

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Miha Pucelj

SISTEMI ZA UPRAVLJANJE SPLETNIH VSEBIN

DIPLOMSKO DELO NA UNIVERZITETNEM ŠTUDIJU

Ljubljana, 2009

Št. naloge: 01522/2008

Datum: 15.10.2008



Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **MIHA PUCELJ**

Naslov: **SISTEMI ZA UPRAVLJANJE SPLETNIH VSEBIN
WEB CONTENT MANAGEMEN SYSTEMS**

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

V zadnjih letih se je pojavila množica odprtokodnih pa tudi komercialnih sistemov za upravljanje spletnih vsebin, ki temeljijo na različnih programskih tehnologijah. Ravno ta pestrost pa uporabnikom pogosto otežuje izbiro, saj se v obilici možnosti in lastnosti preprosto izgubijo. Kandidat naj v svojem delu opiše osnovno arhitekturo sistemov za upravljanje spletnih vsebin in opredeli nekaj njihovih najpomembnejših lastnosti. Izmed teh lastnosti naj opredeli najvažnejše parametre, ki so pomembni pri zasnovi sistema za podporo odločanju o izbiri sistema. Na osnovi izbranih parametrov naj opiše nekaj najpopularnejših sistemov za upravljanje spletnih vsebin in na dobljenih podatkih ilustrira uporabo dobljenega odločitvenega sistema.

Mentor:

doc. dr. Matjaž Kukar



Dekan:

prof. dr. Franc Solina

IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani/-a _____,

z vpisno številko _____,

sem avtor/-ica diplomskega dela z naslovom:

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal/-a samostojno pod mentorstvom (naziv, ime in priimek)

in somentorstvom (naziv, ime in priimek)

- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki »Dela FRI«.

V Ljubljani, dne _____ Podpis avtorja/-ice: _____

Zahvala

Zahvaljujem se svojemu mentorju, doc. dr. Matjažu Kukarju, za njegovo strokovno pomoč in dobro usmerjanje pri pisanju diplomske naloge. Rad bi se zahvalil tudi staršem in ostalim bližnjim, ki so mi ves čas študija stali ob stani in me moralno in finančno podpirali.

KAZALO

POVZETEK	1
ABSTRACT	2
1 Uvod.....	3
2 Sistemi za upravljanje spletnih vsebin	5
2.1 Upravljanje z vsebinami	5
2.2 Sistemi za upravljanje vsebin na splošno	6
2.3 Celovito upravljanje vsebin	7
2.4 Sistemi za upravljanje spletnih vsebin	8
2.4.1 Odprtokodni in komercialni sistemi za upravljanje spletnih vsebin	8
2.4.2 Funkcionalnosti sistemov za upravljanje spletnih vsebin	10
3 Arhitektura CMS sistema	12
3.1 Statične spletne strani	12
3.2 Dinamične spletne strani.....	12
3.3 CMS sistem.....	13
3.3.1 Sistem za objavljanje vsebine	15
3.4 Potisni in povlečni CMS sistemi (angl. Push and Pull CMS Systems)	17
3.4.1 Potisni CMS sistem	17
3.4.2 Povlečni CMS sistem	20
3.4.3 Sklep o potisnih in povlečnih sistemih	21
4 Predstavitev CMS sistemov	23
4.1 Joomla 1.5	24
4.2 Plone 3.1.7	27
4.3 Drupal 6.9	29
4.4 Typo3 4.2.3	31
4.5 DotNetNuke 3	33
4.6 WordPress 2.7.1	36
4.7 Silverstripe 2.2.3	39
4.8 Microsoft Office SharePoint Server 2007	41
4.9 Xoops 2.3.2	44
4.10 dotCMS 1.7	46
5 Uporaba teorije odločanja pri izbiri CMS	50

5.1	Odločanje	50
5.2	Elementi odločitvenega procesa.....	50
5.2.1	Množica variant.....	50
5.2.2	Preferenčna relacija	51
5.3	Funkcija koristnosti	51
5.4	Faze odločitvenega procesa	52
5.4.1	FAZA 1: Identifikacija odločitvenega problema	52
5.4.2	FAZA 2: Identifikacija kriterijev	52
5.4.3	FAZA 3: Definicija funkcije koristnosti.....	53
5.4.4	FAZA 4: Opis variant	54
5.4.5	FAZA 5: Vrednotenje in analiza variant	54
5.5	Večparametersko odločanje	54
5.5.1	Metoda večparameterskega odločanja.....	55
5.5.2	Utežena vsota.....	56
6	Odločitveni model za izbiro CMS	57
6.1	Identifikacija odločitvenega problema	57
6.2	Identifikacija kriterijev	57
6.2.1	Osnovna zmogljivost	57
6.2.2	Razširljivost z dodatnimi funkcionalnostmi	58
6.2.3	Enostavnost za uporabo z vidika uporabnika	60
6.2.4	Enostavnost za uporabno z vidika administratorja.....	62
6.2.5	Oblikovanje izgleda.....	62
6.2.6	Interoperabilnost (angl. Interoperability)	63
6.2.7	Varnost.....	64
6.2.8	Skalabilnost.....	66
6.2.9	Tehnične značilnosti.....	67
6.2.10	Popularnost sistema	68
6.3	Definicija funkcije koristnosti.....	70
6.3.1	Osnovne funkcije koristnosti	70
6.3.2	Funkcije združevanja.....	70
6.4	Opis variant	71
6.4.1	Osnovna zmogljivost	71
6.4.2	Razširljivost z dodatnimi funkcionalnostmi	72
6.4.3	Enostavnost uporabe za uporabnika	73
6.4.4	Enostavnost uporabe za administratorja.....	73
6.4.5	Oblikovanje izgleda.....	74
6.4.6	Interoperabilnost	74

6.4.7	Varnost.....	75
6.4.8	Skalabilnost.....	75
6.4.9	Popularnost	76
6.5	Vrednotenje in analiza variant	78
6.5.1	Opis delovanja odločitvenega modela.....	78
6.5.2	Profili uporabnikov	79
6.6	Splošne ugotovitve	81
7	Zaključek.....	83
	Dodatki	85
	Viri.....	87

KAZALO SLIK

Slika 1: Primer ECM sistema [2].	9
Slika 2: Statična spletna stran [2].	12
Slika 3: Dinamična spletna stran [2].	13
Slika 4: Splošna zgradba CMS sistema [2].	15
Slika 5: Push CMS Sistem [9].	17
Slika 6: Pull CMS sistem, [9].	20
Slika 7: Osredje sistema Joomla.	24
Slika 8: Nadzorna plošča sistema Joomla.	26
Slika 9: Izgled vmesnika Plone	27
Slika 10: Prikaz "pogledov" v sistemu Plone.	28
Slika 11: Vmesnik za določanje pravic dostopa v sistemu Plone.	29
Slika 12: Osredje sistema Drupal.	30
Slika 13: Administrativno področje v sistemu Drupal.	31
Slika 14: Ozadje sistema Typo3 - dodajanje nove strani in vsebine.	32
Slika 15: Dodajanje vsebine v kolone	33
Slika 16: Razdelitev DotNetNuke strani na 5 področij.	34
Slika 17: Kontrolna plošča sistema DotNetNuke.	35
Slika 18: Določanje pravic vlog nad modulom v sistemu DotNetNuke.	36
Slika 19: Osredje sistema WordPress.	37
Slika 20: Administrativna plošča sistema WordPress.	38
Slika 21: Osredje sistema SilverStripe.	39
Slika 22: Administratorska plošča sistema SilverStripe.	40
Slika 23: Vmesnik za ustvarjanje skupin in določanje pravic.	41
Slika 24: Hierarhija SharePoint vsebnikov [28].	42
Slika 25: Osredje sistema SharePoint 2007.	43
Slika 26: Centralna administracija – izbira predloge strani za novo spletno aplikacijo.	43
Slika 27: Osredje sistema Xoops.	44
Slika 28: Osnovna razdelitev strani v Xoops-u [30].	45
Slika 29: Ozadje sistema - administracija blokov.	45
Slika 30: Osredje sistema dotCMS.	46
Slika 31: Ozadje sistema dotCMS [32]	47
Slika 32: Shema dodeljevanja pravic v sistemu dotCMS [32].	48

