

Tampereen ammattikorkeakoulu  
Tietojenkäsittelyn koulutusohjelma  
Mikko Ojala

OPINNÄYTETYÖ

JULKAISUJÄRJESTELMÄN KÄYTTÖÖNOTTO - JOOMLA!

Työn ohjaaja Petri Heliniemi  
Työn tilaaja Veljekset Keskinen Oy  
Tampere 03/2009

Tekijä(t)	Mikko Ojala
Työn nimi	Julkaisujärjestelmän käyttöönotto - Joomla!
Sivumäärä	56+18
Valmistumisaika	maaliskuu 2009
Työn ohjaaja	Petri Heliniemi
Työn tilaaja	Veljekset Keskinen Oy

---

## TIIVISTELMÄ

Työ tehtiin toimeksiantona Veljekset Keskinen Oy:lle. Aloitin keväällä 2007 työharjoittelun / äitiyslomasijaisuuden Veljekset Keskinellä ja alusta asti oli suunnitelmissa, että myös opinnäytteen tulon tekemään sinne. Ajankohdaksi suunniteltiin alkukesää 2008. Työnaihe selvisi kun kävi ilmi, että uutta versiota julkaisujärjestelmästä oli tarkoitus ottaa käyttöön ja, että ohjeistusta tarvitaan uuden käyttöön.

Työn tavoitteena toimeksiantajan näkökulmasta oli saada uudet ohjeet henkilöstölle uuden julkaisujärjestelmän käyttöä varten. Tarkoituksena oli myös käydä läpi julkaisujärjestelmän käyttöönottoprosessi ja saada siitä kattava teoriaosuus työlle. Henkilökohtaisena tavoitteen tekijällä oli myös perehdyttää itseään julkaisujärjestelmän käyttöön ja siihen liittyviin ongelmiin.

Tiedot ja taidot työn aiheesta olivat alussa keskinkertaiset. Perehtymällä aihealuetta käsitteleviin teoksiin ja lähteisiin, hankittiin työn tekemiseen vaadittava tietämys. Opettelemalla työssä käsiteltävän julkaisujärjestelmän käytön sai ohjeiden, sekä teorian laatiminen uuden näkökulman.

Käytetyt työ- ja tutkimusmenetelmät olivat harkittuja ja etukäteen suunniteltuja mutta käytetyn aineiston hankinnan kohdalla ne vaihtelivat. Aineiston hankinta oli alussa vaikeaa ja sopivia lähteitä ei tahtonut löytyä. Lähdekritiikki oli tarkkaa, koska suurin osa lähteistä oli verkosta löydettyjä. Verkosta oleviin lähteisiin pyrittiin kelpuuttamaan pelkästään sellaiset, jotka oli julkaistu luotettavilla sivustoilla. Työn edetessä lähteiden etsintä parani, uusien hakusanojen ja hakukriteerien avulla.

Työn päätuloksina syntyivät Joomla!-julkaisujärjestelmän käyttöohjeet Veljekset Keskinen Oy:lle ja kattava tarkastelu julkaisujärjestelmän käyttöönottoprosessista. Päätulokset ovat hyödynnettävissä vastaaviin julkaisujärjestelmän käyttöönottoihin. Ohjeiden osalta tulee tietysti muistaa, että osa niiden sisällöstä on tehty ajatellen Veljekset Keskinen Oy:n käyttötarkoituksia.

Writer(s)	Mikko Ojala
Thesis	Content management system implementation - Joomla
Pages	56+18
Graduation time	March 2009
Thesis Supervisor	Petri Heliniemi
Co-operating Company	Veljekset Keskinen Oy

---

## ABSTRACT

This thesis work was carried out as an assignment to Veljekset Keskinen Oy. In spring 2007, I started doing my work placement and worked as a maternity leave substitute at Veljekset Keskinen Oy. Already early on it was planned that I would also do my thesis there. The thesis work was planned to take place in early summer of 2008. The subject of my thesis became clear when it turned out that my employer was going to update their content management system. They also needed instructions concerning the use of the updated version of the content management system.

From the employer's point of view, the goal was to get a new user's guide to help the company's personnel to use the new content management system. Another goal was to study the content management system implementation process and make it a comprehensive theory part for the thesis. I had a personal goal to familiarize myself with the usage of a content management system and its problems.

My skills and knowledge about the topic were mediocre in the beginning. By doing some research on the matter, I achieved the required knowledge. Studying the content management system used in the thesis also gave me another point of view in writing the user's guide and the theory part of the thesis.

The work and research methods used in the thesis were planned beforehand. Only the methods that concerned gathering the material that was used varied during the process. In the beginning, it was hard to acquire the material. The fact that a large portion of the material was found on the Internet made the criticism toward it crucial and required precision. I tried to accept only Internet material that was found on trusted websites. As the writing process proceeded, the search of new material got easier because of new keywords and search criteria.

The main results of this thesis were Joomla! -user's guide for the employer Veljekset Keskinen Oy and a comprehensive analysis of the content management system implementation. The results can be utilized in comparable content management system implementations. However, it should be taken into notice that a part of the Joomla! -user's guide content is written specifically to meet the needs of Veljekset Keskinen Oy.

# Sisällysluettelo

1 JOHDANTO .....	5
2 MIKÄ ON JULKAISUJÄRJESTELMÄ? .....	7
2.1 YLEISTÄ .....	7
2.2 RAKENNE.....	8
2.3 TARKOITUS JA KÄYTTÖ.....	9
2.4 JULKAISUJÄRJESTELMIEN ERI MUODOT .....	10
2.4.1 RASKAAT JÄRJESTELMÄT .....	10
2.4.2 KESKIRASKAAT JÄRJESTELMÄT .....	11
2.4.3 KEVYET JA ULTRAKEVYET JÄRJESTELMÄT .....	11
2.4.4 OPEN SOURCE JÄRJESTELMÄT .....	12
3 JULKAISUJÄRJESTELMÄN VALINTA .....	13
3.1 JULKAISUJÄRJESTELMÄN VALINTAKRITEERIT .....	13
3.2 KÄYTTÖYMPÄRISTÖN TEKNINEN MÄÄRITTELY.....	15
3.3 VAIHTOEHDOT JOOMLA!:N LISÄKSI .....	17
3.4 YHTEENVETO .....	18
3.5 JOOMLA!:N OMINAISUUDET .....	19
4 JULKAISUJÄRJESTELMÄN KÄYTTÖÖNOTTOPROSESSI .....	21
4.1 JOOMLA!:N HANKINTA JA ASENNUS .....	21
4.2 JULKAISUJÄRJESTELMÄN RÄÄTÄLÖIMINEN .....	22
4.2.1 LIITÄNNÄISET JA MODUULIT .....	22
4.2.2 LAAJENNUKSET.....	25
4.3 YHTEENSOPIVUUS JA AJANTASAISUUS / YLLÄPITO .....	25
4.3.1 JOOMLA!-VERSIOT JA PÄIVITYKSET .....	26
4.3.2 KÄYTTÖYMPÄRISTÖN PÄIVITYKSET .....	27
4.3.3 MUUT YLLÄPIDOLLISET ASIAT .....	28
5 JOOMLA! KÄYTÄNNÖSSÄ .....	29
5.1 ULKOASU.....	29
5.2 SIVUPOHJA.....	31
5.3 SISÄLLÖN MÄÄRITYS .....	33
5.4 TOIMINTOJEN HAVAINNOLLISTAMINEN ESIMERKKIEN AVULLA .....	34
5.4.1 ESIMERKIT .....	35
6 JULKAISUJÄRJESTELMÄN OHJEET .....	42
6.1 YLEISTÄ OHJEISTA JA NIIDEN TEKEMISESTÄ.....	42
6.2 OHJEIDEN SUUNNITTELU JA TOTEUTUS.....	43
6.3 ARTIKKELIN LISÄÄMINEN .....	45
6.4 TARJOUSSIVUJEN LISÄÄMINEN .....	45
7 JOHTOPÄÄTÖKSET JA YHTEENVETO .....	46

# 1 Johdanto

Perehdyin julkaisujärjestelmän käyttöönottoprosessiin, sekä tein toimeksiantajan ympäristöön soveltuvan käyttöoppaan koskien julkaisujärjestelmä Joomla!':n käyttöä. Tarkastelen tässä opinnäytetyöraportissani käyttöönottoprosessin teoriaa ja käyttöoppaan tekemiseen liittyviä asioita.

Toimeksiantajana toimii Suomen suurin kyläkauppa eli Veljekset Keskinen Oy. Veljekset Keskinen Oy työllistää noin 400 vakituista työntekijää ja kesän aikana tämä luku lähes kaksinkertaistuu kesätyöntekijöiden astuessa taloon. Aloitin työskentelyn toimeksiantajalla keväällä 2007. Alkuun tein työharjoittelua, joka jatkui sitten vielä sijaisuutena vuoden 2008 syksyyn. Harjoittelun yksi tavoitteista oli, että löydämme yhdessä työnantajan kanssa sopivan opinnäytetyön aiheen. Sopiva aihe löytyi, kun kävi ilmi, että työantajalla oli julkaisujärjestelmän vaihdos edessä.

Päätavoitteena oli laatia opinnäytetraportti, joka sisältää kattavan ja selkeän kuvauksen julkaisujärjestelmän käyttöönotosta. Käyttöönottoprosessin tarkastelun lisäksi tarkoitus oli kirjoittaa ohjeen laatimiseen liittyvästä teoriasta. Lisäksi tavoitteena oli luoda toimeksiantajan käyttöympäristöön soveltuva käyttöopas julkaisujärjestelmä Joomla!':a varten. Tämän käyttöoppaan takia työhön otettiin mukaan myös ohjeistukseen liittyvää teoriaa. Käyttöoppaan teko oli hyödyllinen tavoite niin toimeksiantajalle kuin itselleni. Toivoin ja osaltani myös tiesin, että tutustuessani julkaisujärjestelmiin ja niihin liittyviin asioihin, auttaisi se näkemään web-suunnittelua toiseltakin kantilta ja täydentäisi koulutuksen aikana opittuja asioita. Tämän takia ehkä yksi itselleni asettama, ei niin virallinen, tavoite oli oppia käyttämään Joomla!':a niin hyvin, että voisin hyödyntää sitä tietoa ja osaamista myöhemminkin.

Lähteinä työssäni olen käyttänyt niin painettua kirjallisuutta kuin myös verkosta löytyvää aineistoa. Kirjallinen lähdeaineisto oli aluksi hankalaa löytää, mutta vähitellen oikeiden kanavien löydyttyä sitä kertyi. Käytetty kirjallisuus on mielestäni aiheeseen soveltuvaa ja toimivaa. Työn aiheen takia verkkolähteet ovat olleet ainoa mahdollisuus hankkia tietoa joistain osa-alueista. Aihe on osaltaan myös koko ajan kehittyvää, joten

verkkolähteet ovat ainoa tapa löytää tietoa asioista. Tämän takia lähdekritiikin kanssa on pitänyt olla erityisen tarkka. Lähtökohtaisesti olen pyrkinyt ottamaan mukaan verkkolähteitä, jotka omaavat selkeät ja helposti jäljitettävät lähdetiedot. Monesti löytynyt lähde on ollut riittämätön, ja sen takia on täytynyt etsiä toinen vastaava paremmilla lähdetiedoilla. Mukana ei välttämättä ole verkkolähteiden osalta kaikista tunnetuimpia, mutta mielestäni lähdekritiikki oli tarpeeksi kova, jotta laatu säilyi.

## 2 Mikä on julkaisujärjestelmä?

Tämä luku esittelee julkaisujärjestelmää yleisesti sekä yhtä sen yleisintä rakennetta. Luvussa luodaan myös katsaus järjestelmän tarkoitukseen ja käyttöön, erilaisiin järjestelmän muotoihin ja pohditaan myös järjestelmän valintaan vaikuttavia asioita.

### 2.1 Yleistä

Julkaisujärjestelmä kuuluu sisällönhallintajärjestelmiin, mutta kuten nimi hyvin kertoo, siinä on keskitytty julkaistavan materiaalin hallintaan. Julkaistava materiaali tarkoittaa jotain sellaista, mikä tulee yleiseen levitykseen, eikä pelkästään sisäiseen käyttöön. Riippuen organisaation koosta, julkaisujärjestelmässä käsiteltävän tiedon määrä voi vaihdella huomattavasti. Esimerkiksi tässä opinnäytetyössä käsiteltävän organisaation julkaisujärjestelmä käsittelee vain WWW-sivut ja niiden kautta tapahtuvan materiaalin julkaisemisen. Toisena ääripäänä julkaisujärjestelmä voisi olla nivottuna kiinni koko organisaation tietojärjestelmään, missä sen kautta hallinnoitaisiin kaikkea julkaistavaan materiaalia, oli kyseessä sitten yrityksen sisäinen tai ulkoinen julkaisu.

Kuten kerrottua, kyseessä on yksi monista eri sisällönhallintajärjestelmien tyypeistä. Muita tyyppejä ovat muun muassa dokumenttienhallintajärjestelmä, verkkokauppajärjestelmä, aineistönhallintajärjestelmä, wiki-järjestelmät ja WWW-sisällönhallintajärjestelmä. Julkaisujärjestelmä on myös synonyymi WWW-sisällönhallintajärjestelmälle tai WWW-julkaisujärjestelmälle. Tämä johtuu siitä, että julkaisu tapahtuu yleensä WWW-pohjaisen kanavan kautta. (Rockley 2003)

## 2.2 Rakenne

Jokaisella järjestelmällä on tietysti olemassa jonkinlainen rakenne ja idea siitä, miten tämä rakenne toimii. Julkaisujärjestelmät eivät ole tässä mikään poikkeus. *Boiko (2005)* esittelee vakio-ominaisuudet julkaisujärjestelmälle, jotka ovat sisällön koostaminen, sisältökokoelman hallinta ja julkaisun hallinta. *Pohjanoksa, Kuokkanen ja Raaska (2007: 212)* esittävät myös oman näkökulman siitä, minkälaisen rakenteen/toiminnot useimmat julkaisujärjestelmät omaavat. Näistä kahdesta otan lähempään tarkasteluun *Pohjanoksan ym. (2007)* rakenteen, jota on havainnollistettu kuvassa 1.



Kuva 1 Julkaisujärjestelmissä useasti esiintyvä rakenne ja sen toiminnot (Pohjanoksan ym. 2007: 212)

**Tiedon tuotantovälineet** tarkoittavat muun muassa julkaisujärjestelmän omia editoreita, joilla voi tuottaa tekstiä tai muuta julkaistavaa materiaalia. Näiden omien editorien lisäksi tuotantovälineet voivat tarkoittaa normaaleja tekstinkäsittelyohjelmia. Tässä työssä tulen tarkastelemaan Joomla!-nimeä kantavaa julkaisujärjestelmää, joka monen muun järjestelmän tapaan käyttää niin sanottua WYSIWYG-editoria, eli What You See Is What You Get, mikä tarkoittaa vapaasti suomennettuna ”Mitä näet, sitä saat”. Julkaisujärjestelmän editorin näkymä vastaa siis pääsääntöisesti sitä näkymää, joka tulee olemaan esimerkiksi yrityksen WWW-sivuilla.

