

TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse instituut
Infotehnoloogia eriala

Taavi Pauskar

Vabavaraliste sisuhaldussüsteemide
kasutajasõbralikkus

Bakalaureusetöö (6 EAP)

Juhendaja: Gunnar Nellis

Autor: “.....“ mai 2011
Juhendaja: “.....“ mai 2011

Lubada kaitsmisele

Professor: “.....“ 2011

TARTU 2011

Sisukord

Sisukord	2
Sissejuhatus	3
2 Vaadeldavad sisuhaldussüsteemid	4
2.1 Avatud lähtekoodiga tarkvarast	5
2.2 Kasutatavusest	5
2.3 Võrreldavatest sisuhaldussüsteemidest	6
1.1.1. WordPress 3.1	6
1.1.2. Joomla 1.6.1	6
1.1.3. Drupal 7.0	7
1.1.4. CMSMadeSimple 1.9.4.1	7
3 Kasutuslood	8
3.1 Olemasoleva lehekülje muutmine	8
3.2 Uue lehekülje lisamine	8
3.3 Menüü restruktureerimine	8
3.4 Esilehele peatüki lisamine	8
3.5 Veebilehe jaluses asuva informatsiooni muutmine	9
3.6 Uue administreerimislehe kasutaja lisamine	9
4 Hindamiskriteeriumid	10
4.1 Kohandatud Nielsen kriteeriumid	10
4.1.1 Ülevaade süsteemi staatusest	10
4.1.2 Seos rakenduse ja reaalse maailma vahel	11
4.1.3 Kasutajaliidese järjekindlus	11
4.1.4 Pigem äratundmine kui meenutamine	11
4.1.5 Paindlikkus ja kasutamise efektiivsus	12
4.1.6 Esteetiline ja minimalistlik ülesehitus	12
4.1.7 Vigade ennetamine ja nendest taastumine	13
4.1.8 Abi ja juhendite kättesaadavus	13
4.1.9 Teiste rakendustega sarnane ülesehitus	14
4.1.10 Jälgi veebirakenduste kujundustavasid	14
4.1.11 Brauseri tööriistade funktsioneerimine	14
4.1.12 Lubada lihtsat andmete sisestamist, mis vähendaks veaolukordi	15
5 Hindamine	16
5.1 CMSMadeSimple 1.9.4.1	16
5.2 Drupal 7.0	18
5.3 Joomla 1.6.1	20
5.4 WordPress 3.1	23
5.5 Hindamise kokkuvõte	25
Kokkuvõte	27
Usability of open source content management systems	28
Kirjandus	29
Lisad	30
Lisa 1	30

Sissejuhatus

Internetis on saadaval palju aktiivselt arendatavaid vabavaralisi sisuhaldussüsteeme (*CMS* – *Content Management System*). Paljud neist on võimalusterohked ja võimaldavad väga suurt oma vajaduste järgi kohandamist, ning neile arendatakse aktiivselt ka laiendusi ja lisasid. Sisuhaldussüsteemi paigaldamine oma serverisse on muudetud lihtsaks ja kiireks, siis antud töö eesmärgiks on uurida, kui lihtsalt on sisuhaldussüsteemi haldamise pool lahendatud, ehk kui kasutajasõbralik on üks sisuhaldussüsteem ühe veebilehe haldajale. Kasutajasõbralikkuse all peame silmas kasutatavust (*usability*) ja kasutaja-kogemust (*user experience*). Kuna paljudes sisuhaldussüsteemides on võimalik administreerimisliidese välimust muuta, siis vastavas töös keskendume vaikimisi määratud välimusele.

Töö käigus koostatakse esmalt sisuhalduse kasutajasõbralikkuse hindamiseks kriteeriumid, seejärel viiakse läbi nelja sisuhaldussüsteemi (WordPress 3.1, Joomla 1.6.1, Drupal 7.0 ja CMSMadeSimple 1.9.4.1) peal hindamine, analüüsid nende kasutajasõbralikkust. Töös kirjeldatakse ühe veebilehe haldamisel põhilised operatsioonid, seejärel hinnatakse kui efektiivselt nende operatsioonide läbiviimine on lahendatud.

Töö eesmärgiks on anda ülevaade levinumate vabavaraliste sisuhaldussüsteemide administreerimisliideste kasutajasõbralikkusest ja tuua välja üldised puudused, millega pole kasutajaliideste loomisel arvestatud.

2 Vaadeldavad sisuhaldussüsteemid

Sisuhaldussüsteemid on mõeldud internetilehekülgede loomiseks ja nende sisu ning kasutajaskonna haldamise lihtsustamiseks. Kuna veebilehtede sisuks ei ole enam ammu ainult tekst, vaid ka erinevad meediumid - pildid, videod, audio, erinevad failid ja lisaks on ka võimalus läbi brauseri kuvada interaktiivseid lahendusi, näiteks Flash rakendused, Java Appletid¹ ning HTML5² levikuga populaarsust koguvad dünaamiline SVG³ ja JavaScripti abil joonistatud vektorgraafika - siis kõige selle mugavaks haldamiseks on mõistlik kasutada sisuhaldussüsteemi.

Ilma sisuhaldussüsteemita tuleb kogu veebilehel olev info eelnevalt käsitsi HTML keelde kodeerida ja omavahel hüperlinkidega siduda - selliselt veebilehte hallates kasvab vajalike muudatuste arv koos lehekülgede arvuga ning koos sellega suureneb võimalike vigade oht. Näiteks staatilises HTML keeles kodeeritud veebilehel, kus on 10 alamlehekülge, tuleb ühe menüüpunkti nimes kirjavea parandamiseks kõik 10 faili omale arvutisse salvestada, neid käsitsi muuta ja seejärel uuesti serverisse üles laadida. Sisuhaldussüsteemi kasutades piisab ainult menüüpunkti nime muutmisest ja sisuhaldussüsteem kuvab edaspidi seda menüüpunkti uue nimega. Lisaks on käsitsi failide muutmine mitme autori korral ülimalt keeruline, sest lihtsasti võib tekkida olukordi, kus mitu inimest täiendavad ühte faili samaaegselt ja sellega seoses jääb alles ainult viimasena faili salvestanud isiku muudatus.

Sisuhaldussüsteemis on ka automatiseeritud leheküljele metaandmete genereerimine, RSS uudisvoo moodustamine, veebilehe SEO⁴ parandamine ja lisatud ka sotsiaalmeediaga sidumise võimalus - näiteks uue lehekülje lisamisega teavitatakse Twitteri ja Facebooki kaudu kõiki sealseid kontakte.

Et sisuhaldussüsteem on veebilehekülgede haldajate põhitöövahend, siis on ka väga tähtis selle kasutajasõbralikkus, et võimalikult vähese õppimisega oleks võimalik kodulehekülge hallata. Ettevõtetele on oluline, et sisuhaldussüsteemi kasutamisega saaksid hakkama ka inimesed, kellel puuduvad oskused HTML koodi kirjutamiseks, vähendades seeläbi veebilehe ülalpidamiskulusid. Kuna antud töös räägime vabavaralistest sisuhaldussüsteemidest, siis põhimõtteliselt on võimalik hästi valitud sisuhaldussüsteemi tarkvara puhul hakkama saada ainult serveri rendiks kuluvate summadega.

¹ Veebibrauseri kaudu kuvatav Java virtuaalmasinas töötav rakendus

² *HyperText Markup Language* versioon 5

³ *Scalable Vector Graphics* – XML keeles kirjutatav vektorgraafika

⁴ Search Engine Optimisation

2.1 Avatud lähtekoodiga tarkvarast

Kõik töös hinnatavad sisuhaldussüsteemid on avatud lähtekoodiga - avaldatud GNU Üldise Avaliku Litsentsi (GNU *General Public Licence* ehk GNU *GPL*) all. Seega nende alla laadimine, kasutamine ja modifitseerimine on tasuta ja on lubatud kõigile, kes seda soovivad.

