon artikkelinsa julkaisusta ja lähettää sen toimitukselle  $\LaTeX$ -muodossa.

# 6 Toimituksen käyttötapaukset

## 6.1 Sisäänkirjautuminen

Toimittaja kirjautuu sisään järjestelmään omalla tunnuksellaan ja pääsee käsittelemään lehteä muiden toimittajien lailla.

## 6.2 Saapuneiden artikkelien hallinta

Toimittaja huomaa kolmen uuden artikkelin saapuneen ja lukee niihin liittyvät tiivistelmät. Ensimmäinen ei toimittajan mielestä kuulu lainkaan lehden aihepiiriin, joten hän lähettää kirjoittajalle hylkäävän päätöksen. Kaksi muuta artikkelia toimittaja lukee kokonaan, ja päättää lähettää ne asiantuntijoille arvioitavaksi.

## 6.3 Asiantuntijoiden valinta

Toimittaja etsii sopivia asiantuntijoita saapuneelle artikkelille "Langaton tietoliikenne hajautetuissa järjestelmissä". Toimittaja on lukenut artikkelin läpi ja löytää asiantuntijalistasta kolme aihepiiriin tuntevaa asiantuntijaa, joille artikkeli lähetetään lausuntoja varten.

## 6.4 Asiantuntijan lisäys

Toimituksessa on pohdittu erään yliopiston professorin lisäämistä asiantuntijatietokantaan. Keskusteltuaan professorin kanssa puhelimesta toimittaja syöttää lomakkeeseen hänen tietonsa ja osaamisalueensa. Professori saa sähköpostitse ilmoituksen lisäämisestä.

## **6.5 Lausuntojen kokoaminen**

Artikkeliin "Langaton tietoliikenne hajautetuissa järjestelmissä" on tullut lausunto kaikilta kolmelta siihen valitulta asiantuntijalta. Yksi asiantuntijoista hyväksyy artikkelin suoraan julkaisuun, kahden muun mielestä se hyväksytään pienin muutoksin. Toimittaja lukee lausunnot läpi ja lähettää niistä haluamansa eteenpäin kirjoittajalle.

## **6.6 Artikkelien valinta**

Toimitus on kokoamassa lehden tammikuun numeroa. Artikkelien määräksi on sovittu viisi. Neljä sopivaa artikkelia on jo saanut hyväksyvät lausunnot asiantuntijoilta, mutta viides odottaa kirjoittajan tekemän artikkeliinsa pieniä korjauksia. Julkaisuun on aikaa kaksi viikkoa, ja toimittaja päättää valita keskeneräisen artikkelin mukaan tammikuun numeroon odottamaan julkaisua.

## **6.7 Artikkelien julkaisu**

Tammikuun numeron julkaisuun on jäljellä viikko. Toimittaja näkee toimituksen valitsemien viiden artikkelin tilan ja toteaa, että viimeinenkin artikkeli lähetettiin korjattuna ja tarkastettiin kaksi päivää sitten.

## **6.8 Korjattujen artikkelien hallinta**

Toimittaja huomaa kirjoittajan saaneen korjaukset valmiiksi ja lähettäneen artikkelinsa uudelleen toimitukselle. Yksi kolmesta artikkelia arvioineesta asiantuntijasta ei ole tavoitettavissa kahteen viikkoon, joten toimittaja päättää valita kolmanneksi asiantuntijaksi jonkun muun.

## **6.9 Käyttäjien tietojen muokkaus**

Toimittaja tutkii käyttäjien tietoja ja huomaa, että yhdeltä käyttäjältä on tullut viikon aikana seitsemän artikkelia, jotka on hylätty välittömästi. Toimittaja päättelee, että kyseessä on pelkkä häirikkö, ja poistaa käyttäjän tietokannasta. Samalla toimittaja muuttaa toisen käyttäjän tilan pelkästä kirjoittajasta asiantuntijaksi.

## **6.10 Raporttien tulostus**

Toimitus haluaa saada tilastotietoa viimeaikaisten artikkelien elinkaarista ja asiantuntijoiden lausunnoista. Toimittaja tulostaa niitä koskevan raportin ulos järjestelmästä tarkempaa analysointia varten.

## 7 Asiantuntijan käyttötapaukset

### 7.1 Sisäänkirjautuminen

Asiantuntija kirjautuu sisään syöttämällä sähköpostiosoitteensa ja toimituksen sähköpostilla lähettämän salasanan niille osoitettuihin kenttiin. Kirjaututtuaan sisään järjestelmään asiantuntija voi muokata tietojaan, tutkia hänelle lähetettyjä artikkeleita sekä lähettää omia artikkeleitaan toimitukselle.

### 7.2 Tietojen muokkaus

Kiireinen asiantuntija vaihtaa työpaikkaa kuudeksi kuukaudeksi. Hän muokkaa tietojaan lomakkeella ja päättää ohjata sähköpostinsa uuden työpaikan osoitteeseen.