Valmis dokumentti, artikkeli tai sivu, jota nyt ollaan julkaisemassa, täytyy seuraavaksi määritellä omalle paikalleen. Tässä yhteydessä käytetään sanaa sivu, kun viitataan muokattavaan sisältöön. Tavallaan tämä tapahtuma kuuluu **tiedon tuotantoon**, mutta se aloittaa jo tiedon tallennustapahtumaa. **Tallennusjärjestelmänä** toimii järjestelmän oma tietokanta, johon julkaistavat sivut tallentuvat. Tietokannasta käsin sivuja hallinoidaan erilaisten **julkaisumenetelmien** avulla. Jokaisella sivulle ja sivuston osalle voidaan määritellä julkaisuehtoja. Näitä ehtoja ovat muun muassa päivä, kellonaika, kanava (intra, WWW yms.).

**Versionhallinta** näyttelee tärkeää osaa, kun halutaan vaikka julkaista uudelleen vanhaa tietoa, joka on päivitetty. Versionhallinnan avulla jo julkaistuista sivuista on tallessa edelliset versiot. Eri versioiden tarkastelu sidottuina eri ajanjaksoihin on myös mahdollista.

**Käyttöoikeuksien hallinta** on työkalu, jota yleensä järjestelmän ylläpitäjä eli järjestelmänvalvoja hallinnoi. Käyttöoikeuksien avulla määritellään, mitä oikeuksia kullakin käyttäjällä on. Järjestelmällä voi olla esimerkiksi kaksi järjestelmänvalvojaa, joilla on kaikki oikeudet, ja tavallisilla käyttäjillä on oikeus esimerkiksi lisätä uusia sivuja tai muokata valmiita.

Joomla! on rakenteeltaan ja toiminnoiltaan myös yllä olevaa kaavaa noudattava. Koska Joomla! on avoimen lähdekoodin alainen ohjelmisto, on siihen saatavilla paljon erilaisia laajennuksia ja lisäosia, mitkä tietysti sekoittavat tätä rakennetta jonkin verran.

## 2.3 Tarkoitus ja käyttö

Monesti tarve tämän kaltaiselle järjestelmälle syntyy tietysti halusta organisoida yrityksen tai jonkun muun keskittymän julkaisutarpeita. Hankkimalla järjestelmä, joka keskittää kaiken julkaistavan materiaalin, tuo huomattavaa selkeyttä varsinkin silloin, kun käyttäjiä on monia satoja. Tietysti selkeytyminen on huomattavissa jo muutaman henkilön käytössäkin. Yksinkertaisesti selitettynä julkaisujärjestelmän avulla on tarkoi-

tus hallinnoida ja julkaista esimerkiksi dokumentteja Internet- tai intranet-palvelussa. Riippuen palvelun koosta, voi julkaisu- ja sisällöntuotantojärjestelmä olla eriytetty toisistaan. Tällaisella ratkaisun avulla taattaisiin hyvä suorituskyky useammalle käyttäjälle. Tällöin myös sisällöntuotanto tapahtuisi luultavammin useammalla järjestelmällä ja julkaisujärjestelmänä toimisi yksittäinen ohjelma. Yleensä kuitenkin sisällöntuotanto ja julkaisu tapahtuvat samalla järjestelmällä. *Pohjanoksa ym. (2007: 210)*

## 2.4 Julkaisujärjestelmien eri muodot

Erilaiset julkaisujärjestelmät voidaan jakaa kolmeen eri ryhmään niiden ominaisuuksien ja kattavuuden perusteella. Jaottelu voidaan tehdä yksinkertaisesti raskaisiin, keskiraskaisiin ja kevyisiin järjestelmiin. Tästä jaottelusta erillään ovat vielä Open Source eli avoimen lähdekoodin järjestelmät. Seuraavan neljän otsikon alla käymme läpi nämä ryhmät.

### 2.4.1 Raskaat järjestelmät

Kookkaimmat ja raskaimmat järjestelmät on tietenkin suunniteltu isoille organisaatioille, joilla käyttäjien määrä on moninkertainen verrattuna muihin organisaatioihin. Käyttäjien määrä kohoaa satoihin, ellei jopa tuhansiin. Myös palveluiden laajuus on aivan omaa luokkaansa verrattuna pienempiin organisaatioihin.

Järjestelmiä voidaan käyttää vain julkaisemiseen, mutta niissä on yleensä työkalut, jotka mahdollistavat sisällönhallinnan tuottamisen ja prosessien ohjauksen. Isoissa organisaatioissa julkaisujärjestelmät ovat myös yleensä kytköksissä muihin tietojärjestelmiin ja tämän ominaisuuden avulla on mahdollista hallita laajoja portaaleja.

Isot julkaisujärjestelmät ovat yleensä suurten tietotekniikkayhtiöiden julkaisemia ja siksi niiden tuki- ja kehittämispalvelut ovat hyvät ja laajat. Kattava julkaisujärjestelmä tuo mukanaan myös ison lisenssinmaksun. Tämän tason järjestelmien lisenssien lähtöhinnat ovat 100-300 000€-luokkaa. (*Pohjanoksa ym. 2007: 214.*)

#### 2.4.2 Keskiraskaat järjestelmät

Keskiraskaat järjestelmät on suunnattu isoille organisaatioille tai niiden osille. Nämä keskiraskaat järjestelmät on rakennettu hallitsemaan hieman erilaisia kokonaisuuksia kuin isot järjestelmät. Keskiraskaiden järjestelmien painotus on enemmän suorassa WWW-sisällönhallinnassa ja julkaisussa. Niiden käyttötarkoitus ei ole esimerkiksi laajojen portaalien, joissa on integroituna suuri osa organisaation tietojärjestelmistä, hallinta. Käyttäjämäärätkin ovat huomattavasti pienempiä, vaikka joissakin tapauksissa voidaan vielä puhua useista sadoista.

Lisenssimaksuissa tullaan myös alaspäin ja hinnoittelu on 20-100 000€-välillä. Koska on kyse järjestelmistä, joissa perustoiminnot eivät ole niin laajoja kuin ison luokan järjestelmissä, on usein välttämätöntä suorittaa räätälöinti- ja käyttöönottoprojekti. Tämän avulla varmistetaan oikeanlaisen järjestelmän hankinta ja vältytään näin maksamasta turhasta. (*Pohjanoksa ym. 2007: 214 -215.*)

#### 2.4.3 Kevyet ja ultrakevyet järjestelmät

Kevyiden järjestelmien alaisuuteen kuuluvat tietysti pk-yritykset, pienet organisaatiot tai isojen organisaatioiden yksittäiset osastot. Kevyissä järjestelmissä ominaisuudet ovat usein hyvät ja riittävät, mutta niiden tekninen toteutus ei yllä raskaiden tai keskiraskaiden järjestelmien tasolle, eli ne eivät kykene käsittelemään ja hallitsemaan samanlaista tietomäärää kuin raskaammat järjestelmät. Eikä niitä siihen ole suunniteltu. Tästä johtuen lisenssihinnoittelukin on tietysti huomattavasti huokeampi. Lisenssien keskitason hinnat liikkuvat 5-10 000€-välillä. Kevyiden järjestelmien käyttöönotto

ja hallinta on yleensä helpompaa kuin raskaiden. Tämä johtuu siitä, että niiden kytkökset muihin organisaation tietojärjestelmiin ovat vähäisiä tai niitä ei ole ollenkaan.

Ultrakevyet järjestelmät eroavat kevyistä siinä, että yleensä ne toimivat sovellusvuokraus menetelmän kautta. Sovellusvuokrauksessa on tarkoitus toimittaa palvelun tilaajalle valmis ympäristö, jossa järjestelmä toimii. Tämä kattaa muun muassa palvelinjärjestelmät ja ylläpidon. Lisenssimaksut näissä ultrakevyissä ratkaisuihin ovat huomattavasti alhaisemmat kuin missään muissa edellä mainituissa. Maksut ovat yleensä muutamana kymmenenä euroa kuukaudessa. (*Pohjanoksa ym. 2007: 215 -216.*)

#### 2.4.4 Open Source järjestelmät

Viimeisimpänä julkaisujärjestelmä muotona ovat avoimen lähdekoodin (engl. Open Source) alaiset järjestelmät. Avoimen lähdekoodin ratkaisut ovat vapaassa käytössä ja niihin ei kuulu lisenssimaksuja. Tekniseltä toteutukseltaan parhaimmat avoimen lähdekoodin järjestelmät yltyvät keskiraskaiden kaupallisten järjestelmien tasolle. Tuki- ja ylläpitopalvelut vaihtelevat avoimen lähdekoodin ratkaisuihin. Kaupallisia tukipalveluita on saatavilla, joillekin avoimen lähdekoodin järjestelmille, mutta ei kaikille. Normaalisti järjestelmiä kehittää vapaaehtoiset yhteisöt ja järjestöt. Avoimen lähdekoodin alaiset julkaisujärjestelmät ovat suositeltavia sellaisille organisaatioille, joiden tietotekninen osaaminen ja resurssit mahdollistavat toimintaympäristön muokkaamisen niin, että nämä ratkaisut toimivat. (*Pohjanoksa ym. 2007: 216.*)

## 3 Julkaisujärjestelmän valinta

Järjestelmän valinta ei ole mikään helppo prosessi, koska vaihtoehtoja on niin paljon. Asioita, jotka tulee ottaa huomioon on paljon, ja monessa pienessäkin asiassa voi mennä vikaan, jos valintaan ei paneuduta kunnolla. Tärkeimpiä kriteereitä ovat tekniset asiat, mutta tulee myös miettiä sitä, miten ohjelmaa mahdollisesti haluttaisiin käyttää.

### 3.1 Julkaisujärjestelmän valintakriteerit

Kun organisaatio on päättänyt ottaa käyttöön julkaisujärjestelmän, on tehty ensimmäinen valinta monien joukossa, eli on päätetty siirtyä niin sanotuista staattisista web-sivuista dynaamisiin. Ympäristössä, jossa käyttäjiä on monia, on hyvä olla yhteinen ohjelma, jossa esimerkiksi sivujen sisällön päivitys ei ole kiinni yhdestä henkilöstä, vaan siihen on oikeudet usealla henkilöllä. Staattisissa sivuissa on yleensä se ongelma, että sivuilla on vain yksi ylläpitäjä, joka vastaa sisällön päivittämisestä. Tämä oli myös yksi niistä syistä, miksi Veljekset Keskinen päätyi käyttämään julkaisujärjestelmää ja sitä kautta Joomla!:a. Muita Joomla!:n valintaan vaikuttaneita asioita ja ominaisuuksia olivat muun muassa aktiivinen kehitys, laaja käyttäjäkunta, laajennukset, LAMP (katso 3.2 Käyttöympäristön tekninen määrittely) ja käyttömaksut. Joomla! oli myös etukäteen tuttu Veljekset Keskiselle. Käytössä oli ollut jo Joomla!:n aiempi versio 1.0 ja nyt oli kyse versiosta 1.5. Vaikka kyseessä oli ennestään tuttu ohjelma, ei se kuitenkaan automaattisesti tarkoittanut, että siirryttäisiin käyttämään Joomla!:n uusinta versiota.

Avoimen lähdekoodin alaiset ratkaisut ovat tunnettuja siitä, että niiden käyttäjäkunta on usein erittäin aktiivista ja innokasta. Verrattuna kaupallisiin, avoimen koodin ohjelmien tuki on erilaista, vaikkakin suurimmilla avoimilla ohjelmistoilla saattaa olla myös kaupalliset tukipalvelut. Yleensä kuitenkin tukea ja opastusta saa yhteisön aktiivisilta ja asiantuntevilta kehittäjiltä ja käyttäjiltä suoraan ja monesti voi olla, että korjaus ongelmaan tulee saman tien päivityksen tai korjauksen muodossa. Tämä ei päde tietysti kaikkiin ohjelmiin, mutta yleisesti voi sanoa avoimen lähdekoodin ohjelmisto-

jen käyttäjien ja kehittäjien tekevän paljon tiiviimpää yhteistyötä kuin kaupallisissa ohjelmistoissa. Tärkeänä asiana voi pitää myös sitä, että erityisesti turvallisuusaukot korjataan nopeasti Joomla!-ssa, koska kehitys on niin tiivistä.

Laajennukset ovat myös osaksi laajan ja aktiivisen käyttäjäkunnan aikaansaannosta. Kuten Joomla!, niin myös sen laajennukset, ovat täysin ilmaisia. Tämä mahdollistaa erilaisten ratkaisujen kokeilemisen ja käyttämisen ilman mitään maksuja ja sitoumuksia. Osaavan käyttäjän käsissä myös omat laajennukset syntyvät kohtuullisen helposti. Kuten sanottua, tämä vaatii jo hieman enemmän osaamista.

Aikaisemmin kävimme läpi Veljekset Keskinen Oy:n julkaisujärjestelmän valintakriteereitä. *Pohjanoksan ym. (2007: 218 -219)* mukaan kriteerejä voidaan jaotella seuraaviin kategorioihin:

- toiminnallinen riittävyys
- kehitys ja jatkuvuus
- hinta ja lisensointimallit.

Veljekset Keskinen kriteerit voidaan suoraan sijoittaa näihin kategorioihin. Katteoria Toiminnallinen riittävyys sisältää muun muassa toimintaympäristön, johon LAMP kuuluu. Katteoria Kehitys ja jatkuvuus, sisältää selvästi aktiivisen kehityksen, laajan käyttäjäkunnan ja laajennukset. Ilmaiset käyttömaksut kuuluvat tietysti katteoriaan hinta ja lisensointimallit. Tästä jaottelusta voi havaita, että valintakriteerit on käyty suhteellisen tarkkaan läpi ja omia vaatimuksia ja tarpeita on mietitty tarkasti.

Tärkeimpänä kriteerinä voidaan kuitenkin pitää sitä, miten hyvin hankinnassa oleva ohjelmisto tulee soveltumaan jo ennestään olemassa oleviin järjestelmiin. Jos yhden ohjelman takia joudutaan hankkimaan viisi muuta ohjelmaa tai ratkaisua tulee silloin miettiä, onko hankinta oikea. Tässäkin asiassa riippuu taas paljon siitä, millainen osaaminen ohjelman hankkijalla on omasta takaa, ja kuinka paljon pitää turvautua ulkopuoliseen apuun. Tästä asiasta ja sen määrittelystä kerrotaan enemmän seuraavassa kappaleessa Käyttöympäristön tekninen määrittely.

### 3.2 Käyttöympäristön tekninen määrittely

Normaalisti, kun hankintaan uutta ohjelmistoa, on mukana kaksi osapuolta; toimittaja ja tilaaja. Tilaajan tehtävänä on selvittää, mitä tarpeita ja vaatimuksia on, ja toimittajan taas selvittää tilaajalle, mitä heidän tuotteellaan on tarjota heille. Samassa myös selvitetään, kuinka hyvin tarjolla oleva tuote soveltuu tilaajan vanhaan järjestelmään ja ympäristöön, eli tekninen puoli tulee ottaa tarkasti huomioon. Teknisen määrittelyn hoitaa aina itse tilaaja, jos vain suinkin mahdollista.

Tämän voisi olettaa olevan normaali lähtötilanne. Joskus kuitenkin, kuten Veljekset Keski-Keski-Keskisen kohdalla, tilanne on se, että tieto taito ja osaaminen ovat sillä tasolla, että toimittajaa ei varsinaisesti tarvita. Omat tarpeet ja vaatimukset, niin tekniset kuin muutkin, tiedetään, ja tällöin otetaan itse selvyys vaihtoehdoista. Toimeksiantajan tapauksessa tämä onnistui myös sen takia, että niiden henkilöiden lukumäärä, jotka olivat mukana julkaisujärjestelmän valinnassa, oli hyvin pieni. Kun muutama henkilö pystyy hallitsemaan koko kokonaisuuden, on esimerkiksi teknisen määrittelyn tekeminen ja ymmärtäminen paljon helpompaa.