Avatud lähtekoodiga tarkvara kolm põhinõuet on [14]:

- **Tasuta levitus** – Tarkvara litsents ei tohi piirata tarkvara levitamist, vahendamist või müümist mõne teise süsteemi koosseisus kolmandatele osapooltele
- **Lähtekood kättesaadav** – Tarkvara lähtekood peab olema soovijatele kompileerimata ja loetaval kujul ning võimalusel ka kompileeritud kujul ligipääsetav
- **Vabalt modifitseeritav** – Tarkvara litsents peab lubama lähtekoodi oma vajaduste järgi modifitseerimist juhul, kui modifitseeritud kood avaldatakse samade tingimiste alusel

2.2 Kasutatavusest

Kasutatavus (*usability*) on süsteemi omadus, mis näitab kui mugav on süsteemi kasutada [13]. Samuti on süsteemi kasutatavuses suur osa ka funktsionaalsusel, kui süsteemi on väga mugav ja meeldiv kasutada, aga ta ei tee mida ta mõeldud tegema on, siis ei ole kasutajad ikkagi rakenduse kasutatavusega rahul, samas kui süsteem teeb täpselt mida ta tegema peab, aga selle tegevuse läbiviimine on liialt keeruline, siis ei ole kasutajad samuti rahul.

Sisuhaldussüsteemides on hea kasutatavus vajalik produktiivsuse tõstmiseks [1] – ebamugavalt uuendatav veebileht uueneb harvemini, sellega muutub sellel olev info vähemaktuaalsemaks ja see omakorda mõjub halvasti lehekülje külastatavusele. Samuti on hea kasutatavusega sisuhaldussüsteem kergesti õpitav ja seetõttu saavad veebilehe sisu uuendamisega hakkama kõik kellel selleks soovi on, mitte ainult inimesed kellel on HTML kodeerimise oskus.

Hea kasutajaliides on [16]:

- **Kerge õppida** – Süsteemi kasutama õppimine peab võimalikult kerge olema
- **Efektivne kasutada** – Kui süsteemi on õpitud kasutama, siis sellega töötamine on efektiivne
- **Kerge meelde tuletada** – Süsteemi kasutamine peab kergesti meelde jääv olema

- **Vähe vigu** – Süsteemis endas peab olema vähe vigu, et kasutajal ei tekiks segadust tarkvara kasutamisel
- **Meeldiv kasutada** – Tarkvara kasutamine peab olema meeldiv, ehk süsteem peab soodustama töö tegemist

2.3 Võrreldavatest sisuhaldussüsteemidest

Antud töös hindamiseks valitud sisuhaldussüsteemid on kõik PHP⁵ keeles kirjutatud, sest PHP rakendusteks sobilike serverite rentimise hinnad on soodsad ja PHP on üks enamlevinumaid programmeerimiskeeli [3]. Lisaks on võimalik seadistada kõiki valitud sisuhaldussüsteeme MySQL⁶ andmebaasitarkvara kasutama, mida pakutakse vaikimisi koos PHP toega serveritega.

1.1.1. WordPress 3.1

Wordpress (WP)⁷ on alates aastast 2003 [2] rendatav sisuhaldussüsteem, mille viimane versioon avaldati 23. Veebruaril 2011, ja seda on Wordpress.org lehelt alla laetud 9238508 korda [4]. WP eeliseks on tema paindlikkus, seda kasutavad sisuhaldussüsteemina nii suured firmad, kui ka väikeseid ajaveebe pidavad inimesed.

WP arendamisel on silmas peetud kasutajamugavust, W3C veebistandardeid⁸ ja veebilehe sisu semantikat ning see on disainitud töötama kohe pärast serverisse paigaldamist.

WP oli esialgselt mõeldud ajaveebi pidamise lihtsustamiseks, kuid oma populaarsuse ja kasutajamugavuse tõttu hakati seda ka muudel eesmärkidel kasutama ja sealt kasvaski välja praegune versioon, mis on täisväärtuslik sisuhaldussüsteem.

1.1.2. Joomla 1.6.1

Joomla puhul on tegemist haruga MamboCMS projektist, Joomla esimene versioon avalikustati 16. Septembril 2005. Lisaks Joomla loodi samal ajal ka paralleelselt Open Source Matters⁹ mittetulundusühing, mis kaitseb projekti juriidilisi- ja finantshuve.

Joomla eeliseks on väga suur laienduste hulk¹⁰ ja suur kasutajaskond¹¹. See on üles ehitatud pidades silmas moodulitest koosnevat ülesehitust, mis teeb Joomla võimeka sisuhaldussüsteemi.

⁵ PHP: Hypertext Preprocessor. www.php.net

⁶ <http://www.mysql.com/>

⁷ <http://www.wordpress.org>

⁸ <http://www.w3.org/>

⁹ <http://opensourcematters.org/>

¹⁰ <http://extensions.joomla.org/> lehel on 14.05.2011 seisuga 7545 laiendust

1.1.3. Drupal 7.0

Drupali¹² esimene versioon valmis 15. Jaanuaril 2001, projekt sai alguse ühest üliõpilaste sisevõrgust, kus sooviti lihtsat võimalust info ja failide vahetamiseks. Drupali eeliseks teiste sisuhaldussüsteemide ees on samuti suur laienduste hulk ja kasutajaskond.

Lisaks ei ole Drupal ehitatud ainult kasutama MySQL andmebaasitarkvara, vaid toetab lisaks ka SQLite ja PostgreSQL andmebaasitarkvara.

Oma võimsa tuuma ja hea laiendatavuse tõttu on Drupal levinud paljude veebilehtede sisuhaldustarkvarana. Näiteks on see võetud aluseks ka Ameerika Ühendriikide Valge Maja¹³ kodulehe tarkvaras.

1.1.4. CMSMadeSimple 1.9.4.1

CMSMadeSimple (CMSMS) esimene versioon lasti välja Juunis 2004, CMSMS on mõeldud poolstaatiliste informatiivsete lehtede haldamiseks, selle disainimisel on keskendutud ennekõike lihtsusele ja vajaliku funktsionaalsuse esiletõstmiseks. Seetõttu on CMSMS võrreldes teiste eelpoolmainitud sisuhaldussüsteemidega väiksemamahulisem ja väiksemate nõuetega süsteemile.

¹¹ <http://forum.joomla.org/> foorumis on 14.05.2011 seisuga 480324 liiget

¹² <http://drupal.org/>

¹³ <http://www.whitehouse.gov/>

3 Kasutuslood

Kasutuslugude kirjeldamisel on lähtunud vajadusest katta võimalikult suur osa sisuhalduse igapäevaselt vajaminevast funktsionaalsusest. Kasutuslugudes pole süsteemi ülesseadmisega seonduvaid kasutuslugusid, sest enamasti ei pea tavakasutajad sisuhaldussüsteemi serverisse paigaldamisega tegelema ning paigaldamine pole ka igapäevane tegevus siis pole installeerimise mugavus oluline.

Kasutuslugude eesmärgiks on kirjeldada põhitegevused, millega puututakse sisuhalduse tarkvara kasutades sagedasti kokku. Et põhitegevuste mugavus ja lihtsus loobki hea kasutatavuse, siis keskendutakse hindamisel just põhikasutuslugude läbimisele, et hinnata samade tegevuste läbiviimise lihtsust erinevatel sisuhaldussüsteemidel.

3.1 Olemasoleva lehekülje muutmine

Kasutaja soovib juba olemasolevale leheküljele lisada juurde kaks lõiku teksti, need vormindada ja muuta menüü teksti ning lehe pealkirja. Lisaks soovib ta eemaldada leheküljelt ühe foto ning lisada juurde kaks fotot ja ühe video YouTubest kohe lehel mängimiseks (*embedded*).