Slika 33: Večparameterski odločitveni model [35].	55
Slika 34: Primer definiranja funkcije koristnosti	78

KAZALO TABEL

Tabela 1: Seznam parametrov za ocenjevanje osnovne zmogljivosti.	58
Tabela 2: Seznam parametrov za ocenjevanje dodatne funkcionalnosti.	60
Tabela 3: Seznam parametrov za ocenjevanje enostavnosti za uporabo za uporabnika.	61
Tabela 4: Seznam parametrov za ocenjevanje enostavnosti uporabe za administratorja.	62
Tabela 5: Seznam parametrov za ocenjevanje oblikovanje izgleda	62
Tabela 6: Seznam parametrov za ocenjevanje interoperabilnosti.	64
Tabela 7: Seznam parametrov za ocenjevanje varnosti.	65
Tabela 8: Seznam parametrov za ocenjevanje skalabilnosti.	67
Tabela 9: Seznam tehničnih lastnosti sistemov.	68
Tabela 10: Ocene osnovne zmogljivosti.....	71
Tabela 11: Ocene dodatnih funkcionalnosti 1.....	72
Tabela 12: : Ocene dodatnih funkcionalnosti 2.....	72
Tabela 13: : Ocene enostavnosti uporabe za uporabnika.....	73
Tabela 14: Ocene enostavnosti uporabe za administratorja.....	74
Tabela 15: Ocene parametrov za oblikovanje izgleda	74
Tabela 16: Ocene parametrov za ocenjevanje interoperabilnosti	74
Tabela 17: Ocene parametrov za ocenjevanje varnosti.....	75
Tabela 18: Ocene parametrov za kriterije skalabilnosti	75
Tabela 19: Rezultati meritev zmogljivosti sistema z programom ab	76
Tabela 20: Ocene parametrov za kriterije popularnosti.....	76
Tabela 21: Podatki o prenosih sistemov	77
Tabela 22: Podatki za oceno stopnje podpore tretje strani	77
Tabela 23: Podatki o številu napisanih publikacij.....	77
Tabela 24: Podatki za oceno mesta nahajanja v rezultatu iskanja po ključni besedah.....	77
Tabela 25: Definicija profila "individualni portal"	79
Tabela 26: Definicija profila za "Portal za manjše skupine ljudi in društva"	80
Tabela 27: Definicija profila "Podjetje in večje organizacije"	80

Razlaga kratic

CMS	Content Management System
WCMS	Web Content Management System
ECM	Enterprise Resource Planning
ERP	Enterprise Resource Planning
MAUT	Multi-Attribute Utility Theory
WYSIWYG	What You See Is What You Get
HTTP	Hypertext Transfer Protocol
XHTML	Extensible Hypertext Markup Language
WebDAV	Web-based Distributed Authoring and Versioning
FTP	File Transfer Protocol
RSS	Really Simple Syndication
NIS	Network Information Service
LDAP	Lightweight Directory Access Protocol

POVZETEK

V pričujoči diplomski nalogi proučujemo arhitekturo in delovanje sistemov za upravljanje s spletnimi vsebinami (CMS) ter na podlagi uporabe teorije odločanja izdelamo odločitveni model za izbiro sistema. Na začetku predstavimo kaj sistemi za upravljanje s spletnimi vsebinami sploh so in nato predstavimo nekaj osnovnih konceptov splošnega upravljanja z vsebinami. Spoznamo se z arhitekturo CMS sistemov, delovanjem enostavnih statičnih in dinamičnih spletnih strani, ter končno še z arhitekturo in delovanjem CMS sistemov. Sledi kratka predstavitev vzorčnih CMS sistemov, ki jih v nadaljevanju diplomskega dela primerjamo po številnih kriterijih in vrednotimo s pomočjo teorije odločanja. Na podlagi opisane teorije prikažemo izgradnjo odločitvenega modela za izbiro CMS sistema. Zaradi enostavnosti in najširše dostopnosti za izgradnjo modela uporabljamo program Excel. Na koncu je definiranih nekaj različnih uporabniških profilov, na podlagi katerih prikažemo delovanje odločitvenega modela.

KLJUČNE BESEDE: sistemi za upravljanje spletnih vsebin, CMS, WCMS, teorija odločanja, odločitveni model, arhitektura CMS sistemov

ABSTRACT

In this thesis the architecture and principles of Content Management Systems (CMS) are examined. Based on the decision theory a decision model for choosing the appropriate CM system was designed. At the beginning the concept of content management systems is explained and some basics of content management introduced. We also examine the CMS architecture, static and dynamic web pages design and, finally, the infrastructure of the CM systems. This is followed by a short introduction of sample CM systems, which are later compared according to numerous criteria and evaluated using the decision theory. On the basis of the described theory the development of a decision model for choosing the CM system is demonstrated. Because of its simplicity and wide accessibility, the Excel program was used to build the decision model. In the final part of the thesis a few different user profiles are defined, which are used to show the usage of the decision model.

KEYWORDS: content management system, CMS, WCMS, decision theory, decision model, CMS architecture

1 Uvod

Na tržišču se je v zadnjih letih pojavilo veliko različnih sistemov za upravljanje spletnih vsebin. Kot alternativa dragim komercialnim sistemom se ponuja množica odprtokodnih sistemov. Predvsem za slednje velja, da temeljijo na najrazličnejših tehnologijah. Uporabljajo različne programske jezike, spletne in aplikacijske strežnike, ter podatkovne baze.

Raznovrstna pa ni samo tehnologija ampak tudi množica funkcionalnosti, ki jih posamezni sistemi nudijo uporabnikom. Vsak sistem ima svoj nabor funkcionalnosti, s katerim opredeljuje svoj namen oziroma ciljno skupino uporabnikov. Nekateri sistemi so že v osnovi bolj primerni za uporabo v večjih podjetjih in organizacijah, drugi pa so primernejši za manjše skupine ljudi ali pa za individualno uporabo. Kljub temu pa uporabnik pri določanju, kateri sistem naj izbere, naleti na težave, saj se vrste funkcionalnosti med sistemi na različne načine prekrivajo ali izključujejo. Odločitve otežuje še zmožnosti za razširitve sistemov, ki lahko sisteme prilagodijo potrebam skoraj vsake od uporabniških skupin. Izdelovalci in razvijalci vsak na svoj način prikazujejo in poudarjajo osnovne koncepte svojih sistemov, kar za bodočega uporabnika pomeni veliko dela, da iz raznih virov in dokumentacij potencialnih sistemov izlušči enakovrstne informacije. Na spletu je sicer mogoče najti veliko primerjav med sistemi, vendar je v večini skupina sistemov za primerjavo ponavadi omejena po neki lastnosti: primerjajo samo sisteme napisane v istem programskem jeziku, pogosto primerjajo le sisteme pod odprtokodno licenco, včasih je število primerjanih sistemov zelo majhno, ocene variant so le opisne, to pa uporabniku še vedno ne pove jasno, kateri sistem je zanj najprimernejši. Obstajajo tudi razni sezname, ki za veliko sistemov prikazujejo katere funkcionalnosti le-ti vsebujejo in katerih ne, vendar pa bi morale biti nekoliko bolj natančno opisane. Različni proizvajalci se namreč istih konceptov lotevajo na različne načine, posledica pa je različna kakovost in funkcionalnost rešitev. Izbiro sistema pa bil zelo olajšala tudi možnost določanja pomembnosti posameznih funkcionalnosti.

Zato bom v diplomskem delu poskušal opredeliti nekaj najpomembnejših lastnosti sistemov za upravljanje spletnih vsebin in jim določiti parametre, ki te lastnosti definirajo. Na podlagi teh lastnosti bom zasnoval odločitveni model, ki bo uporabniku pomagal pri izbiri sistema glede na njegove želje in potrebe. Izbral bom deset vzorčnih sistemov, o katerih se danes pogosto govori, jih na osnovi določenih parametrov skušal opisati ter preizkusiti delovanje zasnovanega odločitvenega sistema za nekaj tipičnih profilov uporabnikov.