Tehtäessä teknistä vaatimusmäärittelyä isommassa organisaatiossa, tapahtuu määrittelytyöryhmätyöskentelyn kautta tai ainakin kokouksia asian tiimoilta tulisi pitää paljon, jotta kaikki osa-alueet tulisi käsiteltyä hyvin. Tavoitteena näillä kokouksilla ja ryhmätyöskentelyillä on teknisen vaatimusmäärittelydokumentin tekeminen. *Pohjanoksan ym. (2007: 217)* mukaan tekninen vaatimusmäärittelydokumentti sisältää yleensä seuraavat asiat:

- verkkopalvelun yleiskuvaus
- palvelun rakenne ja toiminnallisuus
- tietovarastojen yleiskuvaus
- teknisten ratkaisujen yleiskuvaus
- rajaukset.

Tämän dokumentin avulla isojen organisaatioiden on helpompi pysyä samassa linjassa eri ryhmien välillä. Tämän kaltaista dokumenttia ei Veljekset Keski-Keski-Keskisen tapauksessa

tehty johtuen siitä, että julkaisujärjestelmän valinnassa olleiden henkilöiden lukumäärä oli niin pieni. Joomla!:n tekniset vaatimukset ovat myös hyvin vaatimattomat ja sen sulauttaminen jo olevassa olevaan toimintaympäristöön on suhteellisen vaivaton tehtävä. Toimeksiantajan tapauksessa tarvittava ympäristö oli jo suoraan saatavilla.

Joomla!:n järjestelmävaatimukset ovat PHP-ohjelmointikielentuki, Apache WWW-palvelin ja MySQL-tietokanta. Veljekset Keskisellä Joomla! toimii jo ennestään tutun LAMP-ympäristön päällä. LAMP lyhenne tulee sanoista Linux, Apache, MySQL ja PHP. LAMP-ympäristö sisältää siis kaiken, mitä Joomla:n järjestelmävaatimukseen kuuluu. LAMP on siis tarkoitettu Linux-ympäristöön, mutta olemassa on myös vastaava Windowsille ja sen nimi on WAMP. Jos LAMP:ssa L-kirjain tarkoittaa Linuxia niin tietysti WAMP:ssa se tarkoittaa Windowsia. Toinen samanlainen Windowsille tarkoitettu on nimeltään WIMP. Tässä versiossa Apache-palvelin on vaihdettu IIS-nimiseen palvelimeen.

MySQL:n ja Apachen yhdistelmä on todettu toimivaksi kaikilla yleisillä käyttöjärjestelmillä Joomla!:n kanssa. PHP vaatii muutaman lisäosan, jotka tulee aktivoida, jotta se toimisi Joomla!:n kanssa. Nämä lisäosat ovat Mod-etuliitteisiä. Taulukossa 1 nämä vähimmäisvaatimukset on eriteltyinä tarkemmin.

**Taulukko 1: Joomla! 1.5 vähimmäisvaatimukset (Wallace 2007: 2)**

Ohjelmisto	Vähimmäisversio	Suositteltu	Optimaalinen	Kielletty
PHP	4.3.x	4.4.7	5.0.x versiot	4.3.9, 4.4.2 , 5.0.4
MySQL	3.23.x tai uudempi	4.1.22	5.0.x versiot	
Apache	1.3 tai uudempi	1.3.9	2.0.61	
Mod_mysql				
Mod_xml				
Mod_zlib				

### 3.3 Vaihtoehdot Joomla!:n lisäksi

Joomla! ei ollut ainoa vaihtoehto Veljekset Keskinen julkaisujärjestelmäksi. Alkuun oli kuitenkin selvää, että kaikkein kalleimmat vaihtoehdot rajataan pois ja keskitytään halvempiin. Veljekset Keskinen Oy on muutenkin IT-ratkaisuissaan keskittynyt enemmän avoimen lähdekoodin ratkaisuihin, ja se vaikutti osaltaan Joomla!:n valintaan. Lisk CMS ja Drupal olivat kuitenkin kaksi muuta harkittua julkaisujärjestelmää. Seuraavassa kappaleissa on esittely molemmista vaihtoehdoista.

#### ***Lisk CMS***

Lisk CMS:n kotisivuilla esitellään ohjelmiston uusinta versiota seuraavasti. Lisk CMS on monipuolinen ja käyttäjäystävällinen sisällönhallintaohjelmisto, jonka on luonut Createch Group -niminen yritys. Ensimmäinen versio julkaistiin lokakuussa 2001 ja nyt on vuorossa Lisk CMS v.4.4, joka sisältää yrityksemme viiden vuoden laajan kokemuksen korkeatasoisten sisällönhallinta ratkaisujen tarjoajana. (*Greatch Group: 2008*)

Lisk CMS täyttää kaikki nykypäivän markkinoiden vaatimukset, joita ovat käyttäjäystävällisyys, uusimpien web-teknologioiden käyttö, hyvä näkyvyys hakukone hauissa, kyky toimia saumattomasti kolmannen osapuolen ohjelmistojen kanssa, turvallisuus ja tuki. (*Greatch Group: 2008*)

Lisk CMS on täydellinen ratkaisu monelle Internet-sivulle, kuten yritys, e-kauppa, yhteisö ja intra-/ extranet/ -järjestelmille. Johtuen Lisk CMS Architecturen ja Lisk Frameworkin tarjoamista mahdollisuuksista, ei Lisk CMS:ä tulisi pitää pelkästään valmiina pakettiratkaisuna. Lisk CMS:n avulla voidaan ottaa huomioon kaikki sivuston näkökulmat ja vaatimukset ja lopuksi tarjota räätälöity ratkaisu, joka sopii täydellisesti asiakkaan tarpeisiin ja tavoitteisiin. (*Greatch Group: 2008*)

## **Drupal**

Drupalista kerrotaan sen kotisivuilla seuraavanlaisesti. Drupal on ilmainen ohjelmisto, joka sallii yksilön tai yhteisön helposti julkaista, hallinnoida ja järjestää suuren määrän web-sivuston sisältöä. Kymmenet tuhannet ihmiset ja organisaatiot käyttävät Drupalia antaakseen toiminnallisuuden monille erilaisille web-sivustoille, mukaan lukien yhteisöjen portaalit, keskustelupalstat, yrityssivut, intranetit, henkilökohtaiset web-sivut ja blogit, harrastajasivut ja e-kaupat. (*Buytaert: 2008*)

Drupal on valmis käytettäväksi siitä hetkestä, kun sen lataat. Siinä on jopa helppokäyttöinen web-asennus. Sisäänrakennettu toiminnallisuus yhdistettynä useisiin vapaasti ladattaviin lisättäviin moduuleihin mahdollistaa muun muassa seuraavanlaiset toiminnot: sisällönhallinta, blogit, yhteiskäyttöiset, foorumit, peer to peer-verkot, uutiskirjeet, podcastaus, kuvagalleriat, tiedoston lataus ja tallennus sekä paljon muuta. Drupal on avoimen lähdekoodin ohjelmisto, jota jaetaan GPL-lisenssin alaisena ja sitä ylläpitää ja kehittää tuhansien ihmisten yhteisö. (*Buytaert: 2008*)

### 3.4 Yhteenveto

Kuten Lisk CMS:n tekstistä hyvin huomaa, on siinä ajatuksena myydä tuotetta. Tekstin mukaan Lisk CMS olisi täydellinen valinta kaikille, koska eihän siitä näytä puuttuvan mitään, ja se soveltuu kaikille ja mihin vain. Tämä ei tietysti pidä paikkaansa, ja siksi eri ohjelmistojen valinta on aina vaikeaa, oli kyse sitten sisällönhallinnasta tai jostain muusta. Lisk CMS ehkä suurin taakka verrattuna Joomla!:an on se, että se on maksullinen. Monesta lisäominaisuudesta joutuu maksamaan, vaikka jotkut ominaisuudet löytyvät vakiona, mitä Joomla!:ssa ei ole, tai ne pitää itse lisätä. Kaiken kaikkiaan hinta/laatusuhde on Joomla!:n puolella.

Drupal on taas hinta-politiikassa samalla linjalla Joomla!:n kanssa, eli se on ilmainen avoimen lähdekoodin ohjelmisto. Toiminnoiltaan Drupal on Joomla!:n kanssa hyvin

samalla tasolla. Yleensä molemmista löytyy samat joko valmiina tai sitten itse lisättävien lisäosien muodossa. Voisi ehkä sanoa, että vakiona Drupalissa on enemmän toimintoja kuin Joomla!:ssa. Veljekset Keskinen kannalta valinta kääntyi kuitenkin Joomla!:n puoleen sen vuoksi, että se oli osaksi jo tuttu ja toisaalta muut vaihtoehdot eivät tarjonneet joko tarvittavia ominaisuuksia tai parempaa käytettävyyttä. Liitteessä 2 on vertailutaulukko näistä kolmesta. Vertailutaulukosta näkee eri ominaisuuksien ja osien saatavuuden, ja ovatko ne ilmaisia vai maksullisia.

### 3.5 Joomla!:n ominaisuudet

Lopuksi tarkastellaan vielä valitun Joomla!-julkaisujärjestelmän vahvuuksia ja heikkouksia. Tarkoituksena ei ole tehdä mitään syvällistä tarkastelua, vaan tuoda esille yleisimmät asiat.

#### **Vahvuudet**

Joomla!:a ei esitellä Drupalin ja Lisk CMS:n tavoin, vaan tässä yksinkertaisesti listaan vain sen vahvuudet ja pohdin sen heikkouksia.

- Helppo asennus.
- Joomla!:ssa on laaja käyttäjäkunta ja tiivis kehitys.
- Mahdollisuus muokata, kirjoittaa ja julkaista artikkeleita missä tahansa pelkän Internet-selaimen avulla. Ei sidottu paikkaan eikä aikaan.
- Vastuu sivuista ja niiden päivityksestä ei tarvitse olla yhden henkilön alaisuudessa. Useammat henkilöt voivat osallistua ylläpitoon ja eri henkilöroolien määrittäminen on helppoa.
- Artikkelien määrää ei ole millään tavalla rajoitettu ja niiden arkistointi on myös mahdollista.
- Hakujen kohdistaminen on myös saatavilla.

- Erilaisten lisäosien saatavuus on hyvä ja niiden asennus helppoa. Esimerkiksi suositut keskustelualueet ovat saatavilla.
- Käyttäjällä on mahdollisuus luoda sivusto ilman mitään osaamista HTML-kielestä tai muusta ohjelmointikielestä.
- Joomla!:n yleisvahvuus on sen käytön helppous. Tänä päivänä toimivien sivujen tekeminen ja ylläpito eivät saa enää olla vain taitavien ohjelmoijien tai muuten tietotekniikkaa osaavien harteilla. Usein voi olla, että ylläpito on pelkästään esimerkiksi yrityksen mainososaston varassa tai ainakin sisällöllinen.

### **Heikkoudet**

Varsinaisia heikkouksia on varmasti yhtä monta kuin on mielipiteitä siitä, mikä on paras julkaisujärjestelmä. Joomla! ei varmasti pärjää kaikilla tasoilla niin hyvin kuin jokin kaupallinen järjestelmä, mutta ei sen ole tarkoituskaan. Se ei myöskään ole avoimen lähdekoodin alaisista järjestelmistä ylivoimaisiin, koska ei voida määritellä, mitkä ominaisuudet tekevät toisesta järjestelmästä paremman kuin toisesta. Sopivuutta voidaan luokitella, mutta mitään yleispätevää tutkimusta siitä, mikä järjestelmä sopii kellekin, ei voi tehdä. Yleensä heikkoudet ovat paikattavissa ja niiden tuoma harmi ei ylitä sitä hyötyä, jonka järjestelmä käyttäjälle tuo. Tämä tietysti silloin, kun käyttäjä on käyttänyt edes jonkin verran aikaa päätöksen tekemiseen.

Joomla!:n kaltaisessa kehittyvässä projektissa sen voisi sanoa olevan niin heikko kuin käyttäjä siitä tekee. Täysin valmista pakettia Joomla! ei heti tarjoa, mutta pienellä välinvälillä siitä saa erittäin helppokäyttöisen ja monipuolisen julkaisujärjestelmän.

## 4 Julkaisujärjestelmän käyttöönottoprosessi

Tämä luku kertoo itse julkaisujärjestelmän käyttöönotosta. Luvussa käydään läpi julkaisujärjestelmän asennus, räätälöiminen sekä ylläpidolliset asiat.

### 4.1 Joomla!-n hankinta ja asennus

Fyysinen käyttöönotto lähtee liikkeelle ohjelmiston asennuksesta. Joomla! on mahdollista asentaa paikallisesti tai palvelimelle. Paikallinen asennus toimii loistavasti, jos haluaa esimerkiksi luoda testiympäristön, jossa kokeilla erilaisia vaihtoehtoja sivustolle. Näin esimerkiksi, silloin jos yrityksellä on jo sivusto ja siihen halutaan tehdä jokin isompi muutos, joka vaatisi normaalisti sivuston laittamisen offline-tilaan, jotta uudistus saataisiin kokeiltua. Paikallinen asennus sivustosta esimerkiksi testikoneelle mahdollistaa tällaisessa tilanteessa testauksen ilman, että oikeaa sivustoa tarvitsee laittaa offline-tilaan.

Kuten teknisessä määrittelyssä kerrotaan, Joomla! vaatii toimiakseen LAMP-ympäristön. Kokemattomalle LAMP-ympäristön asentaminen voi olla vaikeaa ja aikaa vievää. Tähänkin asiaa on saatavilla erilaisia valmiita asennuspaketteja, joilla vaadittavien ohjelmien asennus tapahtuu yhdellä kertaa. Juuri nämä paketit ovat loistavia, kun on tarpeen asentaa testiympäristö yhdelle koneelle, eikä halua koskea itse tuotanto- eli palvelinympäristöön.

Koska Joomla! on avoimen lähdekoodin alainen ohjelma, ja näin ollen se ei tarvitse lisenssiä, voi sen ladata suoraan osoitteesta <http://www.joomla.org/>. Varsinaista asennusta en käy läpi vaihe vaiheelta, koska siihen on olemassa jo valmiit oppaat ja niiden kopioiminen ei toisi mukanaan mitään lisäarvoa. Mitä kuitenkin asennuksesta on hyvä tietää, ovat seuraavat asiat. Jos ollaan asentamassa Joomla!-a ensimmäistä kertaa, kannattaa se asentaa ensin kokeiluksi vain paikallisesti. Toiseksi, Joomla! vaatii toimiak-

seen esimerkiksi LAMP-ympäristön.

Veljekset Keskisellä Joomla! on käytössä kahdella palvelimella. Toinen on varsinainen tuotantopalvelin, jossa ovat muun muassa yrityksen omat kotisivut ja toinen on niin sanottu testipalvelin. Testipalvelimella on muun muassa aina Joomla!:n uusin versio, jotta sitä voidaan saman tien testata ja nähdä, miten eri muutokset vaikuttavat käyttöön. Joomla!:sta on vielä yleensä myös paikallinen asennus ainakin yhdellä koneella. Kätevän sivupohjan siirron ja uudelleenasetuksen ansiosta näiden eri versioiden välillä testailu ei ole kovin vaikeaa, mikä mahdollistaa näin ollen mahdollisten virheiden havaitsemisen helposti.