3.2 Uue lehekülje lisamine

Kasutaja soovib lisada uue lehekülje pealkirjaga “Kontakt” nii et see oleks otse peamenüüst ligipääsetav, lisaks soovib ta et see oleks menüüs järjekorras teine link. Leheküljele lisatakse tabel kolme tulba ja nelja reaga. Enne ja pärast tabelit on kaks lõiku teksti ja igal lõigul oma alampealkiri.

3.3 Menüü restruktureerimine

Kasutaja soovib peamenüüs paigutada kõik menüüd täpselt vastupidisesse järjekorda. Lisaks soovib ta lisada menüüle tekstiga „Kontakt“ alamlehe, kus oleks pilt, millele vajutades avaneks uues aknas Regio kaardirakendus ja pildi all on väikses kirjas juhised õigele korrusele liikumiseks.

3.4 Esilehele peatüki lisamine

Kasutaja soovib lisada esilehele veebilehte tutvustava tekstilõigu ja selle alla pildi.

3.5 Veebilehe jaluses asuva informatsiooni muutmine

Kasutaja soovib veebilehe jaluses asuvat teksti muuta ning lisada sinna hüperlingi „Kontakt“ alammenüüsse. Lisaks soovib ta lisada lingi pealkirjaga “Üles”, mis viitaks sama lehekülje päisesse.

3.6 Uue administreerimislehe kasutaja lisamine

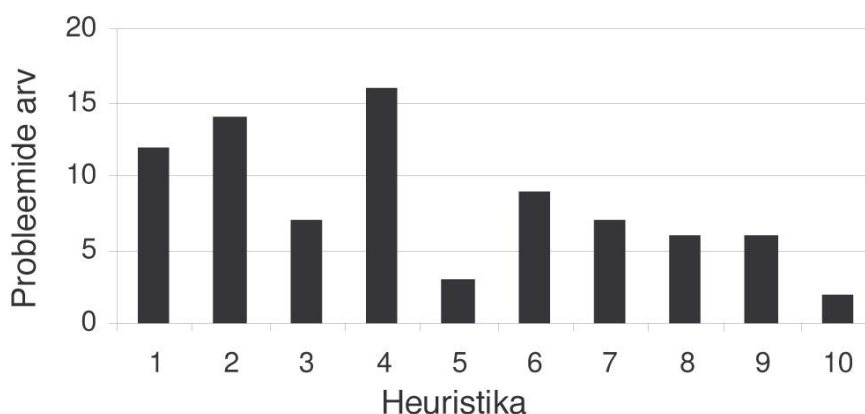
Administraator soovib lisada uue kasutaja sisuhaldussüsteemi administreerimisliidesele. Kasutajal peab olema lehekülgede muutmise ja lisamise õigus, aga uus kasutaja ei tohi olla õigusi kasutajate lisamiseks ega laienduste ja lisade paigaldamiseks, eemaldamiseks ja muutmiseks.

4 Hindamiskriteeriumid

Hindamiskriteeriumid tuginevad Jakob Nielsen'i koostatud kümnel heuristilise kasutatavuse hindamise punktil [7], mida on kohandatud sisuhaldussüsteemide jaoks [5]. Hindamiskriteeriumitele lisatakse juurde punktid, mida tuleks sisuhaldustarkvara kasutatavuse hindamisel silmas pidada ja nende punktide alusel leitud puudujääkide arvust kujuneb iga vaatluse all olevas sisuhaldussüsteemi lõpphinne.

4.1 Kohandatud Nielsen'i kriteeriumid

Sisuhaldussüsteemidele kohandatud hindamiskriteeriumid saadi kasutades algseid Nielsen'i poolt välja töötatud heuristikad ja hinnates selle abil Hollandis üht levinud sisuhaldustarkvara „GX WebManager¹⁴“. Tulemustest selgus, et leitud vigade arv erineb punktide lõikes liialt palju ($M = 8.2$, $sd = 4.3$), sellest järeldati, et kõik punktid ei ole veebipõhiste rakenduste hindamise jaoks adekvaatsed ja otsustati neid täiendada.



Joonis 1 Nielsen'i heuristikate alusel leitud probleemide arv

Algsest kümnest punktist jäi alles kaheksa (1. – 8.) ja uuringu tulemusena lisati juurde neli (9. - 12.). Uute punktide alusel läbi viidud hindamise tulemustes oli leitud vigade arvu kõikumine tunduvalt väiksem ($M = 3.11$, $sd = 1.45$).

4.1.1 Ülevaade süsteemi staatusest

Kasutajad ei saa alati kindlad olla brauseri poolt pakutud tagasisidele, nagu näiteks laadimisriba staatus või muud teavituse viisid Internetiühenduse puudumisest. Kolmandate süsteemide poole pöördumine ei pruugi alati olla edukas ja ka see võib pärssida rakenduse tööd. Süsteem ise peab kasutajaid teavitama mõistliku aja jooksul taustal toimuvate protsesside tulemustest.

¹⁴ <http://www.gxsoftware.com/en/products/web-content-management.htm>

Näiteks peaks süsteem teavitama järgmistest toimingutest:

- Ühendus andmebaasiga on häiritud
- Kasutajasessioon hakkab aeguma
- Taustal toimuvate AJAX¹⁵ päringute toimumine ja nende tulem (kuvada animatsioon, kui käivitub asünkroonne päring, päringu lõppedes kuvada teadet)

4.1.2 Seos rakenduse ja reaalse maailma vahel

Süsteemi keelekasutus peab olema kasutajatele tuttav, soovitatavalt igapäevane ja ei tohi kasutada liialt tehnilisi termineid. Andmed peavad olema esitatud loomulikus ja loogilises järjestuses, ning menüü linkide pealkirjad peavad kirjeldama selle funktsionaalsust. Lisaks peab süsteemis toimuma andmete liikumine (*data flow*) samas järjekorras, mis järjekorras tööülesande teostus toimub, asünkroonse lahenduse puhul peab kasutajale teada olema, et andmete muutmine ei toimu reaalsajas.

- Kasutaja töövoog ja süsteemi töövoog on vastavuses
- Menüü linkide taga on oodatav funktsionaalsus
- Keelekasutus pole tavakasutaja jaoks eksitav ega liialt keeruline

4.1.3 Kasutajaliidese järjekindlus

Samade mõistete, väljendite ja kasutajaliidese elementide kasutamine läbiv kasutamine kogu süsteemis, et kasutajatel ei tekiks segadust süsteemi kasutamisel, kuna läbi süsteemi on kasutatud sarnaste tegevuste läbiviimiseks erinevaid termineid.

- Kasutada samade funktsionaalsuste kirjeldamiseks samu termineid või ikoone
- Läbi süsteemi on kasutusel sarnane kujundus ja visuaalne ülesehitus

4.1.4 Pigem äratundmine kui meenutamine

Süsteem ei tohi koormata kasutajate mälu – kasutaja ei pea korduvat informatsiooni ühelt leheküljelt teisele liikudes meeles pidama. Elementidel kus on võimalik kasutada vaikeväärtusi tuleks neid ka kasutada ja kõik vajalikud tegevused, objektid ja lehekülje osad peavad olema kasutajale kohe näha ja teistest eristatavad ning ligipääsetavad.

- Kasutaja peab teadma kus leheküljel ta asub
- Võimalikult vähe kasutada rippmenüüsid [8]
- Rippmenüüdes kasutada tühjade väärtuste asemel vaikeväärtusi

¹⁵ Asünkroonne JavaScript ja XML

4.1.5 Paindlikkus ja kasutamise efektiivsus

Kasutaja peab saama kohandada kasutajaliidest oma eelistuste järgi, näiteks seadistatava abimenüü kaudu ja klaviatuuri kiirklahvikombinatsioone määrates. Kasutajaliidese kohandamine võimaldab peita vähem vajalikke funktsioone ja esile tuua rohkem kasutatavaid võimalusi. Kiirklahvid peaksid vastama üldlevinud klahvikombinatsioonidele, et vähendada kasutajate õppimisperioodi, aga samas ei tohi häirida brauserite vaikimisi kiirklahvikombinatsioone.