Diplomsko delo je razdeljeno na 8 poglavij. V naslednjem bom najprej opredelil kaj sistemi za upravljanje s spletnimi vsebinami sploh so in katere so njihove tipične lastnosti in funkcionalnosti. Predstavil bom tudi nekaj osnovnih konceptov splošnega upravljanja vsebin. V poglavju »Arhitektura CMS sistema« se bom poglobil v splošno zgradbo sistemov in predstavil nekatere tipične koncepte delovanja. Opisal bom delovanje statičnih in dinamičnih spletnih stani ter predstavil, kako so ti koncepti uporabljeni pri delovanju CMS sistemov. V naslednjem poglavju bom govoril o okolju v katerem sem posamezne sisteme preizkušal ter podal opise za vsakega od sistemov. V poglavju »Teorija odločanja« bom predstavil nekatere osnovne koncepte iz teorije odločanja, ki jih bom uporabil pri gradnji odločitvenega modela. Poglavje z naslovom »Odločitveni model« po fazah odločitvenega procesa predstavlja izgradnjo odločitvenega modela. Zajema identifikacijo odločitvenega problema in kriterijev, opis vseh variant ter vrednotenje variant po nekaj profilih uporabnikov. V zadnjem poglavju bom podal ugotovitve, kateri od izbranih sistemov so na podlagi preizkušanja odločitvenega modela najprimernejši za določene profile uporabnikov. Komentiral bom pravilnost delovanja odločitvenega modela in predstavil smernice nadaljnjega razvoja tega odločitvenega modela.

2 Sistemi za upravljanje spletnih vsebin

Kaj sploh so WCMS? Danes so zelo poznani tudi kot »CMS sistemi« oziroma natančnejše »WCMS sistemi«. Kartica CMS izhaja iz angleških besed »Content Management System«, »W« pa izhaja iz angleške besede »Web« in tako označuje spletno naravnost sistema. Dobesedni prevod izraza CMS, to je sistem za upravljanje vsebin, nam pove, da tukaj ne gre samo za upravljanje spletnih, ampak tudi drugih vrst vsebin.

V nadaljevanju bom opisal kako v splošnem poteka upravljanje vsebin, navedel bom nekatere osnovne pojme, principe, tehnologije in sisteme, ki se pojavljajo v zvezi z upravljanjem vsebin. Nato bom skušal umestiti sistem za urejanje spletne vsebine v širši okvir celovitega upravljanja vsebin.

2.1 Upravljanje z vsebinami

Upravljanje z vsebinami je skupek tehnologij in procesov, ki podpirajo življenjski cikel digitalnih vsebin. Pod digitalne vsebine štejemo besedila, multimedijske ter ostale vsebine, ki jih lahko zapišemo v datotečni obliki. Načini in nameni upravljanja vsebin se od področja do področja razlikujejo. Življenjski cikel digitalne vsebina tipično šest razvojnih stopenj[1]:

- nastanek,
- posodobitev,
- objava,
- prevod,
- arhiviranje in
- umik iz objave.

Naj na kratko opišem naštetih stopenj. Neko vsebino ustvari en ali več avtorjev. Ko vsebino imamo, jo lahko objavimo na spletu, shranimo na lokalno omrežje ali pošljemo po elektronski pošti. Čez čas se lahko pokaže potreba po posodobitvi vsebine. Če imamo opravka z večjezičnim okoljem, je potrebno le-to tudi prevesti. Objave moramo pogosto tudi arhivirati, zato mora biti, da zagotovimo obstojnost, shranjena na ustreznem mediju. Zadnja stopnja pa je umik iz objave.

Upravljanje vsebin ni samo izbira in implementacija tehnologije, ampak gre tudi za identifikacijo informacijskih potreb, ovrednotenje informacij in določitev postopka kako le te

na podlagi tega kar najboljše unovčiti. Potrebno zagotoviti nadzor nad vsebino od nastanka do uporabe, to pa pomeni tudi spreminjanje načina dela ljudi[2].

Z življenjskim ciklom vsebine povezujemo tudi nekatere vloge uporabnikov in njihove odgovornosti pri tem. Naj omenim nekaj najbolj pogostih:

- Avtor (angl. Author) vsebine je zadolžen za ustvarjanje in urejanje vsebine.
- Urednik (angl. Editor) je zadolžen za usklajevanje sporočila vsebine, prevod.
- Odgovorni urednik (angl. Publisher) odobri/objavi vsebino.
- Upravljaavec (angl. administrator) je odgovoren za upravljanje pravic dostopa do vsebin, prav tako nudi pomoč uporabnikom.
- Gledalec, uporabnik, gost (angl. Viewer, Consumer, Guest) je oseba, ki bere ali uporablja objavljeno vsebino
- Druge vloge (angl. Other roles) lahko določimo glede na potrebe.

2.2 Sistemi za upravljanje vsebin na splošno

Sistemi za upravljanje vsebin (angl. Content Management System, v nadaljevanju CMS) so v splošnem programske rešitve, ki na uporabniku prijazen način omogočajo ustvarjanje, upravljanje, urejanje, objavljanje in preiskovanje različnih vsebin, kot so na primer novice, članki, dokumenti, vodiči, dokumentacije, brošure. Poleg upravljanja elektronskih dokumentov pa pod zmožnosti upravljanja spadajo tudi razne datoteke, slike, zvočni in video zapisi. Vsebina je urejeno shranjena, poleg tega pa sistemi nudijo tudi storitve kot so časovno odvisno objavljanje, spremljanje delovnih tokov dokumentov (angl. Workflow), nadzor različic (angl. Versioning), kontrolo dostopa (angl. Access Control) preko vlog in z njimi določenimi pravicami. CMS sistemi tako skrbijo za celoten življenjski cikel vsebine od nastanka do arhiviranja.

Na tem področju se pojavljajo se različne oblike in poimenovanja sistemov kot so na primer upravljanje spletnih vsebin(angl. Web Content Management), upravljanje digitalnih sredstev (angl. Digital Asset Management), upravljanje digitalnih zapisov (angl. Digital Records Management), upravljanje elektronske vsebine (angl. Electronic Content Management). Vsak od naštetih sistemov ima glede na zahteve nekoliko drugačen koncept, vendar je vsem skupno upravljanje z vsebinami in njihova distribucija uporabnikom.

Najširši okvir upravljanja vsebin zajemajo sistemi za celovito upravljanje vsebin, ki ga bom opisal v naslednjem razdelku.

2.3 Celovito upravljanje vsebin

Celovito upravljanje vsebin (angl. Enterprise Content Management, v nadaljevanju ECM) je skupek strategij, metod in tehnologij, ki se jih uporablja za zajem, upravljanje, shranjevanje, ohranjanje in prezentacijo vsebine in dokumentov povezanih z organizacijskimi procesi. ECM orodja in strategije tako omogočajo upravljanje z neorganiziranimi informacijami. Pri ECM rešitvah ne govorimo samo o enem zaprtem programskem produktu, ampak o kombinaciji več različnih tehnologij in sistemov oziroma komponent, ki so potrebne in prilagojene potrebam posameznih organizacijskih in poslovnih procesov. Rešitev celovitega upravljanja vsebine se od sistemov za upravljanje spletnih vsebin razlikuje predvsem v treh glavnih idejah [3].

1. ECM se pojavlja lahko kot integracijski vmesni sloj, ki presega vertikalno aplikacijsko arhitekturo. Uporabnik se niti ne zaveda, da uporablja ECM. Ta sloj zagotavlja infrastrukturo za vzpostavitev nove tri-nivojske arhitekture, ki je spletno usmerjena. Veliko vlogo pri implementaciji ECM imajo pri tem EAI (Enterprise Application Integration) in SOA (service oriented architecture).
2. ECM se pojavlja lahko tudi kot neodvisne storitve, ki zagotavljajo uporabo informacij neodvisno od vira in načina uporabe. Koncept storitve omogoča da različne aplikacije lahko uporabljajo funkcionalnosti na enem mestu, kar preprečuje redundanco in olajšuje vzdrževanje. Tukaj ima pomembno vlogo standardizacija vmesnikov aplikacij, ki storitve uporabljajo.
3. Sistem ECM je lahko tudi podatkovno skladišče, v katerega podatke in dokumente shranjujejo vse aplikacije. Le-te imajo tako na enem mestu na voljo vse podatke, to pa zmanjšuje možnost redundance in nekonsistentnosti informacij. Tukaj ima pomembno vlogo integracija vsebine.