## 4.2 Julkaisujärjestelmän räätälöiminen

Joomla!:n perusasennus tuo mukanaan hyvät ominaisuudet, joilla pääsee hyvin alkuun sivujen luonnissa. Silloin kun on tarve lisätä uusia ominaisuuksia sivustoon, on yleensä turvaututtava julkaisujärjestelmän räätälöintiin. Räätälöintiä voi suorittaa esimerkiksi erilaisilla liitännäisillä, moduuleilla ja laajennuksilla. En käy läpi näitä kaikkia, mutta otan tarkasteluun perinteisimmät ja myös niitä, jotka ovat käytössä Veljekset Keskisen julkaisujärjestelmässä.

### 4.2.1 Liitännäiset ja moduulit

Liitännäiset voivat tuoda monenlaisia eri toimintoja sivuille. Liitännäiset voivat kuulua muun muassa sivujen todentamiseen, sisältöön ja hakutoimintoihin. Myös jotkut editorit ovat liitännäisiä.

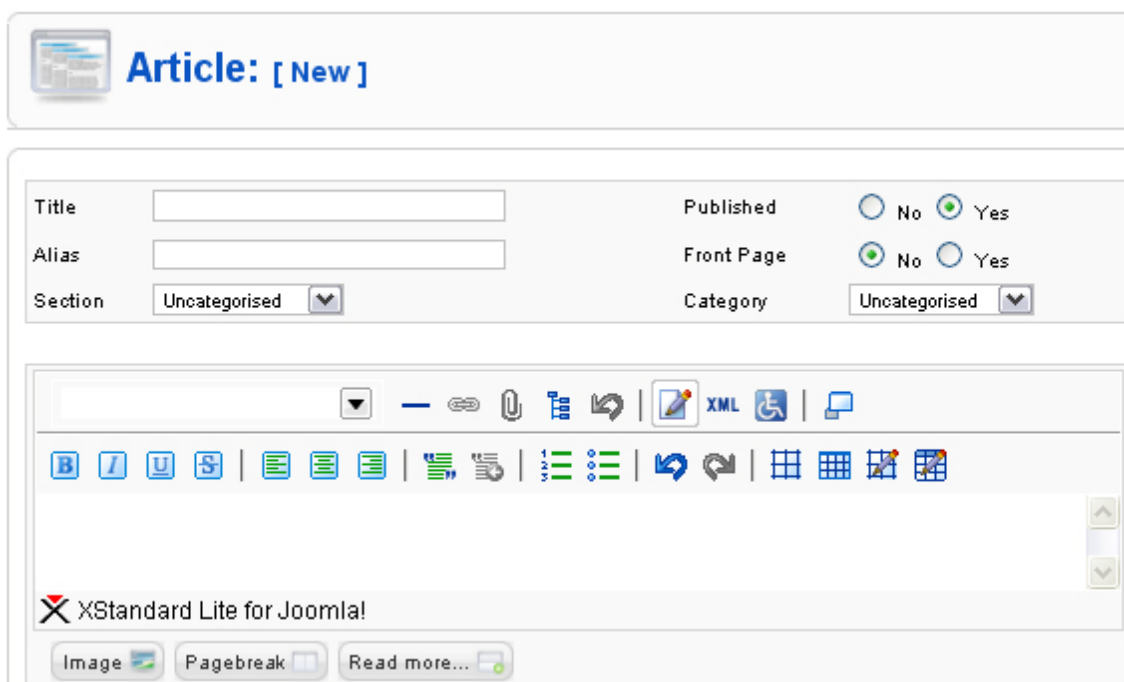
Lähdetään liikkeelle parilla liitännäisellä, jotka tulevat Joomla!:n mukana. Nämä ovat nimeltään TinyMCE ja XStandard Lite for Joomla!. Molemmat näistä liitännäisistä

ovat niin sanottuja tekstieditoreita ja molemmat ovat tyypiltään WYSIWYG-editoreita. Näiden editoreiden avulla voidaan luoda ja muokata sisältöä sivuille. TinyMCE on Joomla!-ssa oletuseditorina ja toiminnoiltaan se on myös laajin. TinyMCE sisältää kaikki normaaliin käyttöön tarvittavat ominaisuudet. Veljekset Keskinen käyttää myös oletuseditorinaan TinyMCE-editoria.

Yleensä näiden kahden peruseditorin lisäksi Joomla!-n perusasennuksessa on mukana kolme kappaletta niin sanottuja editoripainikkeita. Image, Readmore ja Pagebreak ovat näiden painikkeiden nimet. Näille on myös oma liitännäistyypinsä, ja se on editors-xtd. Painikkeiden käyttötarkoitukset ovat seuraavat:

- Image = kuvan lisäys artikkelin tekstiin
- readmore = Jakaa tekstin esikatseluun ja itse tekstiin
- pagebreak = Tekee sivunvaihdon tekstiin.

Kuvat 2 ja 3 ovat Joomla!-n editointinäkymät XStandard Lite for Joomla!- ja TinyMCE-editoreista. Näissä kuvien alareunassa näkyvät myös Image-, Pagebreak- ja Read More-painikkeet.



**Kuva 2: XStandard Lite for Joomla! editorinäkymä**

The image shows the TinyMCE editor interface for creating a new article. At the top, there is a header 'Article: [ New ]'. Below this, there is a form with several fields and options:

- Title:** A text input field.
- Alias:** A text input field.
- Section:** A dropdown menu with the option '- Select Section -'.
- Published:** Radio buttons for 'No' and 'Yes', with 'Yes' selected.
- Front Page:** Radio buttons for 'No' and 'Yes', with 'No' selected.
- Category:** A dropdown menu with the option 'Select Category'.

Below the form is a rich text editor toolbar with various icons for bold, italic, underline, text color, background color, bulleted list, numbered list, link, unlink, insert link, insert image, HTML, undo, redo, and other editing functions. Below the toolbar is a large text area for writing the article content. At the bottom of the editor, there are three buttons: 'Image', 'Pagebreak', and 'Read more...'.

**Kuva 3: TinyMCE-editorinäkömää**

Kuten kuvista näkyy, TinyMCE:n artikkelin muokkaus mahdollisuudet ovat monipuolisemmat kuin XStandard Lite for Joomla!-ssa.

Liitännäisten lisäksi lisäominaisuuksia Joomla!-an tuovat moduulit. Yksinkertaisimmillaan moduulit ovat sivuston osia, jotka sisältävät tietoa. Esimerkiksi jokin valikko voidaan määrittellä moduuliksi. Tarkastellaan vielä paremmin tätä menu-esimerkkiä. Sivupohjan index.php-tiedostossa (Liite 4) on määritelty moduulin paikka seuraavasti: `<jdoc:include type="modules" name="user3" />`. Tälle moduulille on annettu nimi "user3". Tämä nimi vastaa Joomla!-n moduulinhallinnassa valitun moduulin asetuksien position-kohtaa. Nyt tämän nimen avulla (user3) voi kohdistaa sen sisällön, mitä haluaa siinä kohtaan näkyvän antamalla Joomla!-n moduulihallinnassa, valitulle moduulille position kohtaan arvon "user3". Sisältö, jonka halutaan näkyvän, voi olla esimerkiksi sivuston navigointivalikko.

Tämän lisäksi on toisenlaisia moduuleja, jotka sisältävät enemmän toiminnallisuuksia. Esimerkkinä tällaisesta on Content Item Module -niminen moduuli, joka on käytössä myös Veljekset Keskisellä. Se mahdollistaa sisällön julkaisemisen missä tahansa sivus-

toa. Käyttäjä saa melko vapaasti määritellä eri artikkelien, menujen ja muun sisältöateriaalin paikan. CIM on muokattavuutensa ansiosta erittäin käytännöllinen moduuli.

#### 4.2.2 Laajennukset

Edellisessä luvussa oli kyse joko pelkistä liitännäisistä tai moduuleista, mutta nyt esitellämme laajennukset. Laajennukset eivät sinänsä eroa liitännäisistä tai moduuleista, vaan ne sen sijaan sisältävät molempia. Laajennuksessa on kyseessä usein jokin laajempi kokonaisuus, jonka voi liittää sivuston toiminnallisuuksiin. Vaikka laajennukset yleensä sisältävät enemmän tavaraa kuin esimerkiksi pelkät liitännäiset tai moduulit, on niiden asennus ja käyttöönotto kuitenkin aivan yhtä vaivatonta.

Hyvinä esimerkkeinä laajennuksista toimivat Joom!Fish ja Jumi. Joom!Fish tuo sivustolle monikielisyyden tuen. Jumi puolestaan mahdollistaa puhtaan PHP:n käyttämisen sivustolla, vaikka artikkelien sisällä. Nämä molemmat laajennukset ovat käytössä Veljekset Keskisellä.

Isoimpien laajennuksien kehitys Joomla!:lle on erittäin hyvin organisoitu ja kehitys on jatkuvaa. Joom!Fish on hyvä esimerkki tästä. Tiedot ja tuki, mitä on saatavilla Joom!Fish:iin, ovat todella hyvää luokkaa ja asioita ei ole jätetty puolitiehen. Veljekset Keskisen kaltaisellekin yritykselle hyvin toimivat sivut ovat yksi markkinoinnin lähde ja tämän kaltaiset hyvin dokumentoidut ja kehitetyt apuvälineet mahdollistavat avoimen lähdekoodin alaisten ohjelmien käytön kalliiden kaupallisten sijaan.

#### 4.3 Yhteensopivuus ja ajantasaisuus / ylläpito

Seuraavaksi käsitellään asiat, jotka pitää ottaa huomioon sen jälkeen, kun järjestelmä on asennettu ja halutaan taata, että päivittäinen käyttö on ongelmaton ja sujuvaa.

Tekstissä keskitytään pääasiassa Joomla:n ja sen käyttöympäristön päivityksiin, mutta katsotaan myös muita satunnaisia ylläpidollisia asioita.

#### 4.3.1 Joomla!-versiot ja päivitykset

Joomla! on koko ajan kehittyvä ohjelmisto, ja sille julkaistaan päivityksiä päivittäin. Eri päivitykset ja versiot, jotka on mahdollista ladata, ovat seuraavanlaiset: Stable Download = Viimeisin vakaa julkaisu, Beta Releases = Beta julkaisu, Nightly Builds = Pika päivitykset, Subversion Access = Juuri sen hetkinen versio. (*How to get Joomla: 2008*)

Seuraavaksi käydään vielä eriteltynä läpi nämä päivitysversiot. Ensimmäisenä on Stable Download eli uusin versio, joka on testattu ja todettu toimivaksi. Tätä suositellaan normaalikäyttäjälle.

Beta Releases tarkoittaa versioita, jotka ovat vielä kehitysasteella. Beta-julkaisujen käyttämistä ei kuitenkaan varsinaisesti suositella, vaan esimerkiksi Nightly Builds -päivitykset ovat suositeltavimpia.

Nightly Builds -päivitykset ilmestyvät ladattavaksi melkein joka päivä samaan aikaan. Niissä on korjattu käyttäjien tai kehittäjien havaitsemat virheet. Näiden päivitysten tarkoitus on antaa Joomla!:n käyttäjille aina paras mahdollinen versio ohjelmistosta. Näitä päivityksiä lataavat yleensä ohjelmaa kehittävät käyttäjät, vaikka Nightly Builds -päivityksiä voi ladata normaalikin käyttäjä, mutta varmuuskopiointi tulee olla kunnossa, jos aikoo päivittää aina uusimman Nightly Build -päivityksen.

Viimeisimpänä on Subversion Access, joka sisältää juuri sen hetken version Joomla!:sta. Tätä versiota ei suositella käytettäväksi normaalikäyttäjille ja aloittelijoille. Tätä versiota käyttävät yleensä ne, jotka ovat mukana esimerkiksi projektissa, joka vaatii

kaikista uusimman version ja tietysti aktiiviset kehittäjät käyttävät näitä. Näiden versioiden kanssa käytetään myös SVN-nimistä versionhallintatyökalua, jonka tarkoituksena on mahdollistaa esimerkiksi lähdekoodin muokkaus usean kehittäjän kesken niin, että kaikkien tekemät muutokset ja kopiot pysyvät ajan tasalla. Subversion:ia levitetään Subversion Licensen alla, joka on avoimen lähdekoodin lisenssi.

#### 4.3.2 Käyttöympäristön päivitykset

Veljekset Keskisellä on käytössä siis LAMP-ympäristö, kuten aikaisemmin teknisessä määrittelyssä todettiin. Tämä ympäristö sisältää Linux-, Apache-, MySQL- ja PHP-tuen. Näitä jokaista voidaan päivittää erikseen, vaikka ne voidaan asentaa saman asennuspaketin avulla. Yleisesti voidaan sanoa, että tietoturvan kannalta näiden jokaisen komponentin päivittäminen aina, kun mahdollista, on kannattavaa. Tietysti koskaan ei kannata päivittää ennen kuin on varma, että päivityksestä ei aiheudu mitään toiminnallisia vaikeuksia.

Suurempien yritysten ei ehkä aina kannata olla etunenässä koittamassa ohjelmistojen uusimpia versioita. Tietysti ohjelmistot pyritään aina julkaisemaan markkinoille testattuina ja valmiina, mutta valitettavasti ei vain ole mahdollista saada sataprosenttisesti valmista tuotetta testauksen avulla. Voi sanoa, että lopullinen testaus tapahtuu vasta, kun ensimmäiset asiakkaat ottavat ohjelman käyttöönsä.

Käyttöympäristön tekninen määrittely (3.2), on taulukossa Joomla! 1.5 vähimmäisvaatimukset, ilmoitettu sopivimmat versiot kustakin komponentista Joomla!:lle. LAMP-ympäristössä on mukana jo suhteellisen monta ohjelmaa, jotka vaativat aika ajoin päivityksiä. Harvoin tietysti mikään näistäkään ohjelmista on niin haavoittuvainen tai huonosti toimiva, että niitä olisi aivan pakko päivittää. Kun kuitenkin sen aika tulee, on syytä olla selvillä kaikista asioista, joihin uusi versio vaikuttaa ja miten. Alla on pieni esimerkki LAMP:issa mukana olevasta Apache-palvelinohjelmistosta, ja sen päivityksestä.

Apachen uusin versionumero on nykyään 2.2.9, joka julkaistiin 13.6.2008 (*Apache Software Foundation: 2008*). Apachelle on myös useita moduuleita, ja niiden avulla toimivia ohjelmistoja, joita ei ole vielä päivitetty. Tämän takia Apachen vanhempia versioita on paljon käytössä, ja näihin vanhempiin versioihin julkaistaan tietoturvapäivityksiä. Apachen versionumeroista selviää, mikä versio on kyseessä. Kun sekundääri-  
nen versionumero on pariton, on kyseessä testiversio ja parillinen tarkoittaa vakaata versiota.

#### 4.3.3 Muut ylläpidolliset asiat

Muita Joomla!-n ylläpitoon liittyviä asioita ovat muun muassa käyttäjien valvonta ja määrittäminen. Joomla!-ssa on mahdollista lisätä, poistaa ja muokata käyttäjiä. Käyttäjille voi määrittellä oikeustasot, ja näin valvoa, kuka pääsee mihinkin osaan ohjelmaa. Tämä on erittäin käytännöllinen ominaisuus, jos ottaa huomioon erilaiset turvallisuusnäkökulmat.

Näitä oikeuksia on syytä edes jollain tasolla valvoa, ja aika ajoin määrittellä ne uudelleen. Sitä mukaan kun ihmiset kehittyvät Joomla!-n käytössä, voi heille antaa enemmän oikeuksia ja mahdollisuuksia vaikuttaa sisältöön.