Kasutajakogemusele annab juurde ka lohistamisvõimaluse (*drag and drop*) lisamine operatsioonisüsteemi ja sisuhaldustarkvara vahele, kasutades selleks kas Adobe Flash¹⁶ raamistikku või HTML5¹⁷ võimalusi.

Mugavaks teksti sisestamiseks peaks rakenduses kasutusel olema „mida näed, seda saad“ (WYSIWYG) teksti redigeerimise vahend, lisaks peaks see olema kohandatav, et peita ebavajalikud või mittesoovitavad võimalused.

Efektiivse kasutajaliidese osa on ka selle kiirus, veebilehe laadimisajad peavad olema võimalikult lühikesed, suuremamahuliste leheküljeosade sisselugemisel tuleb kasutada asünkroonset laadimist või järk-järgulist laadimist, et kasutaja saaks võimalikult kiiresti oma tegevusega alustada [9].

- Menüüd on kohandatavad ja ümberpaigutatavad
- Ekspertkasutajatele on võimaldatud kiirklahvikombinatsioonide kasutamine
- Lehekülje laadimisajad peavad olema võimalikult kiired [12]
 - Kuni 0.1 sekundit on aeg mil kasutaja jaoks toimub lehekülje laadimine hetkeliselt
 - Kuni 1 sekund on aeg kui kasutaja jätkab veel oma tegevusi häirimatult
 - Alates 10 sekundist hakkab kasutaja tähelepanu hajuma ja lehekülje laadimisajad on tõsine probleem kasutatavusele.

4.1.6 Esteetiline ja minimalistlik ülesehitus

Tuleb hoiduda kasutajale kuvamast ebaolulist või harva tarvilikku informatsiooni, üleliigse informatsiooni kuvamine tõmbab kasutaja tähelepanu oluliselt ja vähendab seeläbi kasutatavust. Samas tuleb hoiduda lehekülgede liiga inforikkaks muutmisest, vajadusel paigutada informatsioon mitmele leheküljele või kasutada viisardeid (*wizard*). Laialdase

¹⁶ <http://www.adobe.com/products/flash.html>

¹⁷ <http://dev.w3.org/html5/spec/Overview.html>

funktsionaalsusega rakendusevaatest luua nii tavakasutaja kui ka ekspertkasutaja vaated, et tavakasutajat kaitsta potentsiaalsete veakohtade ja peita ebavajalike valikuvõimalusi, samas võimaldades lihtsat ligipääsu laiendatud funktsionaalsusele.

- Pikad sisestusvormid jaotada mitmesse ossa
- Viia kasutajale kuvatava informatsiooni hulk miinimumini
- Kiire ekspertkasutaja ja tavakasutaja vaadete vaheline ümberlülitamine ning selle käigus juba sisestatud informatsiooni säilitamine

4.1.7 Vigade ennetamine ja nendest taastumine

Kasutajatele tuleb tagada alati varuväljapääs, näiteks võimalus pöörduda tagasi lehekülje eelmise versiooni juurde. Pöördumatute tegevuste puhul tuleb kasutajat teavitada, et tehtav tegevus ei ole tagasipööratav ja võib põhjustada andmekadu.

Veatekstit tuleb kuvada nende elementide juurde, mis antud vea põhjustasid ja võimalusel pakkuda juhiseid vea parandamiseks. Kuvatavad veatekstit peab hoidma lihtsate ja üheselt mõistetavatena, aga need ei tohi sisaldada üleliigset tehnilist informatsiooni.

Samuti tuleb kaitsta süsteemi kasutajatepoolsete vigade eest, näiteks ei tohiks vale sisendi korral süsteem lakata töötamast (SQL¹⁸ injektsioon või mõne muu programmi teksti käivitamine).

- Kasutajale on üheselt selge mis vea põhjustas
- Kasutaja tehtud viga ei tohi süsteemi halvata
- Vormi esitamisel tehtud viga ei tohi kaotada sisestatud informatsiooni.
- Sisestusvälju peab alati süsteemi poolt valideerima
- Salvestada kasutaja poolt sisestatud andmeid perioodiliselt, et ära hoida brauseri sulgemise või tõrkumise korral tekkivat andmekadu

4.1.8 Abi ja juhendite kättesaadavus

Alati tuleb tagada kasutusjuhendite ja võimalusel kontekstist sõltuva abi kättesaadavus. Võimaluse korral tuleb integreerida abitekstit otse kasutajaliidesesse (vormiväljade juurde käivad seletused ja töövoos selgitused) ja need tuleb selgelt eristada muust lehe funktsionaalsustest (abimenüüde juures teistmoodi hiirekursor, teine värvilahendus, erinevad ikoonid).

- Igal vaatel on võimalus avada selle vaate abileht

¹⁸ Structured Query Language

- Lisada töövoogu abistavad selgitused
- Vormielementide pealkirjade kohal lisatekstina kuvatav info selle välja kohta

4.1.9 Teiste rakendustega sarnane ülesehitus

Sisuhaldussüsteemi ülesehitus peaks jälgima üldiseid kasutajaliidese disainimise reegleid (nii veebirakenduste kui ka rakendusprogrammide) ja töövoogusid. Kui jäljendatakse rakendustarkvara kujundusmalle, siis peaks neid jälgima terviklikult – võimalikult palju jäljendatava tarkvara võimalusi peaks säilima ja kui nende omaduste saavutamine tavavahenditega pole võimalik, siis kasutada brauseri laiendusi.

- Kasutada standardseid ikoone [10]
- Hoiduda kasutamast mittestandardseid kasutajaliidese lahendusi
- Kasutajamugavuse suurendamiseks kasutada brauseri laiendusi - mitme faili üleslaadimise võimaldamine, lohistamine (*drag and drop*), Java Appletiga või Adobe Flashiga realiseeritud failihaldur

4.1.10 Jälgi veebirakenduste kujundustavasid

Vormide ülesehitamiseks tuleb kasutada standardseid vormielemente nagu raadionupud (*radio button*), märkeruudud (*checkbox*), nupud (*button*), vältida hüperlinkide puhul topeltkliki kasutamist ning hüperlingid selgelt muust tekstist eristada, eriti kui hiire kursor on lingi kohal (*hover*) või kui link on fookuses (*focus*).

- Hüperlingid on muust tekstist selgelt eristatud ja neil on määratud erinev kujundus külalstatud (*visited*) lingi korral ja hiirega ülelibisemise (*hover*) korral
- Vormide ehitusel on jälgitud W3C standardeid [11]
- Navigeerimisel ei tohi kasutada topeltklikki
- Üleliigse info vältimiseks kuvada selgitavaid tekste elemendilt hiirega ülelibisemise korral

4.1.11 Brauseri tööriistade funktsioneerimine

Veebirakendused töötavad internetibrauseris ja seega tuleb arvestada sellega, et brauseri menüüd ja navigatsioonielemendid võivad enda alla võtta märgatava osa kasutaja ekraanipinnast. Lisaks tuleb võimalikult selgelt eristada rakenduse menüüd brauseri menüüdest. Samuti tuleb tagada brauseri põhifunktsioonide toimimine (Edasi ja tagasi liikumise nuppude töö tagamine, ekraani ja teksti suurendamise võimalus brauseri poolt).