Takšna rešitev deluje pravilno, če je uporabniku nevidna. Pri tem uporabljene tehnologije so infrastruktura, ki podpira specializirane aplikacije in storitve potrebne v poslovnem in organizacijskem procesu. ECM je tako zbirka komponent, ki sestavljajo več nivojski model, ki povezuje vse tehnologije potrebne za obdelavo, upravljanje in prikazovanje strukturirane in

ne-strukturirane vsebine. Rešitev ECM zna tako razdelati in upravljati z vsemi informacijami, ki jih na primer na višjem nivoju obdeluje sistem za upravljanje spletnih vsebin.

Slika 1 prikazuje primer ECM sistema, ki poleg upravljanja spletnih vsebin obsega še dva druga kanala za objavljanje vsebine: sistem za tiskanje, sistem za elektronsko pošto in sistem za ponujanje vsebin v standardiziranem formatu [2].

2.4 Sistemi za upravljanje spletnih vsebin

WCMS sistem je v sistem za upravljanje vsebin, ki je implementiran kot spletna aplikacija in ponuja možnost upravljanja spletnih vsebin. Uporabnikom nudijo avtorska orodja, s pomočjo katerih lahko ustvarjajo in urejajo vsebino brez znanja o programskih in označevalnih jezikov (angl. Markup Languages). Vse delo z sistemom poteka preko spletnega vmesnika.

Prezentacijski sloj portala je ločen od vsebine in je določen s t.i. predlogami (angl. Template). Omenjena lastnost tako omogoča enostavno spreminjanje zunanjšega izgleda, ne da bi zaradi tega morali posegati v shranjeno vsebino.

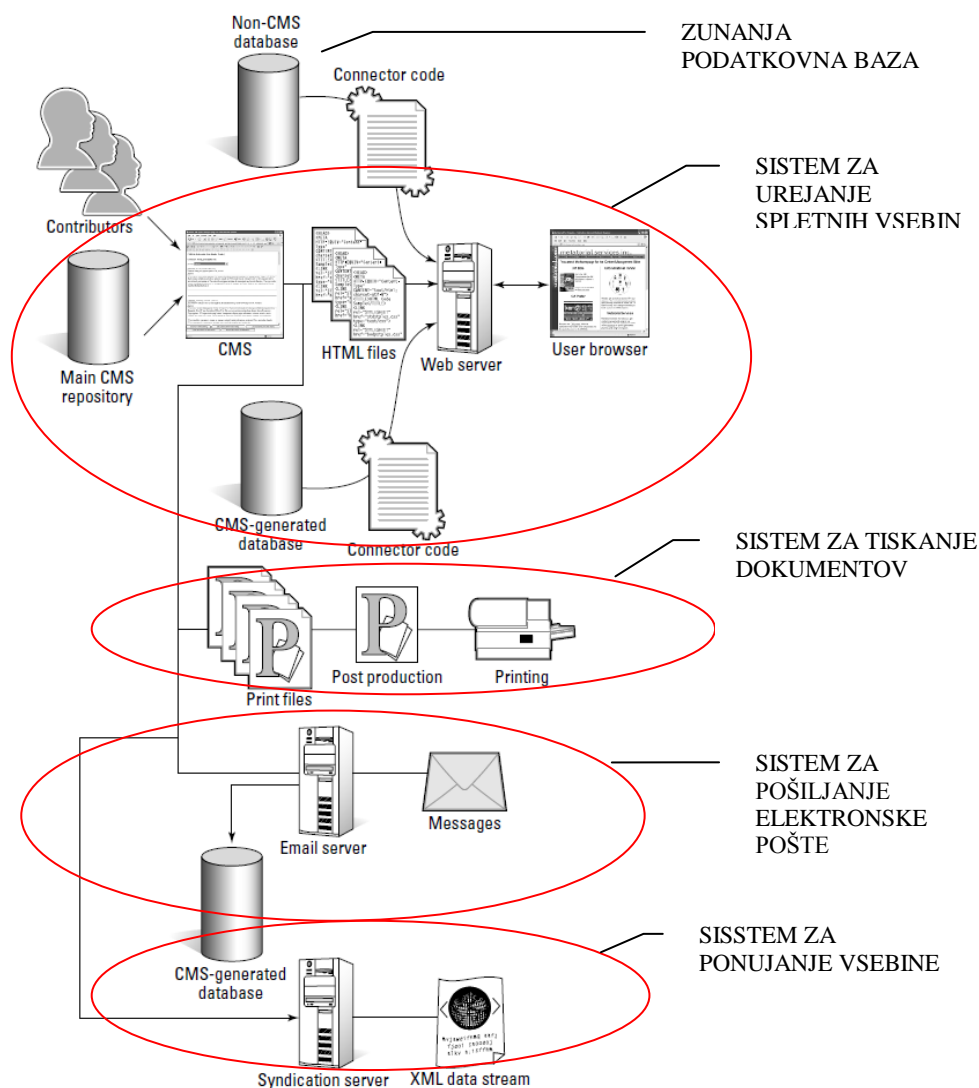
Večina sistemov uporablja za shranjevanje vsebine podatkovno bazo v tekstovni obliki, nekateri pa shranjujejo vsebino v datoteke na disk.

Za sisteme velja, da je vzdrževanje primerno tudi za neizkušene oziroma ne-tehnične uporabnike, vendar sama postavitve sistema in njegova osnovna prilagoditev pa zahteva računalniško nekoliko bolj izobražene uporabnike.

Velikokrat se namesto WCMS uporabljal samo CMS. Zato se bom od tukaj naprej z izrazom CMS omejil le na sisteme za upravljanje spletnih vsebin.

2.4.1 Odprtokodni in komercialni sistemi za upravljanje spletnih vsebin

Na trgu se pojavlja veliko število CMS sistemov, ki so na voljo pod odprtokodno ali komercialno licenco. Prvi so še posebej priljubljeni, saj so brezplačni in imajo zato široko bazo uporabnikov in s tem zagotovljeno podporo.



Slika 1: Primer ECM sistema [2].

Odprtokodni CMS sistemi

Odprtokodni sistemi ponujajo veliko funkcionalnosti in dobro prilagodljivost brezplačno. To je velika prednost pred komercialnimi, katerih cena se lahko giblje med nekaj sto in več tisoč evri. Vendar potrebno tukaj poudariti, da so tudi odprtokodne rešitve namenjene pridobivanju zaslužka. Uporabnikom so na voljo sicer brezplačno, vendar se moramo zavedati, da mora kljub temu nekdo podpirati razvoj sistema. Ponudniki tako nudijo plačljive storitve kot so podpora uporabnikom, implementacija sistema, prilagoditve sistema in razvoj dodatnih funkcionalnosti, če seveda uporabnik to želi in potrebuje.

Za odprtokodne sisteme velja, da načeloma nimajo prave tehnične podpore, kot jo nudijo ponudniki komercialnih sistemov. Pri odprtokodnih sistemih je pomemben dejavnik skupnost, ki nek odprtokodni sistem razvija in zanj tudi skrbi. Tako je podpora je na voljo v obliki forumov, blogov in spletnih portalov, ki jih vzdržujejo člani takšnih skupnosti. Koncept odprte kode seveda omogoča, da uporabniki, če imajo seveda dovolj znanja, sami spreminjajo programsko kodo in tako prilagajajo sistem svojim potrebam [4].

Pojavlja se veliko argumentov proti implementaciji odprtokodnih sistemov, vsi pa nekako ciljajo na pojem negotovost, ki se jo pripisuje odprtokodnim rešitvam zaradi »prostovoljne« baze razvijalcev in pomanjkanja zanesljive in hitre tehnične podpore [5].