## 5 Joomla! käytännössä

Tämä luvun tarkoituksena on selvittää yleisellä tasolla normaalin Internet-sivuston syntyä sekä kertoa, mitä asioita Joomla!:n kanssa tulee ottaa huomioon, jotta sillä luotu sivusto toimii oikein. Käsiteltävinä asioina ovat ulkoasu, sivupohja eli template, sisällön määrittäminen ja lopuksi esimerkit Joomla!:n käytöstä.

### 5.1 Ulkoasu

Ulkoasun luominen Joomla!:an on täysin vapaasti valittavissa. Jos päätetään rakentaa sivuston itse alusta alkaen, ei rajoituksia periaatteessa ole, mutta jos päädytään valmiiseen sivupohjaan, on sivun eri elementtien asettelussa ja muissa osissa ennalta määrätty paikat, jotka tulee ottaa huomioon, mietittäessä sivujen rakennetta ja ulkonäköä. Mikään ei toki estä valmiin sivupohjan muokkausta omiin tarkoituksiin.

Yleensä sivujen tekeminen aloitetaan luomalla niin sanottu layout eli sivun pohjapiirustus tai taitto. Periaatteessa se on valmis kuva siitä, miltä valmis sivu tulisi näyttämään. Kuva luodaan yleensä kuvankäsittelyohjelmalla, josta se sitten pilkotaan eri osiin kuten tausta, otsikko ja vaikka sivujen menu-painikkeet. Nämä kuvat (osat) asetetaan niille tarkoitettuihin paikkoihin sivustolla, jolloin sivuston ulkoasu on näkymältään samanlainen, kuin mikä se oli alkuperäisessä kuvassa eli layoutissa. Tämä vaatii onnistuakseen osaamista muun muassa HTML-kielestä (Hyper Text Markup Language) sekä CSS:stä. Seuraavaksi kerrotaan enemmän näistä kahdesta.

Nykypäivänä suurimmassa osassa sivuissa on käytössä niin sanottu ”tabelleless design” eli sivujen rakenne ja taitto ei ole tehty HTML-kielessä olevaa TABLE-elementtiä käyttäen, vaan tilalle on tullut DIV-niminen elementti ja sen rinnalle Cascading Style Sheets eli CSS-tyylit.

Näin W3C (World Wide Web consortium) määrittelee TABLE-elementin: ”The HTML table model allows authors to arrange data, text, preformatted text, images, links, forms, form fields, other tables, etc. into rows and columns of cells.” (*Raggett, Le Hors, Jacobs 1999*). Vapaasti käännettynä edellinen lainaus tarkoittaa sitä, että TABLE-elementti mahdollistaa erilaisen tiedon esittämisen taulukon muodossa, joka koostuu riveistä vaaka- ja pystytasossa.

DIV-elementti puolestaan määritetään näin: ”The div element, in conjunction with the id and class attributes, offers a generic mechanism for adding extra structure to documents. This element defines no presentational idioms on the content. Thus, authors may use this element in conjunction with style sheets, the xml:lang attribute, etc., to tailor HTML to their own needs and tastes.” (*Axelsson, Birbeck, Dubinko, Epperson, Ishikawa, McCarron, Navarro, Pemberton 2006.*) Vapaasti suomennettuna tämä tarkoittaa, että DIV-elementti yhteistyössä id- ja class-attribuuttien kanssa tarjoaa mahdollisuuden lisätä lisärakenteita dokumentteihin. Elementti ei anna mitään määrittelyjä dokumenteille, vaikka tekijät voivat käyttää tätä elementtiä yhdessä tyyli-tiedostojen kanssa räätälöidäkseen haluamansa lopputuloksen.

Loppujen lopuksi TABLE- ja DIV-elementti ovat käyttötarkoitukseltaan hyvin erilaisia. TABLE-elementti on tarkoitettu pelkästään taulukoiden tekemiseen ja esittämiseen, kun DIV-elementillä voi tehdä periaatteessa ja esittää periaatteessa mitä vaan sisältöä www-sivulla. Tietysti nämä kaksi elementtiä ovat vain pieni osa niistä elementteistä, joita HTML-kielestä on käytössä, kun sivuja tehdään, mutta merkitykseltään ne ovat suurimmat. Nykyään ei myös välttämättä puhuta enää HTML-standardista, vaan nyt on käytössä XHTML-standardi, jonka oli aluksi tarkoitus korvata HTML. Näiden kahden erojen selvittäminen ei kuitenkaan ole olennaista tämän julkaisujärjestelmän käyttöönoton kanssa, joten työssä ei paneuduta niihin sen syvemmin.

CSS-tyyli-tiedostot ovat se, jonka avulla viimeiset hiomiset tehdään sivuston ulkoasuun. CSS-tyylit määrittelevät muun muassa DIV-elementtien lopullisen sijainnin ja esimerkiksi sen, mitä grafiikkaa ne sisältävät. CSS-tiedostot yksinkertaisesti luovat sivuston ulkonäön. TABLE-elementin käytöstä ei kuitenkaan ole kokonaan luovuttu,

mutta sen käyttöä on vähennetty huomattavasti. Sitä käytetään vain taulukkomuotoisen tiedon esittämiseen. CSS-tyylien suurin hyöty on, että kun sivustolle kerran on määritelty oikeanlainen tyyli, voi sisältöä muokata vapaasti CSS-tiedoston kautta. Edellyttäen tietysti, että eri elementit on määritelty oikein koko sivuston osalta CSS-tiedostoon.

W3C määrittää CSS-tyylit seuraavasti: ”Cascading Style Sheets (CSS) is a simple mechanism for adding style (e.g. fonts, colors, spacing) to Web documents.” (*Bos: 2008*). Yksinkertaisesti käännettynä CSS:n avulla muokataan Internet-dokumenttien ulkoasua muun muassa fonttien ulkonäköä.

Edelle esiteltyjä tekniikoita apuna käyttäen voidaan luoda sivuston ulkoasu. Näillä tekniikoilla on mahdollista luoda tasoltaan monenlaisia sivustoja ja lopputulos on kiinni siitä, kuinka hyvin sivuston tekijä tai tekijät nämä taitaa.

## 5.2 Sivupohja

Sivupohja (template) on sivun aihio tai runko. Sivupohja koostuu tiedostoista, joiden avulla hallinnoidaan sivuston ulkoasua. Sivupohjaa ei voi pitää kokonaisena Internet-sivuna, eikä se myöskään ole täydellinen sivuston malli. Sivupohja on sivuston perusta, joka auttaa Joomla!:a tekemään kokonaisen sivuston. Luodakseen sivuston, joka on niin sanotusti kokonainen, toimii sivupohja yhteistyössä Joomla!:n tietokantoihin tallennetun tiedon kanssa. Sivupohjaan on siis määritelty mitä, milloinkin Joomla!:n tietokannan tiedoista sivulla näytetään. (*North 2007:2*)

Joomla!:n sivupohja koostuu siis useammasta tiedostosta. Joomla!:a pyörittävään ytimeen on määritelty seuraavat kaksi tiedostoa, joiden pitää ehdottomasti olla olemassa. Tiedostot nimeltään **templateDetails.xml** ja **index.php** ovat sellaisia, joiden täytyy olla juuri edellä mainitun kaltaisia, koska näillä nimillä Joomla!:n ydin niitä kutsuu. Tiedostonimet ovat myös niin sanotusti case-sensitiivisiä eli niiden isoilla ja pienillä kir-

jaimilla on myös merkitystä. Nämä tiedostot on sijoitettu Joomla!-n kotihakemiston **templates**-kansioon. Templates-kansiossa on vielä eriteltynä jokainen sivupohja omaan kansioon eli hakemistorakenne voisi olla seuraavanlainen:

*C:\Ohjelmatiedostot\Joomla15\templates\omatemplate\templateDetails.xml.*

### ***templateDetails-tiedosto***

Tämä tiedosto on XML-muodossa olevaa metatietoa, joka kertoo Joomla!-lle, mitä toisia tiedostoja tarvitaan silloin, kun käytetään sivustoa, joka käyttää tätä sivupohjaa. Periaatteessa tämä tiedosto on luettelo muun muassa kuvatiedostoista, joita on käytössä sivustolla. Tekijä, oikeudet ja muut löytyvät myös tiedoston tiedoista.

Liite 3 on esimerkki oikeasta sivupohjan kyseisestä tiedostosta. Tämä esimerkki on otettu Comanche-tuotteiden esittelysivustosta eli kyseessä on siis täysin toimiva ja valmis esimerkki tästä tiedostosta.

### ***index-tiedosto***

Kyseessä on sivuston tärkein tiedosto, koska `index.php`-n avulla määritetään se, miten Joomla!-n eri komponentit ja moduulit sijoittuvat. Tiedosto on yhdistelmä PHP:tä ja (X)HTML:llä. Tästäkin tiedostosta on esimerkki liitessä 4. Samoin kuin aiempi esimerkki `templateDetails.xml`-tiedostosta, tämäkin kuuluu Comanche-sivustoon.

On olemassa tietysti muitakin tiedostoja, mutta ne eivät ole pakollisia. Näille tiedostoille on vakiintunut seuraavanlaisia nimiä:

*/omatemplate/***template\_thumbnail.png** = sivuston esikatselukuva,

*/omatemplate/css/***template.css** = sivuston CSS tyyli-tiedosto,

*/omatemplate/images/***logo.png** = mikä tahansa sivuston kuva.

Kuten aiemmin todettiin, Joomla!-n sivupohja ei ole yksittäinen tiedosto, vaan kokonaisuus erilaisia tiedostoja, jotka muodostavat niin sanotun sivupohjan. Tämän vuoksi, jos puhumme sivupohjan muokkauksesta, on hankala kiteyttää sitä yhteen tapaan tai menetelmään. Muokatessamme tiedostoja, kuten `templateDetails.xml`, muokkaamme samalla sivupohjaa ja sen sisältöä, ja jos muokkaamme CSS-tyylitiedostoja, niin muokkaamme sivupohjan ulkoasua. Itse Joomla!-n käyttöliittymässä tapahtuva osioiden ja kategorioiden muokkaus voi vaikuttaa joko sivuston toiminnallisuuteen tai ulkoasuun ja tämä myös on sivupohjan muokkausta.

### 5.3 Sisällön määrittäminen

Kun sivuston ulkoasu ja sivupohja on luotu, voidaan siirtyä sivuston sisällön määrittelyyn. Määrittelyllä tarkoitan, että luodaan tavallaan rakenne sivun sisällölle. Samoin kun `index.php`, tiedosto luo rakenteen sivuston osille, kuten grafiikalle, menuille ja otsikoille. Voisimme myös puhua valikkorakenteesta eli siitä, mitä kaikkea materiaalia löytyy eri linkkien takaa. Apuvälineet, joiden avulla tämä määrittely tapahtuu, ovat nimeltään **Osiot**, **Kategoriat** ja **Artikkelit**. Nämä tulevat englanninkielien sanoista **Section**, **Category** ja **Articles**.

Alkuun on syytä kertoa hyvä muistisääntö, jolla osiot ja kategoriat pysyvät järjestyksessä ainakin omassa muistissa. **Osiot** voivat sisältää useita **kategorioita**, ja **kategoriat** voivat sisältää useita **artikkeleita**. Esimerkkinä voidaan perustaa osio nimeltä uutiset, jolla on kategoriat kotimaa ja ulkomaat. Nämä kategoriat sisältävät sitten artikkelit niin, että kotimaata koskevat ovat kotimaa-kategoriassa ja ulkomaata koskevat tietyt ulkomaat-kategoriassa. Tämän saman voi ajatella vielä konkreettisesti niin, että on linkki uutiset, josta avautuu kotimaan sekä ulkomaan uutisotsikot, jotka ovat sitten artikkelit.

Luotaessa uutta artikkelia, määritellään sille aina **Osio** ja **Kategoria**. Näin voidaan kertoa Joomla!-lle, mihin kohtaan sivua artikkeli kuuluu. Tämä on myös käyttäjälle

erittäin tärkeä tieto, josta näkee heti, missä artikkeli sijaitsee sivustolla. Fyysisesti artikkelit ovat kaikki samassa paikassa, mutta sitten, kun Joomla! koostaa sivun näytölle, hakee se **Osio-** ja **Kategoria-**tietojen avulla oikean artikkelin.

Käytettäessä valmista Joomla!-sivupohjaa on niissä yleensä jo valmiiksi määriteltynä joitain osioita ja kategorioita. Jos oma sivupohja rakennetaan alusta asti, on tietysti selvä, että osiot ja kategoriat tulee määritellä. Osioille ja kategorioille on molemmille oma niin sanottu manager-osio, eli paikka, missä niiden osia voi hallinnoida. Osiot ja kategoriat, eli valikkorakenne, on aina vapaasti muokattavissa, ja siksi sen ei heti ensimmäisellä kerralla tarvitse olla lopullinen. Tietysti hyvin suunniteltu heti alusta asti vähentää muokkaamistarpeita sitten myöhemmin.

## 5.4 Toimintojen havainnollistaminen esimerkkien avulla

Kuten otsikko kertoo, käydään seuraavaksi läpi kaksi esimerkkiä siitä, miten Joomla! varsinaisesti toimii. Tarkoituksena ei ole käydä jokaista Joomla!:n toimintoa läpi, vaan esimerkeiksi on valittu kaksi perustoimintoa, jotka havainnollistetaan esimerkksisivuston avulla.

Kuten aikaisemmin luvussa 3.5 Joomla!:n ominaisuudet todettiin, Joomla!:n avulla on mahdollista luoda sivusto ilman, että osaa HTML-kieltä tai jotain muuta ohjelmointikieltä. Sivustona käytetään Joomla!:n asennuksen mukana tulevia esimerkksisivustoja. Alun perin tarkoituksena oli käyttää Veljekset Keskinen Oy:lle tehtyä Comanche-tuotemerkin sivustoa. Sivustosta ei kuitenkaan koskaan tehty täydellistä versiota. Ainoastaan sivujen template eli sivupohja on valmis, mutta sisältöä eli artikkeleita ja kuvia ei ole vielä toteutettu. Aikaisemmin mainitut liitteet 2 ja 3 ovat siis täysin kelvollisia esimerkkejä Joomla!:n tiedostoista, vaikka ovatkin tästä kyseisestä Comanche-sivustosta, koska ne ovat valmiita.

### 5.4.1 Esimerkit

Visuaalisten ja ulkoasuun liittyvien muutosten tekeminen Joomla!:lla ei välttämättä ole sen helpompaa kuin, jos sivut tehdään ilman sitä, koska Joomla! käyttää myös CSS-tyylejä kuten normaali nykyaikainen web-sivusto. Tämän takia perinpohjainen ulkoasun muokkaaminen ei käy niin käden käänteessä ja helposti kuin esimerkiksi sisällön ja esitettävän materiaalin. Perinpohjaisella ulkoasun muutoksella tarkoitan esimerkiksi erilaisen graafisen ulkoasun tuomista sivulle, ja vaikka sivuston logon muuttamista. Toki jonkinasteinen ulkoasun muokkaaminen onnistuu kätevästi, mutta tällöin on lähinnä kyse sivuston eri osien paikoista, esimerkiksi siitä, esitetäänkö navigaatiopalkki horisontaalisesti vai vertikaalisesti.

Jos kuvitellaan sivusto, joka on luotu ihan normaalisti ilman minkään julkaisujärjestelmän apua, niin tiedämme, että tekstin lisääminen sivustolle tulee toteuttaa suoraan lähdekoodiin. Samoin tekstin muotoilu pitää lisätä joko suoraan lähdekoodiin tai sitten CSS-tyylien kautta. Joomla:ssa tekstin muokkaaminen käy kätevästi WYSIWYG-editorien kautta. Tällöin tekstin muokkaus on yhtä helppoa kuin tekstinkäsittelyohjelmalla ja lopputuloksen näkee suoraan ruudulla.