- Edasi (*Forward*) ja tagasi (*back*) liikumise nupud peavad töötama

- Leheküljed peavad olema järjehoidjatesse lisatavad (*bookmarkable*)
- Brauseris teksti suurendamine peab toimima
- Rakenduse menüüd ei tohi sarnaneda kasutaja operatsioonisüsteemi omadega, et vältida segaduse tekkimist

4.1.12 Lubada lihtsat andmete sisestamist, mis vähendaks veaolukordi

Kuna andmete sisestamine on sisuhaldussüsteemi põhiosa, siis peaks see olema kasutajale mugavalt lahendatud. Süsteem peab võimaldada andmeid sisestada eri formaatides ja võimalusel kuvama andmete sisestamise formaadi – telefoninumbri formaat, aadressi esitamise kuju - või pakkuma andmete valimiseks intuitiivsemaid võimalusi – näiteks kuupäeva valikul kuvada kalender ja värvitooni valikul kuvada värvipalett sellelt värvitooni valimise võimalusega. Lisaks peavad olema kohustuslikud vormi väljad selgelt eristatavad mittekohustuslikest väljadest.

- Vabateksti sisestamiseks on kasutusel „mida näed, seda saad“ tekstiredaktor
- Kuupäevade valikul kasutada kalendrit
- Võimalusel kasutada vormide eelvalideerimist JavaScripti¹⁹ abil
- Eristada kohustuslikud väljad mittekohustuslikest ja hoida mittekohustuslike väljade arvu minimaalsena

¹⁹ <https://developer.mozilla.org/en/JavaScript>

5 Hindamine

Kasutame peatükis 4.1 loetletud kohandatud Nielsen'i heuristikaid vastavatel sisuhaldussüsteemidel. Leitud kasutatavuse probleemid kategoriseerime täiendatud Nielsen'i heuristikate nimekirja alusel, tuues iga heuristika juurde vastava sisuhaldustarkvara tugevused ja nõrkused. Nõrkusteks märgitakse kasutuslugude läbimisel märgatud vastuolusid hindamispunktidega. Tugevusteks märgitakse vastavusi hindamispunktides kirjeldatuga ja lisaks tähelepanekud, mis muutsid kasutuslugude läbimist võrreldes teiste süsteemidega lihtsamaks või mugavamaks.

Hindamise käigus läbitakse kõik kirjeldatud kasutuslood, et ühtsete kriteeriumite alusel hinnata sama funktsionaalsust. Kui mõne kasutusloo edukaks täitmiseks on vaja pöörduda sisuhaldussüsteemi abilehtede poole, siis on antud süsteemil tõsiseid kasutatavuse probleeme.

Hindamise käigus leitud probleemide koguarv on toodud iga sisuhaldusesüsteemi hindamise peatüki alguses. Tabelisse on märgitud kõik leitud vead olenemata nende tõsidusest. Suuremad probleemid on välja toodud iga süsteemi kohta peatükis 5.5

5.1 CMSMadeSimple 1.9.4.1

Tabel 1 CMSMadeSimple probleemide arv

Heuristika number:	1	2	3	4	5	6	7	8	9	10	11	12	Kokku
Leitud vigade arv:	2	0	1	1	3	0	2	2	1	0	1	2	15

Tabel 2 CMSMadeSimple hinnang punktide lõikes

1. Ülevaade süsteemi staatusest	
Head: <ul style="list-style-type: none">Kuvatakse veateade kui ühendus andmebaasiga on katkenudSüsteemi poolt on teavitused uuenduste olemasolust ja vigadest	Vead: <ul style="list-style-type: none">Vigade esinemisel kuvatakse vaikimisi PHP veateatedFailide üleslaadimisel ei kuvata teadet õnnestumisest ja ebaõnnestumisest
2. Seos rakenduse ja reaalse maailma vahel	
Head: <ul style="list-style-type: none">Keelekasutus on lihtneMenüüde ülesehitus on loogilineMenüü tekstide taga peituv funktsionaalsus on oodatav	Vead: <p>Pole</p>

3. Kasutajaliidese järjekindlus	
Head:	Vead:
<ul style="list-style-type: none"> Keelekasutus on ühtlane 	<ul style="list-style-type: none"> Vaskpoole abimenüü pole igas vaates nähtav ja see on ainus koht kus on lingid abilehtede ja foorumitele
4. Pigem äratundmine kui meenutamine	
Head:	Vead:
	<ul style="list-style-type: none"> Kiirmenüüd on rippmenüüdena, ja selle kaudu toimingute tegemine säästab umbes 2 hiireklikki.
5. Paindlikkus ja kasutamise efektiivsus	
Head:	Vead:
<ul style="list-style-type: none"> Peamenüü on kohandatav Võimalik luua oma kiirmenüü Lehekülje lisamise lisavalikud on kergesti ligipääsetavad ja võimalusterohked 	<ul style="list-style-type: none"> Suure ekraaniresolutsiooni korral on ekraanikasutus ebaefektiivne Poolelioleva lehe sisestamise vormilt ära liikudes ei hoiatata võimaliku andmekao eest Piltide haldamine on tülikas
6. Esteetiline ja minimalistlik ülesehitus	
Head:	Vead:
<ul style="list-style-type: none"> Uue lehekülje lisamisel on lisavalikud ära peidetud ja kuvatakse ainult vajalikke vormivälju 	Pole
7. Vigade ennetamine ja nendest taastumine	
Head:	Vead:
	<ul style="list-style-type: none"> Vormil tehtud vead võiks olla vigaste elementide juures Õnnestus SQL injektsiooni teostada
8. Abi ja juhendite kättesaadavus	
Head:	Vead:
	<ul style="list-style-type: none"> Ei ole linki abilehele Ei ole integreeritud abi
9. Teiste rakendustega sarnane ülesehitus	
Head:	Vead:
	<ul style="list-style-type: none"> Lisavormid avanevad kihina kujunduse peal ja meenutavad oma väljanägemiselt süsteemiaknaid
10. Jälgi veebirakenduste kujundustavasid	
Head:	Vead:
<ul style="list-style-type: none"> On edukalt kasutatud sakkidega (<i>tabbed</i>) navigatsiooni 	Pole

11. Brauseri tööriistade funktsioneerimine	
Head:	Vead: <ul style="list-style-type: none"> Alammenüü sakkide vahel liikumine ei kajastu ajalooos – brauseri Edasi ja Tagasi nupud ei tööta igas vaates
12. Lubada lihtsat andmete sisestamist, mis vähendaks veolukordi	
Head:	Vead: <ul style="list-style-type: none"> Tekstiredaktor on vähevõimekas Kohustuslike väljade märkimine pole pidev

5.2 Drupal 7.0

Tabel 3 Drupali probleemide arv

Heuristika number:	1	2	3	4	5	6	7	8	9	10	11	12	Kokku
Leitud vigade arv:	1	2	3	0	2	1	2	4	2	1	1	1	20

Tabel 4 Drupali hinnang punktide lõikes

1. Ülevaade süsteemi staatusest	
Head: <ul style="list-style-type: none"> Andmebaasiühenduse puudumise korral kuvatakse teade Asünkroonsete päringute toimumine ja tulem on selgelt näha 	Vead: <ul style="list-style-type: none"> Süsteemivead on peitmata ja põhjuse leidmine raskendatud
2. Seos rakenduse ja reaalse maailma vahel	
Head: <ul style="list-style-type: none"> Menüüpunktide taga peituv funktsionaalsus on arusaadav 	Vead: <ul style="list-style-type: none"> Et administreerimisliides on integreeritud süsteemi, on selle kasutamine alguses harjumatu Keelekasutus on kohati liialt tehniline
3. Kasutajaliidese järjekindlus	
Head: <ul style="list-style-type: none"> Keelekasutus on ühtlane Veateated ja infoteated on visuaalselt selgesti eristatavad 	Vead: <ul style="list-style-type: none"> Teadete tekst ei muutu kui administreerimisaken suletakse ja selle all on eelnevast toimingust jäänud teade Vahel avaneb menüü kihina rakenduse kohal, vahel suunatakse uuele lehele Kasutusel olev sakkidega (<i>tabbed</i>) navigatsioon ei jälgi sakkide kasutamise loogikat [15]