### 7.3 Lausuntopyyntöön vastaaminen

Järjestelmä lähettää asiantuntijalle sähköpostitse ilmoituksen saapuneesta lausuntopyyntöstä. Kirjaututtuaan sisään asiantuntija voi lukea artikkeliin liittyvän tiivistelmän ja ladata itselleen koko artikkelin, minkä jälkeen hän joko hyväksyy tai hylkää lausuntopyyntön.

### 7.4 Lausunnon antaminen

Luettuaan artikkelin kokonaan asiantuntija valitsee yhden seuraavista päätöksistä:

- Hyväksytty
- Hyväksytty pienin muutoksin
- Hyväksytty suurin muutoksin
- Hylätty

Asiantuntija voi kirjoittaa kahdenlaisia lausuntoja: toiset lähtevät kirjoittajalle, toiset ovat pelkästään toimituksen nähtävissä.

## 8 Arkkitehtuurisuunnitelma

Järjestelmä jaetaan seuraaviin osajärjestelmiin:

- Web-käyttöliittymät

- Datan hallinta ja käsittely
- Tietokanta.

## 8.1 Web-käyttöliittymät

Järjestelmän käyttäjille näkyvin osa on luonnollisesti käyttöliittymä. Järjestelmällä on kolme käyttäjäryhmää: artikkelien kirjoittajat, asiantuntijat ja toimitus. Jokaisella ryhmällä on oma JSP-sivuilla toteutettava web-pohjainen käyttöliittymä. Myös Java Servlet -tekniikkaa voidaan käyttää käyttöliittymissä. Tietoturvaan kiinnitetään merkittävästi huomiota käyttöliittymien toteutuksessa, jottei käyttäjiin tai artikkeleihin liittyvää informaatiota pääse ulkopuolisten käsiin.

## 8.2 Datan hallinta ja käsittely

Tämä osajärjestelmä muodostaa järjestelmän varsinaisen toiminnallisuuden toimien komponenttina käyttöliittymien ja tietokannan välillä. Hallittavaan ja käsiteltävään dataan sisältyy mm. artikkelit, artikkelien kirjoittajat ja asiantuntijat. Osajärjestelmä toteutetaan normaaleina Java-luokkina.

## 8.3 Tietokanta

Järjestelmän sisältämä data sijaitsee tietokannassa, jota Datan hallinta ja käsittely -osajärjestelmä käsittelee. Tietokanta toteutetaan PostgreSQL-tietokantaohjelmistolla.



## 9 Testaus

Ohjelmiston laadun varmistamiseksi projektiryhmä testaa ohjelmiston testausvaiheessa, joka alkaa osittain toteutusvaiheen aikana yksikkötestauksen osalta ja jatkuu toteutusvai-

heen jälkeen muilta osin. Mitä varhaisemmassa vaiheessa testausta suoritetaan, sitä helpompi virheet on korjata ja niiden haittavaikutukset jäävät vähäisimmiksi. Tietokantaoperaatioiden tarkastamiseksi luomme mahdollisimman varhaisessa vaiheessa esimerkkietokannan, jolla tarkistetaan kaikki ohjelmistossa käytettävät tietokantaoperaatiot. Tarkka testauksen suunnitteludokumentti valmistuu omana dokumenttinaan ennen testausvaiheen alkamista. Testaus suoritetaan vaiheittain yksikkötestauksena, integraatiotestauksena, validointitestauksena, sekä järjestelmätestauksena. Ryhmä korjaa testausvaiheessa löytyneet virheet, mikäli ohjelmiston toteutusvaihe ei ole päättynyt.

## **9.1 Yksikkötestaus**

Yksikkötestaus suoritetaan käyttäen JUnit-yksikkötestauskehystä, jossa ohjelman yksittäiset komponentit testataan heti niiden valmistuttua. Tarkoituksena on testata komponentin toimivuus dokumentoidulla tavalla. Jokaisessa komponentissa pyritään arvoalue-testaukseen, eli valitaan yksi testitapaus arvoalueen sisältä ja lisäksi kaksi testitapausta jokaisen arvoalueen rajan molemmin puolin.

## **9.2 Integraatiotestaus**

Testattujen komponenttien yhteistoiminta testataan integrointitestauksena sitä mukaa, kun kokonaisuuksia saadaan kasaan. Testauksen tavoitteena on varmistaa komponenttien toimivuus yhdessä ja dokumentoida löytyneet virheet. Virheet korjataan, mikäli ohjelmiston toteutusvaihe ei ole päättynyt.

## **9.3 Validointitestaus**

Validointitesteissä tarkistetaan, onko ohjelmisto toteutettu asiakkaan vaatimusten mukaan, vai onko jotakin jäänyt kokonaan toteuttamatta, tai toteutettu toisella tavalla. Ohjelmiston osalta projekti on onnistunut, kun prioriteetin 1 toiminnot toimivat määrittelyn mukaisesti ja ohjelmakoodi on helposti ylläpidettävää.

## **9.4 Järjestelmätestaus**

Lopuksi järjestelmä testataan kokonaisuudessaan esimerkkikäyttötapausten avulla.

# Liite 1. Käyttötapauskaavio