#### ***Esimerkki 1: Tekstin muokkaus***

Aloitetaan yksinkertaisella esimerkillä. Kuvassa 4 on esimerkksisivuston etusivu. Teksti on samanlaista kauttaaltaan ja ainoastaan otsikot eroavat.

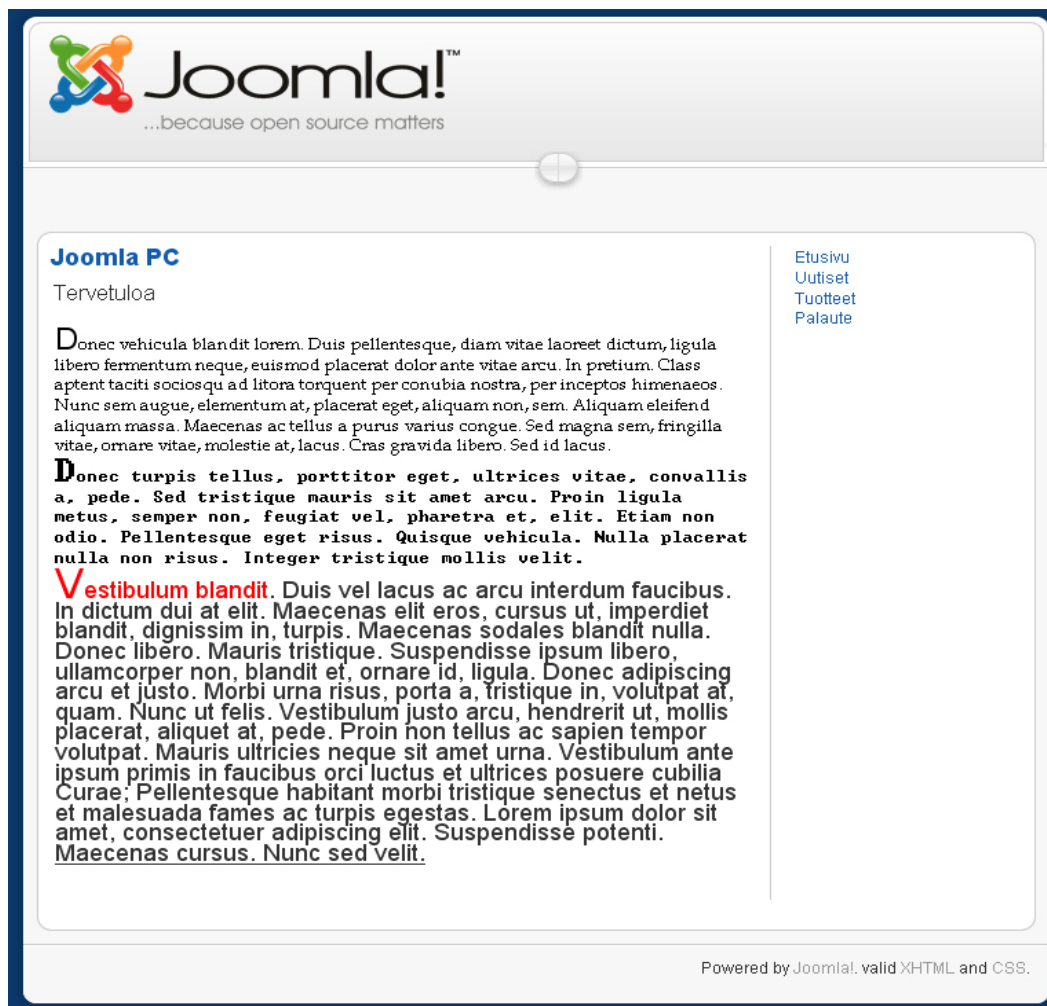


**Kuva 4: Esimerkkisivu ennen muotoilua**

Avataan esimerkkisivuston etusivu-artikkeli TinyMCE-editoriin. Kuva 5 on editorin näkymä, kun artikkeli on muokkaustilassa. Kuvasta näkyy artikkelille määrättyjä perustietoja, kuten otsikko, alias, osio, kategoria, julkaisutila ja tieto siitä, onko se julkaistuna etusivulla. Keskitymme nyt kuitenkin itse artikkelin tekstiin.

Pienen ja yksinkertaisen muokkauksen jälkeen artikkeli voi näyttää kuvan 6 kaltaiselta. Kuvan 6 tehdyt muutokset ovat tehty samoja periaatteita käyttäen kuin esimerkiksi tekstinkäsittelyohjelmassa. Mitään muokkausta lähdekoodiin tai CSS-tyylitiedostoihin ei tarvitse tehdä. Kuvassa 7 on esimerkkisivu muokattuna selain-näkymässä.





Kuva 7: Esimerkkisivu muokattuna

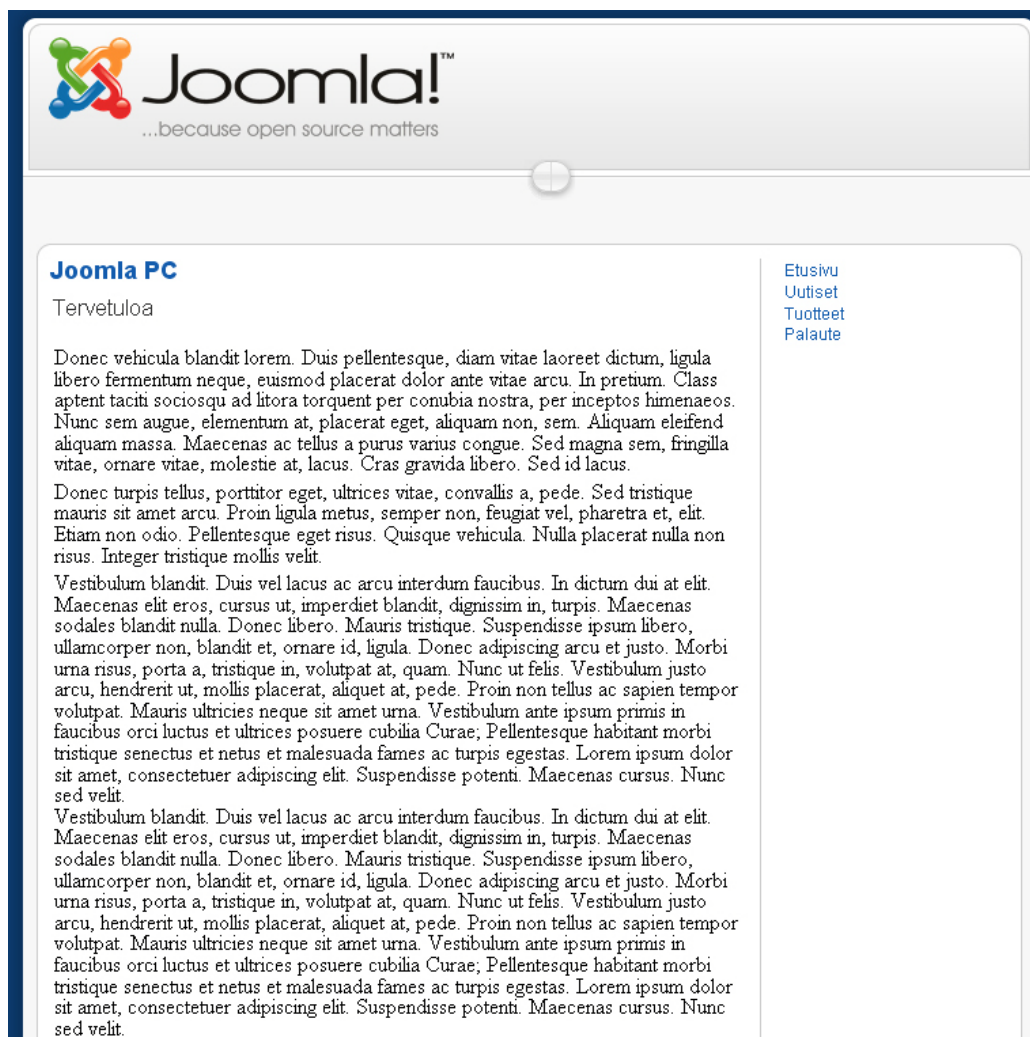
## Esimerkki 2: Read more, image ja page break painikkeet

Liitännäiset ja moduulit-kappaleen (4.2.1) kohdalla esiteltiin kolme editoripainiketta; Image, Pagebreak ja Read more.



Image-painikkeen avulla artikkelin lisättiin kuva, Pagebreakin avulla artikkelille laitettiin sivunvaihto ja Read more teki artikkeliin esikatselun ja piilotti lopun tekstin painikkeen taakse. Read more toimii artikkelissa myös yksin, mutta Pagebreakin kanssa se mahdollistaa artikkelin jakamisen sivuihin.

Kuvitellaan, että esimerkkisivuston etusivulla oleva artikkeli sisältää kuvan 8 mukaan niin paljon tekstiä, että se ei mahdu ruudulle.



**Kuva 8: Esimerkkisivu, jossa paljon tekstiä**

Lisäämällä Pagebreak- ja Read more-toiminnot ja nähdään, miten sivun dynamiikka muuttuu. Kuvassa 9 näkyvät katkoviivat merkitsevät, että artikkeliin on lisätty Read more (punainen) ja Pagebreak (harmaa).

**Article: [ Edit ]** Preview

Title:  Published:  No  Yes

Alias:  Front Page:  No  Yes

Section:  Category:

**B I U ABC** | **Paragraph** | Times New Roman | 3 (12 pt)

Donec vehicula blandit lorem. Duis pellentesque, diam vitae laoreet dictum, ligula libero fermentum neque, euismod placerat dolor ante vitae arcu. In pretium. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Nunc sem augue, elementum at, placerat eget, aliquam non, sem. Aliquam eleifend aliquam massa. Maecenas ac tellus a purus varius congue. Sed magna sem, fringilla vitae, ornare vitae, molestie at, lacus. Cras gravida libero. Sed id lacus.

---

Donec turpis tellus, porttitor eget, ultrices vitae, convallis a, pede. Sed tristique mauris sit amet arcu. Proin ligula metus, semper non, feugiat vel, pharetra et, elit. Etiam non odio. Pellentesque eget risus. Quisque vehicula. Nulla placerat nulla non risus. Integer tristique mollis velit.

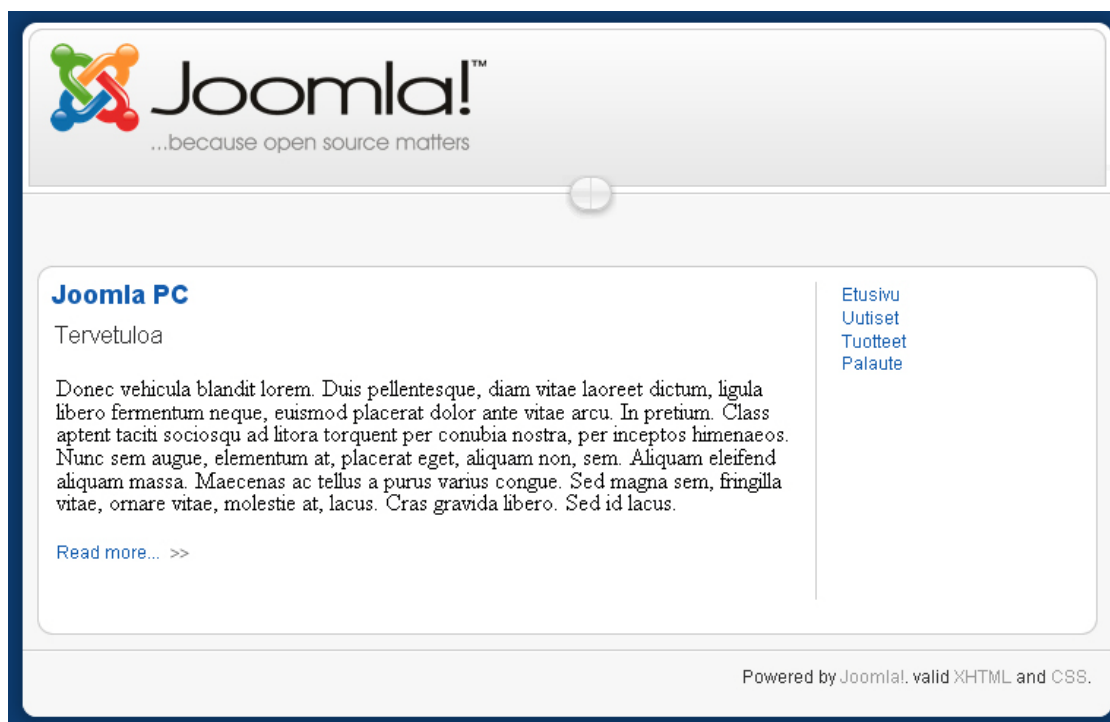
Vestibulum blandit. Duis vel lacus ac arcu interdum faucibus. In dictum dui at elit. Maecenas elit eros, cursus ut, imperdiet blandit, dignissim in, turpis. Maecenas sodales blandit nulla. Donec libero. Mauris tristique. Suspendisse ipsum libero, ullamcorper non, blandit et, ornare id, ligula. Donec adipiscing arcu et justo. Morbi urna risus, porta a, tristique in, volutpat at, quam. Nunc ut felis. Vestibulum justo arcu, hendrerit ut, mollis placerat, aliquet at, pede. Proin non tellus ac sapien tempor volutpat. Mauris ultricies neque sit amet urna. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse potenti. Maecenas cursus. Nunc sed velit.

---

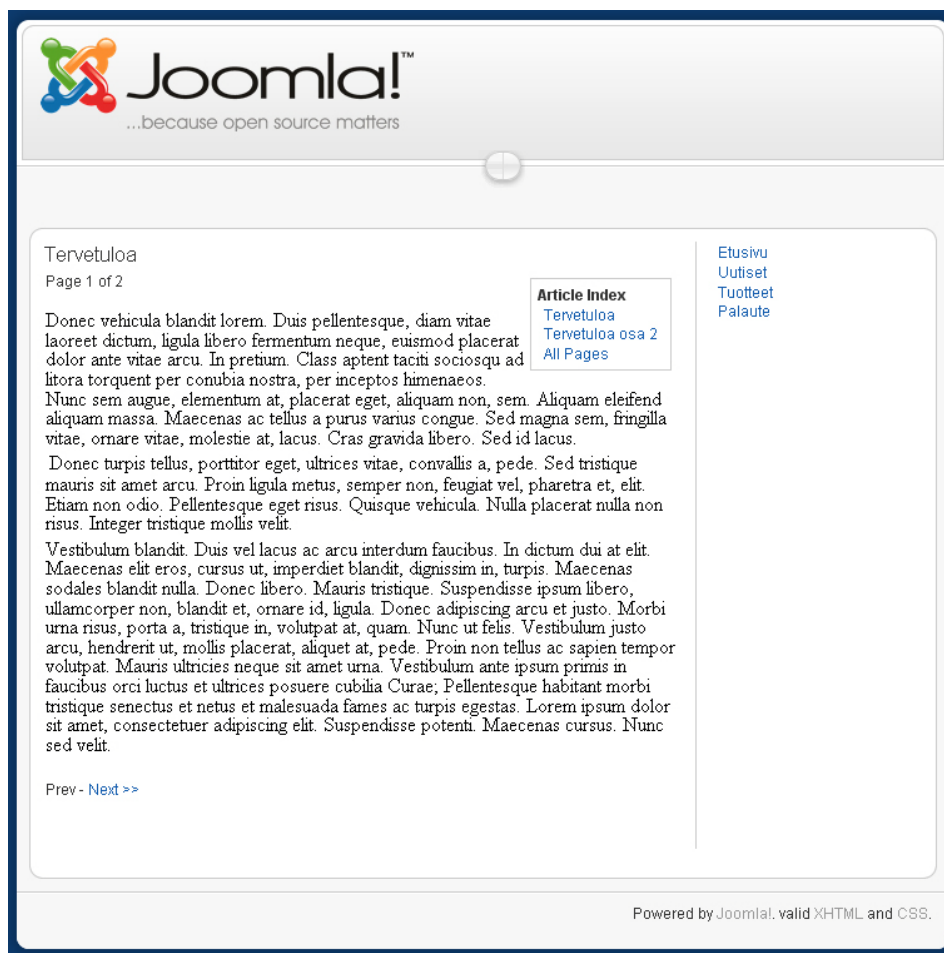
Vestibulum blandit. Duis vel lacus ac arcu interdum faucibus. In dictum dui at elit. Maecenas elit eros, cursus ut, imperdiet blandit, dignissim in, turpis. Maecenas sodales blandit nulla. Donec libero. Mauris tristique. Suspendisse ipsum libero, ullamcorper non, blandit et, ornare id, ligula. Donec adipiscing arcu et justo. Morbi urna risus, porta a, tristique in, volutpat at, quam. Nunc ut felis. Vestibulum justo arcu, hendrerit ut, mollis placerat, aliquet at, pede. Proin non tellus ac sapien tempor volutpat. Mauris ultricies neque sit amet urna. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse potenti. Maecenas cursus. Nunc sed velit.