4. Pigem äratundmine kui meenutamine	
Head:	Vead:
<ul style="list-style-type: none"> Asukoht administreerimislehel on selgesti arusaadav 	<ul style="list-style-type: none"> Pole
5. Paindlikkus ja kasutamise efektiivsus	
Head:	Vead:
<ul style="list-style-type: none"> On võimalik kiirmenüüd kohandada Kiirklahvikombinatsioonide kasutamise võimalus Olles administraatori kasutaja alt sisselogitud, saab kiiresti liikuda lehelt vastava sektsiooni muutmise lehel, kiirendab töövoogu 	<ul style="list-style-type: none"> Lehekülje laadimisajad võivad pikaks venida kui avatakse suuri lehti Puudub vaikimisi piltide üleslaadimise funktsionaalsus
6. Esteetiline ja minimalistlik ülesehitus	
Head:	Vead:
<ul style="list-style-type: none"> Menüüde osa ekraanist on viidud miinimumini Väikese ekraaniresolutsiooniga ekraanide puhul on kujundus kasutatav Lisafunktsionaalsus on peidetud, kuvades ainult vajalikku osa lehest 	<ul style="list-style-type: none"> Vormielemendid ja nende vahed on liialt suured
7. Vigade ennetamine ja nendest taastumine	
Head:	Vead:
<ul style="list-style-type: none"> Lehekülje kustutamise korral hoiatatakse kasutajat et tema poolt tehtav tegevus pole tagasipööratav 	<ul style="list-style-type: none"> Brauseriakna sulgemise korral ei hoiatata kasutajat Lehekülje muutmise lehelt ära liikudes ei salvestata muudatusi ega hoiatata kasutajat
8. Abi ja juhendite kättesaadavus	
Head:	Vead:
	<ul style="list-style-type: none"> Abi võimalus on väga vähestele osadele lisatud Abi avamine suunab lehelt ära ja pooleli olev töö kaob Abi pole põhjalik Abi sees viitavad lingid rakenduse administreerimise vormidele
9. Teiste rakendustega sarnane ülesehitus	
Head:	Vead:
	<ul style="list-style-type: none"> Teiste SHS-idega erinev ülesehitus Nimekirjadel massioperatsioone tehes on tegevuse valimine ja kinnitamine enne nimekirja ja eraldi konteineri sees

10. Jälgi veebirakenduste kujundustavasid	
Head:	Vead: <ul style="list-style-type: none"> • Tekstisisesed külastatud hüperlingid ei muuda värvi kui neid on külastatud
11. Brauseri tööriistade funktsioneerimine	
Head: <ul style="list-style-type: none"> • Lehekülgi on võimalik järjehoidjasse lisada 	Vead: <ul style="list-style-type: none"> • Edasi nupp brauseritel ei tööta teatud lehekülje osades
12. Lubada lihtsat andmete sisestamist, mis vähendaks veaolukordi	
Head: <ul style="list-style-type: none"> • Rippmenüüdes on vaikeväärtused valitud • Keeruliste formaatidega sisestusväljade juures on formaat ette antud • Värvivalik on lahendatud interaktiivse värvipaletiga 	Vead: <ul style="list-style-type: none"> • Puudub vaikimisi tekstiredaktor

5.3 Joomla 1.6.1

Tabel 5 Joomla probleemide arv

Heuristika number:	1	2	3	4	5	6	7	8	9	10	11	12	Kokku
Leitud vigade arv:	1	3	4	3	6	3	2	3	2	2	2	2	33

Tabel 6 Joomla hinnang punktide lõikes

1. Ülevaade süsteemi staatusest	
Head: <ul style="list-style-type: none"> • Andmebaasi vea korral teavitatakse kasutajat • On süsteemi olemas lehekülge kus kuvatakse detailselt süsteemi staatust 	Vead: <ul style="list-style-type: none"> • Suurte failide üleslaadimisel ei näidata progressiriba.
2. Seos rakenduse ja reaalse maailma vahel	
Head:	Vead: <ul style="list-style-type: none"> • Rakenduse ülesehitusest on raske aru saada • Keelekasutus on segane • Uue lehekülje lisamine ilma abilehte lugemata ebaõnnestus

3. Kasutajaliidese järjekindlus	
<p>Head:</p> <ul style="list-style-type: none"> Kasutajaliidese on kasutatud sama ülesehitust 	<p>Vead:</p> <ul style="list-style-type: none"> Rakenduses on kerge ära eksida Olles vormil, muudetakse kogu peamenüü mitteaktiivseks Vormide ja tabelite kasutamise olukorrad ja ülesehitused pole süsteemi läbivalt ühesed Kuvatakse vormielemente, millel puudub reaalne funktsioon
4. Pigem äratundmine kui meenutamine	
<p>Head:</p>	<p>Vead:</p> <ul style="list-style-type: none"> Uue lehekülje lisamisel ei pakuta selle lisamist menüüsse Rippmenüüdes on punkt „Kasuta vaikeväärtus“, kasutajal pole teada mis see vaikeväärtus on Vormi pealkirjad on väheütlevad ja menüüst pole asukoht üheselt tuvastatav
5. Paindlikkus ja kasutamise efektiivsus	
<p>Head:</p> <ul style="list-style-type: none"> Ebavajalikke menüüblokke saab peita Lehekülje laadimisajad on kiired 	<p>Vead:</p> <ul style="list-style-type: none"> Menüüd pole kohandatavad Tekstiredaktori võimekus on väike Puudub kiirklahvikombinatsioonide kasutamise võimalus Akordion tüüpi menüüsid ei saa kõiki lahti teha, uue avamine sulgeb eelmise menüü Lehekülje lisamine ja selle menüüs näitamine on aeganõudev ja keeruline Lehekülje paigutus pole kohandatav
6. Esteetiline ja minimalistlik ülesehitus	
<p>Head:</p> <ul style="list-style-type: none"> Ekraanipinna kasutus vormidel on väike 	<p>Vead:</p> <ul style="list-style-type: none"> Sisestusvormidel kuvatakse liialt mittekohustuslikke välju Vajalike toimingute tegemiseks on vaja liikuda lisamenüüdesse Menüüde osa ekraanist on liialt suur

7. Vigade ennetamine ja nendest taastumine	
<p>Head:</p> <ul style="list-style-type: none"> • Laialdaselt kasutatud JavaScripti abil vormide valideerimist 	<p>Vead:</p> <ul style="list-style-type: none"> • Uue kasutaja lisamisel emaili saatmine ebaõnnestus, kuigi süsteem lisas kasutaja – seda kasutajat ei saa ilma andmebaasis muudatusi tegemata aktiveerida • Õnnestus vigase sisendiga süsteemi mooduli töö halvata
8. Abi ja juhendite kättesaadavus	
<p>Head:</p> <ul style="list-style-type: none"> • Vormielementide juures kuvatakse abi 	<p>Vead:</p> <ul style="list-style-type: none"> • Abi leht avaneb hüpikaknas ja see blokeeritakse brauseri poolt • Kontekstiga seotud abi on lühike • Administreerimisliidese peamenüüs olev abileht on mitteaktiivne uue lehekülje sisestamise ajal
9. Teiste rakendustega sarnane ülesehitus	
<p>Head:</p>	<p>Vead:</p> <ul style="list-style-type: none"> • Rakenduse toimimise loogika ei ole teiste programmidega sarnane • Icoonid on segadustekitavad
10. Jälgi veebirakenduste kujundustavasid	
<p>Head:</p> <ul style="list-style-type: none"> • Hüperlingid on selgelt muust tekstist eristatud • Informatiivne tekst üksikelementide kohta on lisatud infokasti, mis ilmub elemendi kohale kui sellest hiirega üle libiseda 	<p>Vead:</p> <ul style="list-style-type: none"> • Administreerimisliidese peamenüü lingid pole lingid, vaid nendele liikudes avanevad rippmenüüd • Alammenüüd pole selgesti eristatavad
11. Brauseri tööriistade funktsioneerimine	
<p>Head:</p> <ul style="list-style-type: none"> • Lehekülgi on võimalik järjehoidjatesse lisada 	<p>Vead:</p> <ul style="list-style-type: none"> • Vormil muudatuste tegemine rikub Edasi ja Tagasi liikumise nuppude töö • Brauseri sulgemise korral ei hoiatada võimalikust andmekaost
12. Lubada lihtsat andmete sisestamist, mis vähendaks veaolukordi	
<p>Head:</p> <ul style="list-style-type: none"> • Kuupäevade valikuks on kasutusel kalender 	<p>Vead:</p> <ul style="list-style-type: none"> • Vormiväljade juures pole formaate välja toodud • Tekstiredaktor ei näita teksti vormindamise stiili