Nekateri primeri odprtokodnih sistemov, ki so trenutno na trgu najbolj prepoznavni [6],[7]:

Drupal, Plone, Joomla, Typo3, WordPress, Xoops, Silverstripe, DotNetNuke, dotCMS, ezPublish, CMSMadeSimple, phpNuke, MediaWiki, TYPOLight, e107, Alfresco, Xaraya.

Komercialni CMS sistemi

Kako dobro se bo sistem obnesel na trgu je v veliki meri odvisno od kvalitetne podpore in možnosti izobraževanja uporabnikov. Slednje je za odprtokodne rešitve skoraj nemogoče najti.

Komercialni sistemi so plačljivi. Proizvajalci teh sistemov za svoje delo enostavno pričakujejo neko nadomestilo. Cene sistemov so lahko visoke, vendar ponudniki zagotavljajo uporabnikom hitro in kvalitetno tehnično podporo, izobraževanja. Komercialni sistem lahko že vsebuje vse potrebne, mogoče tudi bolj specifične funkcionalnosti, tako da je posledično implementacija hitrejša kot pri odprtokodnih sistemih.

Uporabniki imajo zaradi prej omenjenih lastnosti do komercialnih sistemov večje zaupanje, zato so tovrstni sistemi bolj konkurenčni na poslovnem področju. [4], [5].

Naj omenim nekaj komercialnih produktov: Microsoft SharePoint Server 2007, Contegro, CMSDragon, Jalios JCMS, Elite CMS, Movable Type.

2.4.2 Funkcionalnosti sistemov za upravljanje spletnih vsebin

Obstaja veliko CMS sistemov, ki se med seboj bolj ali manj razlikujejo, vendar pa jih združujejo naslednje funkcionalnosti:

- **Uporaba predlog za oblikovanje izgleda**

Izgled shranjene in na novo ustvarjene vsebine je določen s predlogami. Tako lahko celoten zunanji videz mogoče urejati na enem mestu.

- **Enostavno urejanje vsebine**

Sistemi vključujejo t.i. orodje »WYSIWYG« (iz angl. What You See Is What You Get), ki uporabniku omogoča enostavno kreiranje in urejanje vsebine.

- **Razširljivost množice funkcionalnosti**

Za vsak sistem obstajajo številni dodatki, kot so vtičniki in moduli, ki jih je mogoče enostavno namestiti v sistem in s tem razširiti funkcionalnost

- **Popravki in nadgradnje**

Razvijalci sistemov skrbijo za redne nadgradnje, ki povečujejo zmogljivosti in sistemu zagotavljajo skladnost s trenutno veljavnimi spletnimi standardi.

- **Delegacija (Podpora upravljanju kontrole dostopa)**

Večina CMS-jev omogoča upravljanje z uporabniki, ki jih je mogoče razporejati v skupine in jim z dodeljevanjem vlog določati pravice in odgovornosti za posamezna področja vsebine. Tako preprečujejo dostop uporabnikom do vsebine za katero niso avtorizirani.

- **Upravljanje dokumentov**

CMS sistem naj bi zagotavljal sredstva za spremljanje celotnega življenjskega cikla dokumentov od nastanka, revidiranja, arhiviranja in izbrisa.

- **Virtualizacija vsebine**

CMS sistem naj bi zagotavljal, da uporabnik dela z navidezno kopijo celotne spletne, množico dokumentov ali nekega dela sistema. To dovoljuje spreminjanje medsebojno odvisnih virov in njihov predogled.

- **Časovno planiranje objav**

Vsebini se lahko vnaprej določa kdaj naj se objavi in kdaj naj se objava zaključi.

- **Decentralizirano vzdrževanje**

Administracija CMS sistema je možna preko spletnega brskalnika, zato ni omejitev glede lokacije in časa.

- **Samodejno generiranje navigacije**

Meniji se generirajo samodejno na podlagi shranjenih podatkov.

- **Podpora spremljanju delovnega toka**

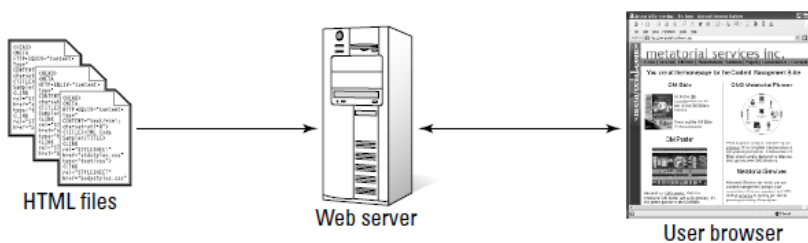
Delovni tok predstavlja posamezne zaporedne ali vzporedne ciklične naloge, ki jih je potrebno izpolniti med življenjskim ciklom neke objave: na primer avtor lahko napiše članek in ga shrani, vendar ga ne more objaviti dokler ga urednik ne pregleda in odgovorni urednik ne da dovoljenja za objavo.

3 Arhitektura CMS sistema

3.1 Statične spletne strani

Statična spletna stran sestoji iz HTML strani in z njimi povezanih datotečnih virov. Vse je shranjeno na aplikacijskem strežniku v obliki datotek. Statične strani so predhodno zgrajene (angl. pre-built), kar pomeni da se ne generirajo na zahtevo in se prikazujejo samo v obliki, ki je določena v skripti strani. Urejanje pomeni zamenjavo stare HTML strani z novo.

Statične spletne strani so preproste in namenjene uporabi, kjer ni veliko spreminjanja in ne zahtevajo prilagajanja/personifikacij. Prednost statičnih strani je hitrost, slabost pa zmožnost prilagodljivosti in razširljivosti.

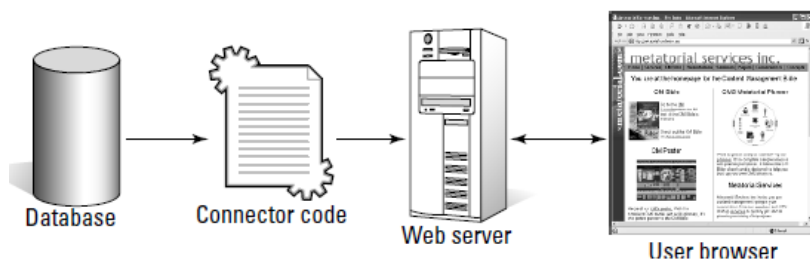


Slika 2: Statična spletna stran [2].

3.2 Dinamične spletne strani

Dinamična spletna stran se generira na zahtevo. Ob kliku uporabnika na neko povezavo podatkovna baza na strežniku prejme zahtevo za poizvedbo. Aktivira se predloga strani (angl. template page). Ta vsebuje HTML in druge programske skripte, objekte in programe, ki znajo interpretirati dano zahtevo, se povezati na podatkovni vir, pridobiti ustrezno vsebino in sprocesirati vse potrebno za izgradnjo HTML strani. Ko je predloga HTML stran ustvarila, jo spletni strežnik pošlje odjemalcu, to je uporabnikovemu brskalniku.

V popolnoma dinamičnih spletnih straneh HTML strani sploh ne obstajajo, ampak obstajajo le mehanizmi, ki jih, kadar je to potrebno, generirajo.



Slika 3: Dinamična spletna stran [2].

Kar velja za dinamično spletno stran velja tudi za CMS sistem. V čem je torej razlika? Razlika je v tem, da dinamična spletna stran nima nobene funkcionalnosti upravljanja vsebine, ampak samo pridobivanje in prikazovanje informacij iz obstoječih virov. Poleg tega pa zna CMS sistem generirati tudi statične strani.

3.3 CMS sistem

Večinoma spletnih strani je kombinacija statičnih in dinamičnih. To pomeni da so določeni deli strani sestavljeni iz HTML datotek, ostali pa se dinamično generirajo iz podatkovne baze. Seveda pa lahko različni deli strani uporabljajo tudi različne podatkovne vire. To velja tudi za CMS sisteme.