**Kuva 9: Etusivu-artikkeli, johon lisätty pagebreak ja read more**

Lopputuloksena kuvassa 10 on huomattavasti selkeämpi etusivu. Kuvassa 10 tekstin jälkeen näkyy linkki ”Read more.>>”, josta pääsee lukemaan loput artikkelin tekstistä. Tällöin näkymä muuttuu kuvan 11 kaltaiseksi. Kuvassa 11 näkyy myös, kuinka Joomla! lisää Article Index-laatikon artikkeliin. Tästä voidaan navigoida artikkelin eri sivuilla tai näyttää se kerralla kokonaan.



Kuva 10: Esimerkkisivun näkymä read more ja pagebreak toimintojen lisäämisen jälkeen



Kuva 11: Esimerkkisivun ensimmäinen sivu

## 6 Julkaisujärjestelmän ohjeet

Osana järjestelmän käyttöönottoprosessia on tietysti käyttäjien ohjeistus, jotta ohjelmaa voidaan käyttää myös suhteellisen itsenäisesti ilman jatkuvaa tukea ja neuvontaa. Tässä luvussa käydään läpi niitä asioita, joita pitää ottaa huomioon, kun tehdään ohjetta ja osaltaan selvitetään liitteenä olevaa ohjeistusta. Tämä luku ei siis sisällä varsinaista ohjeistusta itsessään.

### 6.1 Yleistä ohjeista ja niiden tekemisestä

Tutkiessani ja etsiessäni sopivaa lähdemateriaalia ohjelmiston ohjeistuksen tekoon huomasin, että sitä ei juuri ole olemassa. Asiaa mietittyäni huomasin kuitenkin, että teknisen alan kirjallisuudessa oli paljon materiaalia erilaisten laitteiden käyttöohjeistuksen tekemisestä. Mielestäni näihin erilaisiin käyttöohjeisiin käytettävät säännöt pätevät yhtäläillä silloin, kun kyseessä on ohjelmiston tai ohjelman käyttöohjeet. Sovellan siis näitä teoksia oman ohjeistukseni tarkastelussa.

Ohjeen tekijällä on tietysti suurin vastuu ohjeen onnistumisesta. Tämä johtuu tietysti siitä, että ohjeen sisältö perustuu siihen tietoon, jonka tekijä joko ennestään omistaa, tai jonka tekijä kerää ohjetta varten. Ei tietysti riitä, että on tietoa, se pitää vielä tekijän itse ymmärtää ja sen jälkeen tehdä siitä ohje, jonka muutkin ymmärtävät. (*Kylänpää 2003: 78*)

*Kauppinen, Nummi ja Savola (2004: 102)* esittävät teoksessaan asioita, jotka soveltuvat ohjelman käyttöoppaan tekemiseen. Kirjoittamisen ja puhumisen käsikirja teoksen mukaan lähtökohdat ohjeen tekijälle ovat seuraavat:

- lukijan mielenkiinnon säilyttäminen
- asioiden selkeä esittäminen

- varmuus, että ennen siirtymistä ohjeen seuraavaan vaiheeseen edelliset asiat ovat lukijalle selkeitä
- ohjeiden helppolukuisuus ja selkeys.

Oikea asenne on siis kirjoittajalle tärkeää. Riippuen siitä, kenelle ohjeita kirjoittaa, tulee ottaa huomioon heidän lähtötasonsa. Oman ohjeistukseni olen pyrkinyt tekemään henkilöille, joille Joomla! on jo tuttu. Uskon kuitenkin, että se on tarpeeksi selkeä ja hyödyllinen myös sellaiselle, joka on vasta-alkaja Joomla!-n käytössä ja vasta opettelee koko ohjelman käyttöä.

Kirjoittajan ei myöskään tulisi sortua liialliseen hienosteluun. Useasti ohjeen kirjoittaja tietää huomattavasti paljon enemmän asiasta kuin sen lukija, joten tekijän ei tule sortua kirjoittamaan liian vaikeaselkoisia ohjeita. Muun muassa liiallinen vaikeiden termien käyttäminen tekstissä tulisi unohtaa, ellei niitä ole hyvin selitetty. Antti Wiio (2004: 74) puhuu teoksessaan juuri siitä, käytetäänkö ohjeistuksessa tekniikan kieltä vai käyttäjän kieltä.

Loppujen lopuksi ohjeella on yksi selkeä ja tärkeä tarkoitus. Ohjeen tulisi toimia käyttäjän ja tekijän välisenä linkkinä. Tämän linkin tarkoitus on vähentää turhaa työtä, lisätä tehokkuutta ja ehkäistä väärinkäyttöä, joka aiheuttaisi ongelmatilanteita. (Kylänpää 2003: 76)

## 6.2 Ohjeiden suunnittelu ja toteutus

Ohjeiden tekemisessä hyvä suunnittelu on tärkeää. Joomla!-n kaltaisen ohjelman ominaisuuksista olisi mahdollista kirjoittaa monikymmensivuinen ohje. Suuren materiaalmäärän edessä tulee miettiä, mitä kannattaa ottaa mukaan ja myös miksi. Tässä helpottaa huomattavasti se, että rajaa ohjeelle tietyn käyttäjäkunnan, jos se suinkin mahdollista. Sisällön kannalta on täysin eri asia, lukeeko ohjetta esimerkiksi järjestelmän ylläpitäjä vai normaali käyttäjä. Tämän työn liitteenä oleva ohje (liite 1), on tehty juuri

niin sanotuille normaaleille käyttäjille. Sisällöltään siinä keskitytään ainoastaan yksinkertaisiin sisällön muokkauksiin ja lisäyksiin menemättä sen enempää Joomla!:n toiminnallisuuksiin tai teknisiin asioihin.

Kun mahdollinen käyttäjäkunta on saatu rajattua, pitää ottaa selville ryhmän tarpeet ohjelman käyttöä koskien. Oman tarvekartoituksen tein muun muassa kyselemällä käyttäjiltä ideoita ja selvittämällä, minkälaisia työvaiheita sivustojen materiaalin ylläpidossa on. Ideoita käyttäjiltä ei omalla kohdallani mahdottomasti tullut, mutta ainakin tuli varmistettua se mahdollisuus. Käytin siis ohjeistuksen suunnittelussa osaksi käyttäjäkeskeistä suunnittelua (*Kuutti: 2003: 140*). Tämä tarkoittaa sitä, että ohjeen suunnittelun alkuvaiheessa kävin asioita läpi loppukäyttäjien kanssa. Joomla! mahdollistaa hyvinkin monipuolisen sisällön muokkauksen, mutta tietyt perusasetukset sisällön osalta tulee olla tietynlaiset, ja juuri näihin ohjeistusta muun muassa kohdistettiin.

Ohjeiden visuaalinen ilme on omalta osaltaan myös tärkeä. Pelkän tekstin avulla on vaikea lähteä liikkeelle. Havainnollistavat kuvat helpottavat ohjeen laatimista ja hyvin mietityt ruutukaappaukset ohjelmasta tuovat myös käytettävyyttä. Varsinkin, kun tehdään ohjetta ohjelman käytöstä, kuvat vähentävät mahdollisten väärinkäsitysten syntymistä. Kuvat tuovat myös mielenkiintoa ohjeisiin ihan pelkällä olemassaolollaan. Liiallinen kuvien käyttö ja esimerkiksi niiden huono asemointi voivat puolestaan vaikeuttaa ohjeen lukua ja käyttöä. Joomla!:n kaltaisen ohjelman kohdalla ruutukuvat ohjeessa ovat melkein pakollisia, jos aikomus on saada asia täysin ymmärretyksi.

Omassa Joomla! ohjeessani on kahdenlaisia ohjeita: valmiin työvaiheen selostus ja yleisohje koskien jotain ohjelman osaa. Valmiissa työvaiheen ohjeessa on selostettu pelkästään kohdat ja vaiheet johonkin tiettyyn asiaan. Yleisohjeissa käydään läpi jokin kokonaisuus selittäen siitä mahdollisimman laajasti. Ei siis keskitytä tekemään jotain tiettyä asiaa. Ohjeiden sisältö muotoutui nykyisenlaiseksi paljolti juuri edellä mainitun käyttäjien tarpeiden kartoituksen avulla.

### 6.3 Artikkelin lisääminen

Kohta artikkelin lisääminen on tehty niin sanotusti normaalin ohjeen mukaisesti. Tällä tarkoitan, että siinä ei ole käsitelty sellaisia asioita, jotka koskisivat vain Veljekset Keskinen Oy:n toimintatapoja. Ohjeistus artikkelin lisäämiseen on hyvin yksinkertainen ja loogisesti vaiheesta vaiheeseen etenevä. Periaatteessa otsikon Artikkelin lisääminen -ohje koostuu kahdesta eri osasta. Aluksi käyttäjä neuvotaan oikeaan valikkoon ja kohtaan, josta uuden artikkelin lisääminen voidaan aloittaa. Tämän jälkeen ohjeistus onkin eri toimintojen ja kohtien selitystä. Koska uusien artikkelien ei tarvitse olla mitenkään tietynlaisia, on hyvä antaa käyttäjälle mahdollisimman kattava kuvaus kaikista eri asetuksista, joiden avulla artikkeleita voi muokata. Siihen olen niistä myös tällä ohjeella pyrkinyt.

### 6.4 Tarjoussivujen lisääminen

Tarjoussivujen lisäämiseen tehty ohje poikkeaa edellisestä artikkelin lisäämisohjeesta. Tämä johtuu siitä, että tässä ohjeen osassa keskitytään sellaiseen asiaan, joka tehdään aina samalla tavalla ja sille on ennalta määrätty vaiheet. Tätä ohjetta voisi käyttää toki soveltaen muuallakin, mutta käytetyt viittaukset tekstissä on tarkoitettu nimenomaan Veljekset Keskinen ympäristöön ja käyttöön.

Tarjoussivujen lisäämiseen tarkoitettussa ohjeessa ei myöskään selitetä eri toimintoja niin laajasti kuin artikkelin lisäämisohjeessa, vaan siinä keskitytään vain tiettyihin vaiheisiin, jotka ovat tarpeellisia tarjoussivun lisäämisen kannalta. Tämän ohjeen osan kohdalla oletetaan, että käyttäjä on aiemmin tutustunut ainakin artikkelin lisäämisohjeeseen tai Joomla! on muuten tuttu. Näin ollen kaikkia toimintoja ei tarvitse käydä niin tarkkaan läpi.

## 7 Johtopäätökset ja yhteenveto

Työn tarkoituksena oli käydä läpi julkaisujärjestelmän käyttöönottoprosessia, käyttäen esimerkkinä toimeksiantajaa. Ohjeistuksen laatiminen oli työn toinen päätavoite. Teoriapohjana työssä käytettiin tekijän omaa tietoa, sekä työn aihealuetta koskevaa lähdemateriaalia. Työn lopputuloksena syntyivät Joomla!:n ohjeistus Veljekset Keskinen Oy:lle ja teoriaosuus, jossa tarkastellaan julkaisujärjestelmän käyttöönottoprosessia.

Työn alussa ongelmaksi muodostui aihetta käsittelevän materiaalin hankkiminen. Varsinkin painettu materiaalia oli vaikea löytää. Tästä johtuen lähdekritiikin kanssa piti olla erityisen tarkka, koska suuri osa lähteistä oli verkosta hankittua. Toimeksiantajan puolelta tullut ongelma liittyi esimerkisivustoon, joka jäi kesken ja aiheutti osaltaan vaivaa työn tekemisessä ja erityisesti esimerkkien laatimisessa.

Ohjeen koostaminen lopulliseen muotoonsa kesti kauemmin kuin oli tarkoitus, vaikka ensimmäinen toimeksiantajan hyväksymä versio saatiin ajoissa valmiiksi. Ohjeen valmistumisen takarajana oli elokuun kolmas viikko. Täysin valmis ja viimeistelty versio syntyi vasta viikolla 42. Tähän vaikuttivat henkilökohtaiset kiireet ja halu panostaa opinnäytetyön teoria osuuteen, jotta aikataulu pitäisi sen suhteen. Muuten yhteistyö toimeksiantajan kanssa sujui mallikkaasti ja tavoitteet täyttyivät.

Tavoitteet täyttyivät työn kannalta hyvin. Julkaisujärjestelmän käyttöönottoprosessin kuvaus onnistui ja siinä käytetty tarkkuus asian kuvaamiseksi on haluamallani tasolla. Myös ohjeistukseen liittyvän teorian laatiminen onnistui hyvin ja siinä on mielestäni pohdittu asioita tarpeeksi laajalla näkökulmalla. Toimeksiantajan kanssa yhteisenä tavoitteena ollut Joomla!:n käyttöoppaan teko meni myös suunnitellusti. Joomla!:n käytön oppiminen ja sen tutkiminen antoi myös paljon uutta tietoa ja täydensi tietoa web-suunnittelusta ja siihen liittyvistä asioista. Oma epävirallinen tavoitteeni oppia käyttämään Joomla!:a niin, että pystyn sillä sivuja tekemään ja hallinnoimaan täyttyi myös.

Menetelmät työn tekemiseen ja lähdemateriaalin hankintaan olivat alussa keskinkertaiset. Tekijän kokemattomuus laajojen raporttien teossa näkyi selvästi. Työn edetessä tieto ja taito lisääntyivät, ja asioiden tarkasteleminen kokonaisuuden kannalta kohentui. Joomla!:a koskevat tiedot lisääntyivät tekijän osalta myös merkittävästi. Ohjeen tekemisen kannalta karttuneet tiedot Joomla!-sta auttoivat ymmärtämään paremmin sitä, miten ohje kannattaa koostaa. Yleisellä tasolla tietous julkaisujärjestelmistä ja niiden eri versioista kasvoi ja tietämys muun muassa järjestelmän valinnasta ja siihen liittyvistä asioista tarkentui.