5.4 WordPress 3.1

Tabel 7 Wordpressi probleemide arv

Heuristika number:	1	2	3	4	5	6	7	8	9	10	11	12	Kokku
Leitud vigade arv:	3	4	4	0	2	3	4	1	1	2	1	3	28

Tabel 8 Wordpressi hinnang punktide lõikes

1. Ülevaade süsteemi staatusest	
<p>Head:</p> <ul style="list-style-type: none"> Kohe sisse logides on näha ülevaade viimastest toimingutest Faile üles laadides on kuvatud edenemisriba Asünkroonsete päringute käivitamine on näha 	<p>Vead:</p> <ul style="list-style-type: none"> Kui sisseloginuna kaob ära baasiühendus, siis muutub rakendus töökõlbmatuks Kui Wordpressi serverid pole kättesaadavad, siis administreerimis-lehel ei laeta kõiki tekstiplokke Asünkroonse päringu ebaõnnestumisest ei teavitata
2. Seos rakenduse ja reaalse maailma vahel	
<p>Head:</p> <ul style="list-style-type: none"> Lehekülgede lisamine on lihtne ja loogiline Linkide pealkirjad on informatiivsed ja nende taga peituv funktsionaalsus on etteaimatav 	<p>Vead:</p> <ul style="list-style-type: none"> Lehekülje järjekorra on liialt keeruline Kohati on kasutatud funktsionaalsuste nimedes vähelevinud väljendeid Esilehe muutmine ilma abilehte lugemata ebaõnnestus Lehekülje jalusesse teksti lisamine ebaõnnestus
3. Kasutajaliidese järjekindlus	
<p>Head:</p> <ul style="list-style-type: none"> Kasutajaliidese ülesehitus on ühtne 	<p>Vead:</p> <ul style="list-style-type: none"> Veateated ja infoteated on ühte värvi Läbisegi on kasutatud termineid <i>Trash</i> ja <i>Delete</i> <i>Cancel</i> vahel tühjendab sisestatud väljad, vahel viib vormilt mujale Päise ja jaluse muutmine käib eri meetoditega
4. Pigem äratundmine kui meenutamine	
<p>Head:</p> <ul style="list-style-type: none"> Menüüst on selgelt näha, kus lehekülge osal asutakse 	<p>Vead:</p> <p>Pole</p>

5. Paindlikkus ja kasutamise efektiivsus	
<p>Head:</p> <ul style="list-style-type: none"> • On võimalus kiirklahve seadistada • Võimalus lehekülje laotust oma vajaduste järgi kohandada • Tekstiredaktoris on võimalik vähekasutatavaid funktsioone peita 	<p>Vead:</p> <ul style="list-style-type: none"> • Laadimisajad on kohati pikad liigse AJAX-i tõttu • Kiirmenüüd on rippmenüüdena, ega kiirenda kasutamist
6. Esteetiline ja minimalistlik ülesehitus	
<p>Head:</p> <ul style="list-style-type: none"> • Võimalik peita ebavajalikke osi • Sisestusväljade arv on viidud miinimumini 	<p>Vead:</p> <ul style="list-style-type: none"> • Ekraanipaigutus on liialt laiutav • Väikestel resolutsioonidel kaovad osa tekstiredaktori nuppudest küljemenüü taha • Pikad abitekstid vormielementide juures tõmbavad liialt tähelepanu
7. Vigade ennetamine ja nendest taastumine	
<p>Head:</p> <ul style="list-style-type: none"> • Vormilt ära liikudes või brauseri akna sulgedes teavitus kui on oht andmekaole • Mitme sisestusväljaga vormidel värvitakse vea tekitanud vormiväli punaseks 	<p>Vead:</p> <ul style="list-style-type: none"> • Postituse lisamärgete lisamisel ei teavitata kui aadressitekst ei sobi • Ebasobiva vormiväärtuse lehekülje nime määramisel ignoreeritakse seda • Tühja failinime sisestamine halvas failihalduri töö • Vaikimisi ei salvestata taustal pooleliolevat artikli sisestamist
8. Abi ja juhendite kättesaadavus	
<p>Head:</p> <ul style="list-style-type: none"> • Igal leheküljel on kuvatud seda lehekülge puudutav abimenüüd ja lingid pikematele abitekstidele • Vormiväljade juurde on lisatud seletavad tekstid 	<p>Vead:</p> <ul style="list-style-type: none"> • Avanevatel menüüdel, milles on lisafunktsionaalsus, puudub abi kuvamise võimalus.
9. Teiste rakendustega sarnane ülesehitus	
<p>Head:</p> <ul style="list-style-type: none"> • Failide üleslaadimisel on kasutatud Flash rakendust, et võimaldada mitme faili korraga serverisse laadimine 	<p>Vead:</p> <ul style="list-style-type: none"> • Alamvormide salvestamiseks peab peavormi salvestama.
10. Jälgi veebirakenduste kujundustavasid	
<p>Head:</p> <ul style="list-style-type: none"> • Vormide sildid (<i>label</i>) viivad fookuse sisestusväljale • Lingid on selgelt eristatud 	<p>Vead:</p> <ul style="list-style-type: none"> • Külastatud lingid on sama värvi • Nuppude (<i>button</i>) kujundus ei ole ühtne

11. Brauseri tööriistade funktsioneerimine	
<p>Head:</p> <ul style="list-style-type: none"> • Brauseri funktsionaalsust ei ole pärsitud rakenduse poolt 	<p>Vead:</p> <ul style="list-style-type: none"> • Leidub funktsionaalsuseid, mis kasutavad hüpikaknaid, mis püütakse brauseripoolse turvalisusmeetmete abil kinni
12. Lubada lihtsat andmete sisestamist, mis vähendaks veaolukordi	
<p>Head:</p> <ul style="list-style-type: none"> • Kohustuslikud väljad on tärniga eristatud • Tekstiredaktor on „mida näed, seda saad“ tüüpi 	<p>Vead:</p> <ul style="list-style-type: none"> • Kuupäevade sisestamine on kohati käsitis – kuigi formaat on ette antud • Tekstiredaktori võimalused on piiratud • Piltide lisamine artiklile on segadusse ajav

5.5 Hindamise kokkuvõte

Kõige vähem kasutatavuse vigu tuli välja CMSMadeSimple²⁰ hindamisel, selle põhjuseks on antud rakenduse lihtsus ja kompaktsus. Võrreldes teiste hinnatavate süsteemidega on CMSMS tunduvalt võimalustevaesem ja selles ei ole vajalik võimaldada lihtsat ligipääsu suurele funktsionaalsusele. Suurimaks veaks antud süsteemi puhul oli integreeritud abi puudumine, kuigi CMSMS kodulehel on kõik vajalik materjal kättesaadav.

Wordpressist tulnud suur vigade hulk tuleneb sellest, et tegu oli algselt ajaveebimootoriga ja seetõttu pole ta disainitud sisuhaldussüsteemiks vaid on selleks arenenud, hoolimata paljudest leitud vigadest oli Wordpressi kasutamine kõige meeldivam, sest selle kasutajaliides oli kõige modernsem ja seal oli kasutatud palju kasutajakogemust parandavaid lahendusi. Wordpressi puhul oli suurimaks puuduseks artiklite ja lehtede paralleelne kasutus, ilma süvenemata oli neil keeruline vahet teha. Lisaks oli võimalik piltide haldamise moodul kergesti kasutuskõlbmatuks muuta.