Slika 4 prikazuje v groben splošno zgradbo CMS sistema, ki ima lahko vse ali pa nekaj od naslednjih komponent:

- **CMS aplikacija**

CMS aplikacija varno stoji za spletnim strežnikom in skrbi za zbiranje in upravljanje vsebine, njenega delovnega toka in za ostalo administracijo. Sama arhitektura se med posameznimi produkti lahko razlikuje. Pri nekaterih izvedbah je ta aplikacija del lokalnega omrežja organizacije za požarnim zidom, kjer skrbi za zbiranje vsebine. Pri nekaterih pa se nahaja zunaj lokalnega omrežja na aplikacijskem strežniku, kjer skrbi za dinamično generiranje spletne strani. Tu govorimo o potisnih (angl. Push) in povlečnih (angl. Pull) CMS sistemih.

- **Repozitorij**

Za aplikacijskim strežnikom stoji podatkovni vir, ki je lahko v obliki relacijske podatkovne baza ali strukturiranem XML formatu. Repozitorij hrani vso vsebino, administrativne podatke in vse ostalo, ker je potrebno za izgradnjo spletne strani. Sem spadajo na primer tudi razne grafične in slogovne datoteke.

- **Množica HTML datotek**

CMS sistem upravlja z vsebino in jo postavlja(angl. Deploy) kot statični del spletne strani.

- **Podatkovni vir CMS sistema (angl. CMS generated DB, Life Data Source)**

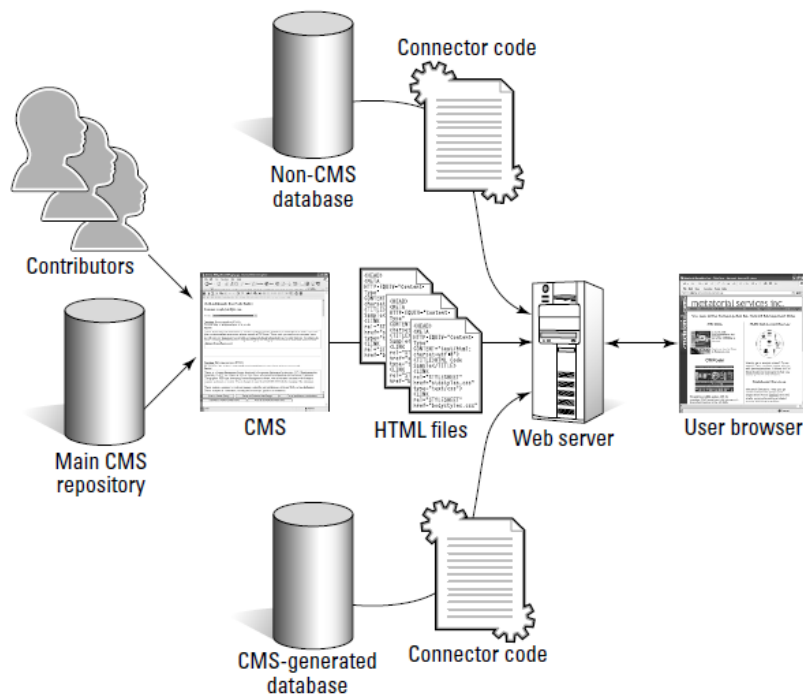
Podatkovni vir CMS sistema je lociran na spletnem strežniku, vsebuje dejansko vsebino in se tako uporablja za generiranje dinamičnih delov spletne strani. CMS sistem lahko podatke iz svojega repozitorija namesti oziroma posreduje podatkovnemu viru CMS sistema. Zaradi takega načina delovanja, se lahko tudi dinamično vsebino upravlja v okolju za požarnim zidom ne da je vsebina objavljena na strežniku in tako dostopna javnosti

- **Drugi podatkovni viri**

Spletni strani je mogoče dodajati tudi druge podatkovne vire, ki pa se jih ne povezuje z samo CMS aplikacijo ampak jo dodamo samo spletni strani. Kot primer naj navedem vključitev transakcijske podatkovne baze za spremljanje prodaje nekega podjetja. Dodan vir teče popolnoma neodvisno od CMS sistema ali predlog strani, informacije pa so dostopne na spletni strani

- **Sistem za objavljanje (angl. Publishing System)**

Množica predlog za objavo vsebine dostopa do podatkov v virih in jih pripravi v taki obliki, kot jih spletna stran zahteva. V nekaterih izvedbah morajo biti podatki že takoj posredovani v končni statični HTML obliki, v drugih pa so samo posredovani podatkovni bazi, ki teče na spletnem strežniku. Podrobnejše bom sistem predlog predstavil v razdelku 3.3.1.



Slika 4: Splošna zgradba CMS sistema [2].

3.3.1 Sistem za objavljanje vsebine

Zato, da je vsebina objavljena na spletni strani, poskrbi sistem za objavljanje (angl. Publishing System).

3.3.1.1 Zgradba sistema za objavljanje vsebin

Sistem za objavljanje sestavljajo:

- **Predloge (angl. Templates)** so glavni element sistema za objavljanje. Predloge premoščajo vrzel med surovimi podatki shranjenimi v repozitoriju in specifičnimi potrebami objavljanja vsebine. Za razliko od predlog za oblikovanje besedila, je ta predloga v bistvu program, napisan v nekem programskem jeziku, ki določa logiko za izgradnjo objave. Predloge vsebujejo več komponent:
 - **statični elementi**, kot so tekst, večpredstavnostne vsebine in skripte, ki se objavljajo brez posebnega procesiranja
 - **klici storitev za objavljanje** (angl. Calls to Publication Services), ki skrbijo za pridobivanje in oblikovanje komponent in meta-podatkov iz repozitorija ter

izvajajo funkcije kot so pretvarjanje vsebine, izvajanje personalizacijskih pravil in gradnja navigacijskih elementov.

- **Kkici storitev izven CMS** (angl. Calls To Services Outside CMS), ki omogočajo širšo integracijo z zunanjimi spletnimi storitvami in podatki.
- **Storitve za objavljanje** (angl. Publishing Services) predstavljajo aplikacijsko logiko in storitev CMS sistema, ki zagotavljajo generiranje objav iz repozitorija. Storitve za objavljanje
 - nalagajo in izvajajo (prej omenjene predloge)
 - zagotavljajo specifične storitve za izvoz podatkov v druge formate (na primer PDF)
 - zagotavlja most do storitev zunaj CMS sistema, ki jih potrebujemo za vključitev zunanjih podatkov.
- **Povezave** do sistemov, ki niso del CMS sistema samega ampak so lahko na primer del sistema za celovito upravljanje vsebin (npr.: povezava do ERP ¹sistema)

3.3.1.2 *Objavljanje vsebine v sistemu za upravljanje spletnih vsebin*

Glavna aktivnost v CMS sistemu je seveda objava vsebine na intranetu ali internetu.

V primeru dinamičnih strani se naenkrat generira samo ena stran. Uporabnikov klik sproži zahtevo za prikaz strani (angl. Page Request) spletnemu strežniku, ta pa le to posreduje naprej in sproži storitev za objavo (angl. Publishing Service), ki izvede naslednji postopek:

1. Naloži predlogo
2. Predlogi poda morebitne parametre, vsebovani v zahtevi za prikaz strani
3. Izvede programsko kodo predloge
4. Izgrajeno spletno stran pošlje spletnemu strežniku za prikaz v spletnem brskalniku

V primeru statičnih strani CMS sistem vse strani generira naenkrat in jih potem nudi kot HTML strani. Administrator sistema statične strani ustvari preko uporabniškega vmesnika. CMS sistem potem v ozadju sam pokliče ustrezne storitve in predloge, ki to stran izgradijo.

¹ ERP (angl. Enterprise Resource Planning)

Kot sem že omenil, spletno stran ustvari predloga iz repozitorija CMS sistema. Poleg tega pa lahko predloga za generiranje strani uporabi tudi nekatere druge vire in zmogljivosti, ki pa niso del storitev CMS sistema.

To je še posebej to razvidno pri dinamičnih straneh, pri katerih storitve za objavlanje vključujejo tako spletni kot tudi aplikacijski strežnik.