Julkaisujärjestelmän valintaa koskevia määrittelyitä voidaan suoraan soveltaa tilanteisiin, joissa tarvitaan tietoa järjestelmän valinnasta. Teoria osuus on hyvinkin yleistettävissä muihin samankaltaisiin projekteihin. Tämä sen takia, että asiat on käyty läpi yleisellä tasolla. Työn liitettä 1, voidaan myös soveltaen hyödyntää Joomla!-n käytössä. Tulee kuitenkin muistaa, että osa ohjeesta on tarkoitettu nimenomaan Veljekset Keski-  
sen tarpeisiin. Osa on kuitenkin yleisesti pätevää. Täytyy muistaa myös, että käyttöoppaan sisältö vanhentuu tietysti sitä tahtia mitä siinä käsitellyt asiat menevät eteenpäin ohjelmassa.

Omasta mielestäni opinnäytetyö projekti sujui alusta loppuun melko hyvin. Tutkimusmenetelmät kurssilla laadittu alustava aikataulu piti hyvin paikkansa. Lopetustoimien tekeminen tosi vei kauemmin kuin suunniteltua. Työn jakaminen osiin heti alussa auttoi työn itse tekemisessä. Jatkoin tätä jaottelua läpi työn ja suunnittelin seuraavien vaiheiden työstöä esimerkiksi viikoksi eteenpäin. Tämän kaltainen jaksotus toimi hyvin ja antoi mahdollisuuden keskittyä tiettyyn osaan työtä aina kerralla. Jaottelu olisi välillä voinut toimia vielä täsmällisemmin. Osien sisällön tarkempi määrittelemineen heti työn alussa olisi tuonut myös selkeyttä ja toimivuutta työn tekemiseen. Nyt työ muotoutui ja muuttui vielä pitkän aikaa aloittamisen jälkeen.

## Lähteet

- Apache Software Foundation, The. 2008. Apache 2.2.9 released. [online] [viitattu 15.9.2008]. <http://httpd.apache.org/>
- Axelsson, Jonny, Birbeck, Mark, Dubinko, Micah, Epperson, Beth, Ishikawa, Masayasu, McCarron, Shane, Navarro, Ann & Pember ton, Steven. 2006. XHTML™ 2.0: 8.4. The div element [online] [viitattu 17.7.2008]. [www.w3.org/TR/xhtml2/mod-structural.html#sec\\_8.4](http://www.w3.org/TR/xhtml2/mod-structural.html#sec_8.4).
- Boiko, Bob. 2007. Content management bible. Yhdysvallat, Indianapolis: Wiley Publishing Inc.
- Bos, Bert. 2008. Cascading Style Sheets [online] [www.w3.org/Style/CSS/](http://www.w3.org/Style/CSS/)
- Buytaert, Dries. 2008. About Drupal. [online ] [viitattu 19.9.2008]. [drupal.org/about](http://drupal.org/about)
- Greatech Group. 2008. Lisk CMS: esittely [online] [viitattu 20.9.2008]. [lisk-cms.com/documents/lisk\\_4\\_4\\_presentation.pdf](http://lisk-cms.com/documents/lisk_4_4_presentation.pdf)
- How to get Joomla! 2008. [online] [viitattu 15.9.2008]. [dev.joomla.org/content/view/17/60/](http://dev.joomla.org/content/view/17/60/)
- Kauppinen, Anneli, Nummi, Jyrki & Savola, Tea. 2004. Kirjoittamisen ja puhumisen käsikirja: Tekniikan viestintä. Helsinki: Edita Prima Oy.
- Kuutti, Wille. 2003. Käytettävyys, suunnittelu ja arviointi. Saarijärvi: Gummerus Kirjapaino Oy.
- Kylänpää, Esa. 2003. Viestintätilanteet: oppi- ja harjoituskirja. Jyväskylä: Gummeruksen Kirjapaino Oy.
- North, Barrie. 2007. Joomla 1.5 Template Tutorial. [online][viitattu 18.8.2008][www.compassdesigns.net/tutorials/208-joomla-15-template-tutorial.html](http://www.compassdesigns.net/tutorials/208-joomla-15-template-tutorial.html)
- Pohjanoksa, Iiro, Kuokkanen, Eevi & Raaska, Timo. 2007. Viesti verkossa: digitaalisen viestinnän käsikirja. Jurva: WS Bookwell Oy.
- Raggett, Dave, Le Hors, Arnaud & Jacobs, Ian. 1999. HTML 4.01 Specification: 11.1 Introduction to tables [online] [viitattu 15.7.2008]. [www.w3.org/TR/html4/struct/tables.html#h-11.1](http://www.w3.org/TR/html4/struct/tables.html#h-11.1)
- Rockley, Ann. 2003. Managing enterprise content: A unified content strategy. Yhdysvallat: Pearson Education, Inc.

- Wallace, Andy. 2007. Joomla! 1.5:installation manual [online] [viitattu 2.6.2008].[downloads.joomla.org/docmanfileversion/1/7/4/17471/1.5 Installation\\_Manual\\_version\\_0.5.pdf](http://downloads.joomla.org/docmanfileversion/1/7/4/17471/1.5%20Installation_Manual_version_0.5.pdf)
- Wiio, Antti. 2004. Käyttäjystävällisen sovelluksen suunnittelu. Helsinki: Edita Prima Oy.

## Liitteet


- Liite 1 Joomla ohje
- Liite 2 CMS vertailutaulukko
- Liite 3 TemplateDetails.xml
- Liite 4 Index.php

## LIITE 2

Product	<a href="#">Joomla! 1.5.7</a>	<a href="#">Drupal 6.4</a>	<a href="#">Lisk CMS 4.4</a>
Last Updated	09.10.08	8/14/2008	03.12.07
<b>System Requirements</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
Application Server	Any that supports PHP (Apache recommended)	PHP 4.3.5+	PHP 4.3.3+
Approximate Cost	Free	Free	500 USD
Database	MySQL	MySQL, Postgres	MySQL, Oracle, MSSQL
License	GNU/GPL v2	GNU GPL	By domain
Operating System	Any	Any	Linux, FreeBSD, Microsoft Windows NT/2000
Programming Language	PHP	PHP	PHP 4.3 or later
Root Access	No	No	Yes
Shell Access	No	No	No
Web Server	Apache	Apache, IIS	Apache
<b>Security</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
Audit Trail	No	Yes	Yes
Captcha	Free Add On	Free Add On	Yes
Content Approval	Yes	Yes	Costs Extra
Email Verification	Yes	Yes	Yes
Granular Privileges	No	Yes	Yes
Kerberos Authentication	No	No	Costs Extra
LDAP Authentication	Yes	Free Add On	Costs Extra
Login History	Yes	Yes	Yes
NIS Authentication	No	No	Costs Extra
NTLM Authentication	No	Free Add On	Costs Extra
Pluggable Authentication	Yes	Yes	Costs Extra
Problem Notification	No	No	Yes
Sandbox	No	No	Yes
Session Management	Yes	Yes	Yes
SMB Authentication	No	No	Costs Extra
SSL Compatible	Yes	Yes	Yes
SSL Logins	Yes	No	Costs Extra
SSL Pages	Yes	No	Yes
Versioning	No	Yes	Costs Extra
<b>Support</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
Certification Program	No	No	No
Code Skeletons	No	Yes	Yes
Commercial Manuals	Yes	Yes	Yes
Commercial Support	Yes	Yes	Yes
Commercial Training	Yes	Yes	Yes
Developer Community	Yes	Yes	No
Online Help	Yes	Yes	Yes
Pluggable API	Yes	Yes	No

**Legend**

No  
Yes  
Free Add on  
Costs Extra  
Limited



Professional Hosting	Yes	Yes	Yes
Professional Services	Yes	Yes	Yes
Public Forum	Yes	Yes	No
Public Mailing List	No	Yes	Yes
Test Framework	Yes	Free Add On	No
Third-Party Developers	Yes	Yes	No
Users Conference	Yes	Yes	No

<b>Ease of Use</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
Drag-N-Drop Content	No	Free Add On	Yes
Email To Discussion	Free Add On	Free Add On	No
Friendly URLs	Yes	Yes	Yes
Image Resizing	Yes	Free Add On	Yes
Macro Language	Yes	Free Add On	Yes
Mass Upload	No	Free Add On	Yes
Prototyping	Yes	Limited	Yes
Server Page Language	Yes	Yes	No
Site Setup Wizard	No	Limited	
Spell Checker	No	Free Add On	No
Style Wizard	No	Limited	No
Subscriptions	No	Free Add On	Yes
Template Language	Yes	Limited	Yes
UI Levels	Yes	No	Yes
Undo	No	Limited	Yes
WYSIWYG Editor	Yes	Free Add On	Yes
Zip Archives	No	No	Yes

<b>Performance</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
Advanced Caching	Yes	Yes	No
Database Replication	No	Limited	Yes
Load Balancing	Yes	Yes	No
Page Caching	Yes	Yes	No
Static Content Export	No	No	Yes

<b>Management</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
Advertising Management	Yes	Free Add On	Yes
Asset Management	Yes	Yes	Yes
Clipboard	No	No	No
Content Scheduling	Yes	Free Add On	Yes
Content Staging	No	Free Add On	No
Inline Administration	Yes	Yes	Yes
Online Administration	Yes	Yes	Yes
Package Deployment	No	No	Yes
Sub-sites / Roots	Yes	Yes	Yes
Themes / Skins	Yes	Yes	Yes
Trash	Yes	No	Yes
Web Statistics	Yes	Yes	Yes
Web-based Style/Template Management	Yes	Yes	No
Web-based Translation Management	Free Add On	Yes	No

Workflow Engine	No	Limited	Yes
-----------------	----	---------	-----

<b>Interoperability</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
Content Syndication (RSS)	Yes	Yes	Yes
FTP Support	Yes	Limited	Costs Extra
iCal	No	Free Add On	
UTF-8 Support	Yes	Yes	Yes
WAI Compliant	No	Limited	Yes
WebDAV Support	No	No	No
XHTML Compliant	No	Yes	Yes

<b>Flexibility</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
CGI-mode Support	Yes	Yes	Yes
Content Reuse	Yes	Limited	No
Extensible User Profiles	Yes	Yes	Yes
Interface Localization	Yes	Yes	Yes
Metadata	Yes	Yes	Yes
Multi-lingual Content	Free Add On	Yes	Yes
Multi-lingual Content Integration	Free Add On	Free Add On	No
Multi-Site Deployment	Free Add On	Yes	No
URL Rewriting	Yes	Yes	Yes

<b>Built-in Applications</b>	<b>Joomla!</b>	<b>Drupal</b>	<b>Lisk CMS</b>
Blog	Yes	Yes	Costs Extra
Chat	Free Add On	Free Add On	Costs Extra
Classifieds	Free Add On	Free Add On	Costs Extra
Contact Management	Yes	Free Add On	Costs Extra
Data Entry	Free Add On	Free Add On	Yes
Database Reports	Free Add On	No	Yes
Discussion / Forum	Free Add On	Yes	Costs Extra
Document Management	Free Add On	Limited	Costs Extra
Events Calendar	Free Add On	Free Add On	Yes
Events Management	Free Add On	Free Add On	Yes
Expense Reports	Free Add On	No	Costs Extra
FAQ Management	Yes	Yes	Yes
File Distribution	Free Add On	Free Add On	Yes
Graphs and Charts	Free Add On	No	Yes
Groupware	Free Add On	Free Add On	Costs Extra
Guest Book	Free Add On	Free Add On	Yes
Help Desk / Bug Reporting	Free Add On	Free Add On	Costs Extra
HTTP Proxy	No	No	Costs Extra
In/Out Board	No	No	Yes
Job Postings	Free Add On	Free Add On	Costs Extra
Link Management	Yes	Free Add On	Yes
Mail Form	Yes	Free Add On	Yes
Matrix	No	No	Costs Extra
My Page / Dashboard	No	Free Add On	Costs Extra
Newsletter	Free Add On	Free Add On	Yes
Photo Gallery	Free Add On	Free Add On	Yes

Polls	Yes	Yes	Yes
Product Management	Yes	Free Add On	Yes
Project Tracking	Free Add On	Free Add On	Costs Extra
Search Engine	Yes	Yes	Yes
Site Map	Free Add On	Free Add On	Yes
Stock Quotes	No	Free Add On	Costs Extra
Surveys	Free Add On	Free Add On	Yes
Syndicated Content (RSS)	Yes	Yes	Costs Extra
Tests / Quizzes	Free Add On	Free Add On	Costs Extra
Time Tracking	No	Free Add On	Costs Extra
User Contributions	Yes	Yes	Costs Extra
Weather	No	Free Add On	Costs Extra
Web Services Front End	Yes	Limited	Costs Extra
Wiki	Free Add On	Free Add On	No

	Joomla!	Drupal	Lisk CMS
<b>Commerce</b>			
Affiliate Tracking	Free Add On	Free Add On	Yes
Inventory Management	Free Add On	Free Add On	Costs Extra
Pluggable Payments	Free Add On	Free Add On	Yes
Pluggable Shipping	Free Add On	Free Add On	Yes
Pluggable Tax	Free Add On	Free Add On	Yes
Point of Sale	Free Add On	No	Costs Extra
Shopping Cart	Free Add On	Free Add On	Yes
Subscriptions	Free Add On	Free Add On	Yes
Wish Lists	Free Add On	Free Add On	Costs Extra

## LIITE 3

```

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE install PUBLIC "-//Joomla! 1.5//DTD template 1.0//EN"
"http://dev.joomla.org/xml/1.5/template-install.dtd">
<install version="1.5" type="template">
  <name>Comanche.fi</name>
  <creationDate>07/16/08</creationDate>
  <author></author>
  <authorEmail></authorEmail>
  <authorUrl></authorUrl>
  <copyright></copyright>
  <license>GNU/GPL</license>
  <version>1.0.0</version>
  <description>TPL_TUURI_COMANCHE</description>
  <files>
    <filename>index.php</filename>
    <filename>templateDetails.xml</filename>
    <filename>template_thumbnail.png</filename>
    <filename>img/footer.png</filename>
    <filename>img/bg.jpg</filename>
    <filename>img/repeat-bg.png</filename>
    <filename>img/maint-bg.png</filename>
    <filename>css/index.html</filename>
    <filename>css/template.css</filename>
  </files>
  <positions>
    <position>breadcrumb</position>
    <position>left</position>
    <position>right</position>
    <position>top</position>
    <position>user1</position>
    <position>user2</position>
    <position>user3</position>
    <position>user4</position>
    <position>footer</position>
    <position>debug</position>
  </positions>
</install>

```

## LIITE 4

```

<?php
// no direct access
defined( '_JEXEC' ) or die( 'Restricted access' );
?>
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="<?php echo $this->language; ?>"
lang="<?php echo $this->language; ?>" >
<head>
  <jdoc:include type="head" />
  <link rel="stylesheet" href="<?php echo $this->baseurl ?>/templates/system/css/system.css"
type="text/css" />
  <link rel="stylesheet" href="<?php echo $this->baseurl ?>/templates/system/css/general.css"
type="text/css" />
  <link rel="stylesheet" href="<?php echo $this->baseurl ?>/templates/comanche/css/template.css"
type="text/css" />
</head>
<body>
  <div id="main">
    <div id="headerbg"></div>
    <div id="minheight"></div>
    <div id="menu">
      <jdoc:include type="modules" name="user3" />
    </div>
    <div id="content">
      <jdoc:include type="component" />
    </div><!-- end of content -->
    <div id="minheightclear"></div>
    </div><!-- end of main -->
    <div id="footer"></div>
    <jdoc:include type="modules" name="debug" />
  </body>
</html>

```