Joomlal esines väga suuri kasutatavuse probleeme, põhiprobleemiks oli töövoogu oluline erinevus teistest süsteemidest ja seetõttu oli Joomla kasutama õppimine kõige vaevalisem. Lisaks olid põhivormid liialt keerulised ja oluline funktsionaalsus oli peidetud kõrvalmenüüdesse.

Drupal erines teistest hinnatavatest süsteemidest oma teistsuguse lähenemisega rakenduse haldusliidesesse – kasutusel ei olnud mitte eraldi lehekülg, vaid administreerimise osa oli integreeritud kujundusse. See lahendus on esmapilgul harjumatu, aga muutis soovitud

²⁰ CMSMS

lehekülgede sisu muutmise väga mugavaks. Drupal oli ka ainus süsteem mille tööd ei õnnestunud viigase sisendiga halvata. Samas ei ole Drupaliga vaikimisi kaasas tekstiredaktorit ja seetõttu võib HTML keelt mitteoskavale isikule Drupali kasutamine liialt keeruline tunduda.

Enim probleeme esines punktides:

- Kasutajaliidese järjekindlus – keskmine 3
- Paindlikkus ja kasutamise efektiivsus – keskmine 3.25
- Vigade ennetamine ja nendest taastumine – keskmine 2,5
- Abi ja juhendite kättesaadavus – keskmine 2,5

Kasutajaliidese ebapidevus ja abi puudumine tuleneb suurtest arendajate hulgast, mistõttu puudub tugev terviknägemus rakenduse väljanägemisest. Kõige enam kasutajat häirivad probleemid tulenevad punktist „Vigade ennetamine ja nendest taastumine“, eriti olukord, kus vormil tehtud viga halvab süsteemi töö, analüüsi käigus õnnestus halvata ebastandardse sisendiga kolme süsteemi töö.

Tulemused ühe tabelina ja koos keskmistega on välja toodud Lisas 1 Tabel 9 Analüüsi tulemused ühes tabelis koos keskmistega.

Kokkuvõte

Käesolevas töös võrreldi vabavaraliste sisuhaldussüsteemide kasutajasõbralikkust nelja levinud sisuhaldussüsteemil, kasutades sisuhaldussüsteemide jaoks kohandatud J.Nielsseni poolt välja töötatud kasutatavuse heuristikaid.

Antud töö eesmärk oli näidata, et vabavaralistel sisuhaldussüsteemide arendamisel on liiga vähe tähelepanu pööratud kasutatavusele. Kõikidel töös analüüsitud süsteemidel esines kasutatavuse probleeme, enamus leitud probleemidest ei olnud otseselt kasutajate tööd segavad, küll aga leidus kõikides vaadeldud süsteemides peale Drupali süsteemivigu, mis võimaldasid teatud komponentide töö halvata.

Põhilised probleemid olid mittejärjepidev kasuajaliides - mis võib kasutajad kergesti segadusse ajada ning nad võivad süsteemis „ära eksida“. Kasutajale mõeldud abi halb kvaliteet või raske ligipääsetavus - kõigil süsteemidel peale CMSMadeSimpeli oli abi kergesti ligipääsetav, aga abi kvaliteet oli komponentide lõikes kõikuv ja abitekstides olev keelekasutus oli tehniline. Samuti oli kasutajaliidese kohandamine liialt kasin, et töö efektiivsuse tõstmiseks pakutavast funktsionaalsusest kasu oleks. Joomla ja Wordpressi analüüsis ei olnud võimalik kasutuslugusid ilma abi lehtedelt abi otsimata läbida, see näitab et töövood on võrreldes teiste süsteemidega liialt keerulised.

Usability of open source content management systems

Bachelor thesis (6 EAP)

Taavi Pauskar

Abstract

This Bachelor thesis contains the usability evaluation of four common open source content management systems (Wordpress 3.1, Joomla 1.6.1, Drupal 7.0 and CMSMadeSimple 1.9.4.1). The purpose of the thesis was to point out common open source content management systems usability problems.

In the beginning everyday use-cases were created to have common tasks to perform on all the systems. Then evaluation criteria were created based on the CMS specific usability heuristics. And finally the heuristics based evaluation was performed by completing the use-cases. Usability errors were written down and then categorized under the evaluation points.

As the results of the evaluation showed, there were a few heavy violations of the heuristics and quite a lot minor problems. At the end we concluded that open source content management systems have many usability problems.

Kirjandus

- [1] J. Robertson, "11 usability principles for CMS prod" Mai 2007.
http://www.steptwo.com.au/papers/kmc_usabilitycms/index.html [14.05.2011]
- [2] "WordPress, Roadmap."
<http://wordpress.org/about/roadmap/> [14.05.2011]
- [3] "TIOBE Software, Tiobe Index."
<http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html> [14.05.2011]
- [4] "WordPress Download Counter."
<http://wordpress.org/download/counter/> [14.05.2011]
- [5] R. Bos, J van Gorp, J. H. Verpoorten, S. Brinkkermper, "Heuristic evaluation of content management systems: CMS specific heuristics" 2005.
- [6] J. Nielsen, R. Molich, "Heuristic evaluation of user interfaces" Aprill 1990
- [7] J. Nielsen, "Heuristic evaluation" 1994
- [8] J. Nielsen „Does User Annoyance Matter?“ Märts 2007
<http://www.useit.com/alertbox/annoyances.html> [20.05.2011]
- [9] J. Nielsen „Website Response Times“ Juuni 2010
<http://www.useit.com/alertbox/response-times.html> [20.05.2011]
- [10] Microsoft, „Windows User Experience Interaction Guidelines“
<http://msdn.microsoft.com/en-us/library/aa511277.aspx> [20.05.2011]
- [11] W3C, „W3C Recommendations, Forms“
<http://www.w3.org/TR/html401/interact/forms.html> [20.05.2011]
- [12] J. Nielsen „Designing Web Usability“ 1999
- [13] J. Nielsen „Usability 101: Introduction to Usability“
<http://www.useit.com/alertbox/20030825.html> [20.05.2011]
- [14] OSI, „The Open Source Definition“
<http://www.opensource.org/docs/osd> [21.05.2011]
- [15] S. Krug, „Don't make me think. A common sense approach to web usability. Second edition“ 2006
- [16] B. Schneidermann ja C. Plaisant, "Designing the User Interface. Strategies for effective human-computer interaction (4th edition)" 2005

Lisad

Lisa 1

Tabel 9 Analüüsi tulemused ühes tabelis koos keskmistega

	Ülevaade süsteemi staatusest	Seos rakenduse ja reaalse maailma vahel	Kasutajaliidese järjekindlus	Pigem äratundmine kui meenutamine	Paindlikkus ja kasutamise efektiivsus	Esteetiline ja minimalistlik ülesehitus	Vigade ennetamine ja nendest taastumine	Abi ja juhendite kättesaadavus	Teiste rakendustega sarnane ülesehitus	Jälgi veebirakenduste kujundustavasid	Brauseri tööriistade funktsioneerimine	Lubada lihtsat andmete sisestamist, mis vähendaks veolukordi	Kokku
Wordpress	3	4	4	0	2	3	4	1	1	2	1	3	28
Joomla	1	3	4	3	6	3	2	3	2	2	2	2	33
Drupal	1	2	3	0	2	1	2	4	2	1	1	1	20
CMSMadeSimple	2	0	1	1	3	0	2	2	1	0	1	2	15
Keskmine	1,75	2,25	3	1	3,25	1,75	2,5	2,5	1,5	1,25	1,25	2	