Spletni strežnik skrbi za sprejemanje zahtev ter potem prikazovanje rezultata, aplikacijski strežnik pa je program, ki zagotavlja predpomnilnik, povezave do podatkovne baze in druge storitve za zagotavljanje delovanja CMS in povečanje njegovih zmogljivosti.

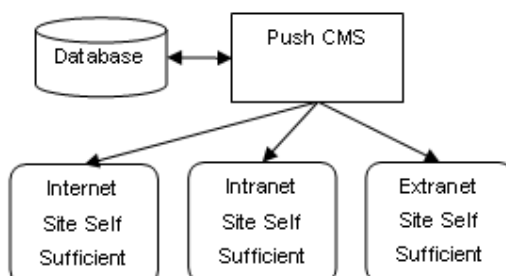
[8]

3.4 Potisni in povlečni CMS sistemi (angl. Push and Pull CMS Systems)

Iz tehničnega in arhitekturnega vidika gledano, poznamo dve vrsti CMS sistemov, in sicer »Push» in »Pull» sisteme. Besedi izhajata iz angleščine, ki pomenita potisni in povleči. S tema dvema besedama je nekako s tehničnega vidika opisan način objavlanja vsebine pri posamezni vrsti sistema.

3.4.1 Potisni CMS sistem

Značilnost potisnega sistema je ta, da za predpomnenje in dostavo vsebine uporablja nepovezane datoteke (angl. flat file). Te datoteke so potem neodvisne od podatkovnega vira. V potisnem sistemu je vsebina iz repozitorija združena v nek standarden format, ki je potem preko odprtih standardov »potisnjena« (iz angl. besede push) do ustreznega spletnega strežnika [9].



Slika 5: Push CMS Sistem [9].

Slika 5 prikazuje primer arhitekture »push« sistema. Vidimo, da se spletna stran nahaja ločeno od samega sistema CMS, ki se skupaj z podatkovno bazo nahaja v nekem lokalnem omrežju.

Primeri sistemov, ki delujejo na ta način ali pa ga vsaj v neki meri podpirajo: MovableType, dotCMS, Alfresco, dotNetNuke, IBD MicroCMS, Magnolia.

3.4.1.1 Prednosti potisnega sistema

- **Iskalnim robotom prijazni URL naslovi**

Ko je neka vsebina v obliki nepovezane datoteke »potisnjena« na spletni strežnik, jo iskalni roboti lažje najdejo, saj so le ti prirejeni za indeksiranje spletnih strani, zgrajenih iz datotek. Naslov take strani je oblike `http://www.domena.com/podjetje/domov.php` , naslovi povlečnega sistema pa so oblike `http://www.domena.com/index.php?ID=123`. Težava je v tem, da roboti težje najdejo povezave, ki vsebujejo spremenljivke, nekateri pri indeksiranju celo ne upoštevajo dela naslova, ki je za »?« . Tako lahko del spletne strani izpade iz indeksiranja. Ključne besede v prijaznih URL naslovih prispevajo k boljšemu uvrščanju med iskalnimi rezultati [9].

- **Enostavnejše objavljanje na več strežnikih**

Objavljanje vsebine na več strežnikih je s push sistemi enostavnejše. Če želimo objavljati še na drugem strežniku, mora administrator centralnemu repozitoriju le dodati informacijo o dostopu do novega strežnika. Vsebina se nato preko odprtih standardov (npr.: FTP, SFTP, VPN, datotečni sistem) prenaša še na druge lokacije.

Pri Pull sistemih pa mora biti na vsakem dodatnem strežniku, ki bo ponujal vsebino, posebej nameščen CMS sistem za dostavo. Pri tem, pa je vedno znova potrebno ustrezno nastaviti tudi sam spletni strežnik, da CMS sistem za dostavo sploh deloval [9].

- **Neodvisnost vsebine spletne strani od CMS sistema**

Ko potisni sistem spletnemu strežniku dostavi vsebino, je ta v popolnoma neodvisnem in samostojnem formatu, to pa omogoča [9]:

- administratorju, da te datoteke premika ali arhivira brez skrbi za to, da bi vsebovane informacije postale nedostopne. Pri povlečnih sistemih pa je

dostopnost informacij vseskozi odvisna od delovanja sistema, saj če na primer baza v nekem trenutku ne deluje, vsebina ni dostopna.

- enostaven prenos vsebine v druge aplikacije ali nove sisteme, saj je shranjena v standardnih odprtih formatih. S povlečnimi sistemi težje omogočimo dostop do vsebine drugim sistemom, saj je vsebina lastniško zaščitena s podatkovno bazo.
- enostavnejšo vzdrževanje »push« sistema, saj je morebitne popravke in nadgradnje sistem potrebno namestiti le na enem mestu, medtem ko je pri »pull« sistemih to potrebno narediti na vseh strežnikih.

- **Krajši čas nalaganja**

Spletni strežniki so optimizirani za delo z datotečnimi spletnimi stranmi. Vsi mehanizmi predpomnenja in postopki za izboljšanje zmogljivosti so prilagojeni konceptu datotek. Ker potisni sistem dostavlja vsebino, prilagojeno spletnemu strežniku, je ta zato sposoben istočasno podpirati več uporabnikov kot povlečni CMS sistem, ki mora za strežbo strani skrbeti sam [9].

- **Integracija infrastrukture**

Potisni sistem se nahaja znotraj lokalnega omrežja, za požarnim zidom, zato je mogoča lažja in varnejša integracija tega sistema z ostalimi komponentami kot je lahko na primer aktivni imenik ali podatkovni strežnik, saj je mogoč neposreden dostop. Povlečni sistem, ki je nameščen na zunanjem strežniku in je tako lahko zunaj požarnega zidu nekega lokalnega omrežja, pa potrebuje uporabo aktivnega imenika poseben dostop, prav tako pa potrebuje poseben dostop tudi svojega lastnega podatkovnega strežnika [9].

3.4.1.2 Slabosti potisnega sistema

- **Neažurnost vsebine**

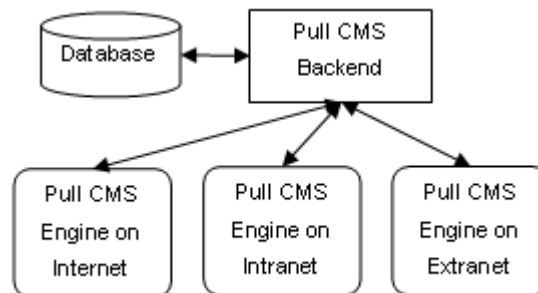
Kadar pride do spremembe vsebine v CMS sistem, le ta na spletni strani ni vidna dokler se nova vsebina ponovno ne objavi v statični obliki na spletnem strežniku.

- **Neprimeren za hitro spreminjajoče se vsebine**

V primeru sistema, kjer se vsebina hitro spreminja, mora potisni sistem ustvariti veliko različnih statičnih strani in jih sproti pošiljati zunanjim strežnikom. To pa lahko povzroči kar precejšnjo preobremenitev sistema.

3.4.2 Povlečni CMS sistem

Pri povlečnih sistemih je na spletnem strežniku nameščen poseben program, »pull CMS«. Ko se na spletni strani pojavi zahteva po dostopu do neke vsebine, ta program iz podatkovne baze »povleče« (iz angl. besede pull) vse potrebne podatke in jih združi v eno spletno stran [9].



Slika 6: Pull CMS sistem, [9]

Slika 6 prikazuje primer arhitekture »pull« sistema, kjer se CMS sistem nahaja lahko nekje na internetu, intranetu ali pa ekstranetu, podatkovna baza pa se nahaja v nekem lokalnem omrežju.

Nekateri povlečni CMS sistemi, ki se pojavljajo trenutno na trgu so: Drupal, Joomla, Wordpress, Plone, Silverstripe, Xoops, Typo3, TypoLight.

3.4.2.1 Prednosti povlečnega CMS sistema

- **Primeren za hitro spreminjajoče vsebine**

Spletna stran se ob vsaki uporabnikov zahtevi dinamično generira. Tako po ustvarjanju ali spremembi neke vsebine ni potrebe po ponovnem generiranju in nalaganju statičnih strani na spletni strežnik, kot je to potrebno pri potisnih sistemih.

To pa pomeni, da je vsebina, ko je zapisana v podatkovno bazo CMS sistema, takoj na razpolago.

3.4.2.2 *Slabosti povlečenga sistema*

- **Daljši čas nalaganja in večja obremenitev sistema**

Ker se mora spletna stran ob vsakem dostopu vedno znova generirati, je čas nalaganja le-te v primerjavi z časom nalaganja posamezne pri »push« sistemih nekoliko daljši. Zaradi vsakokratnega nalaganja in predpomnenja je bolj obremenjen tudi CMS sistem in podatkovni strežnik.

- **Odvisnost od delovanja CMS sistema**

Ker se vsebina generira na zahtevo je v primeru nedelovanja CMS sistema ali podatkovne baze, postane vsebina spletne strani nedostopna.

- **Višji stroški obratovanja**

V primeru večjih spletnih strani je potrebno za tekoče delovanje zagotoviti zmogljivejšo strojno opremo, kar poveča stroške obratovanja sistema.

3.4.3 **Sklep o potisnih in povlečnih sistemih**

Potisni sistemi imajo v primerjavi z pull sistemi kar nekaj prednosti, vendar je težko trditi kateri pristop je boljši. V določenih okoliščinah, kjer se vsebina ne spreminja hitro je mogoče primernejši push sistem, kjer pa je dinamika spreminjanja vsebine večja pa je boljši pull sistem, pa čeprav morda zahteva zmogljivejšo in posledično dražjo strojno opremo.

Med tema dvema vrstama sistemov gre samo za dva različna tehnološka pristopa, ki se na zunaj niti ne opazita. Tako nek poslovni uporabnik sistema teh razlik ne bo opazil. Razlika je opazna le z vidika načina upravljanja.

4 Predstavitev CMS sistemov

Naloga diplomskega dela je primerjava sistemov za upravljanje spletnih vsebin. Za postavitev testnega okolja sem uporabil Microsoftov navidezni stroj Virtual PC 2007. Uporabil sem operacijske sisteme Windows XP in Windows Server 2003 R2. Za potrebe CMS sistemov sem ustvaril 6 navidezni strojev. Sisteme s podobnimi tehničnimi značilnostmi sem nameščal skupaj.

Na kratko bom predstavil 10 sistemov, in jih med seboj primerjal. Pri izboru sistemov sem se oziral po tistih, ki so bolj znani, se večkrat pojavljajo raznih člankih ali pa so mogoče že prejeli kakšno nagrado. Upošteval sem tudi zahtevo, da so v nabor vključeni sistemi narejeni v različnih programskih jezikih, ter da po možnosti uporabljajo različne podatkovne baze.

Seznam izbranih sistemov:

1. Joomla 1.5 (odprtokodni)
2. Plone 3.1.7 (odprtokodni)
3. Drupal 6.9 (odprtokodni)
4. Typo3 4.2.3 (odprtokodni)
5. DotNetNuke 3 (odprtokodni)
6. WordPress 2.7.1 (odprtokodni)
7. Silverstripe 2.2.2 (odprtokodni/komercialni)
8. Microsoft Office SharePoint Server 2007 (komercialni)
9. Xoops 2.3.2 (odprtokodni)
10. dotCMS 1.7 (odprtokodni)

Sistemi se med seboj precej razlikujejo po funkcionalnosti, tehničnih lastnosti, strukturi uporabniških vmesnikov, načini upravljanja z vsebinami in še čem. Zaradi takšnih razlik, se bom pri predstavitvi posameznega sistema posvetil le konceptom, za katere menim, da so za sistem posebej značilni. Nekatere podrobnosti sistemov bodo zajete v poglavjih, v katerih bom sisteme med seboj primerjal po posameznih parametrih.

4.1 Joomla 1.5

Joomla je odprtokodni sistema za upravljanje s spletnimi vsebinami, ki temelji na jeziki PHP in podatkovni bazi MySQL. Za spletni strežnik običajno uporablja Apache, deluje pa tudi na Microsoft IIS.

Sistem je v razdeljen na ospredje, ki je vidno obiskovalcem in prijavljenim uporabnikom, ter ozadje, ki je namenjeno administratorju spletne strani. Za navigacijo po ospredju so na voljo trije meniji: osnovni (angl. Main Menu), zgornji (angl. Top Menu) in uporabniški (angl. User Menu).

Joomla! 1.5 - "Experience the Freedom!". It has never been easier to create your own dynamic Web site. Manage all your content from the best CMS admin interface and in virtually any language you speak.

...because open source matters

Home

search...

[About Joomla!](#)
[Features](#)
[News](#)
[The Community](#)

Main Menu

- Home
- Joomla! Overview
- Joomla! License
- More about Joomla!
- FAQ
- The News
- Web Links
- News Feeds

Latest News

- Joomla! License Guidelines
- Content Layouts
- The Joomla! Community
- Welcome to Joomla!
- Newsflash 4

Popular

- Joomla! Overview
- Extensions
- Joomla! License Guidelines
- Welcome to Joomla!
- What's New in 1.5?

Welcome to the Frontpage
 Joomla! Community Portal
 Written by Administrator
 Saturday, 07 July 2007 09:54
 The Joomla! Community Portal is now online. There, you will find a constant source of information about the activities of contributors powering the Joomla! Project. Learn about Joomla! Events worldwide, and see if there is a Joomla! User Group nearby.
 The Joomla! Community Magazine promises an interesting overview of feature articles, community accomplishments, learning topics, and project updates each month. Also, check out JoomlaConnect™. This aggregated RSS feed brings together Joomla! news from all over the world in your language. Get the latest and greatest by clicking [here](#).
 Last Updated on Saturday, 07 July 2007 09:54

We are Volunteers
 Written by Administrator
 Saturday, 07 July 2007 09:54
 The Joomla! Core Team and Working Group members are volunteer developers, designers, administrators and managers who have worked together to take Joomla! to new heights in its relatively short life. Joomla! has some wonderfully talented people taking Open Source concepts to the forefront of industry standards. Joomla! 1.5 is a major leap forward and represents the most exciting Joomla! release in the history of the project.
 Last Updated on Saturday, 07 July 2007 09:54

Joomla! Security Strike Team
 Written by Administrator
 Saturday, 07 July 2007 09:54
 The Joomla! Project has assembled a top-notch team of experts to form the new Joomla! Security Strike Team. This new team will solely focus on investigating and resolving security issues. Instead of working in relative secrecy, the JSST will have a strong public-facing presence at the Joomla! Security Center.
 Last Updated on Saturday, 07 July 2007 09:54
[Read more... >>](#)

Joomla! License Guidelines
 Written by Administrator

Millions of Smiles
 Written by Administrator

Who's Online
 We have 1 member online

Advertisement
 Featured Links:
[Joomla!](#)
 Joomla! The most popular and widely used Open Source CMS Project in the world.
[JoomlaCode](#)
 JoomlaCode, development and distribution made easy.
[Joomla! Extensions](#)
 Joomla! Components, Modules, Plugins and Languages by the bucket load.
[Joomla! Shop](#)
 For all your Joomla! merchandise.

Who's Online
 Joomla! is used for ?
 Community Sites
 Public Brand Sites
 eCommerce
 Blogs
 Intranets
 Photo and Media Sites
 All of the Above!

Login Form
 Hi admin,

Slika 7: Ospredje sistema Joomla.

Vsebino predstavljajo članki (angl. Articles). Prikazujejo se na ospredju sistema. Za Joomla je značilen tri nivojski sistem kategorizacije vsebine:

1. Področje (angl. Section).– najvišji nivo
2. Kategorija (angl. Category) – pripada enemu področju
3. Članek – pripada eni kategoriji

Področja in kategorije se lahko seveda poljubno dodaja, tako da število elementov znotraj posameznih nivojev ni omejeno.

Joomla sestavljajo:

- komponente (angl. Components),
- moduli (angl. Modules) in
- vtičniki (angl. Plug-ins).

Komponente predstavljajo (dodatne) funkcionalnosti, vsebujejo svojo poslovno logiko in imajo navadno svoje mesto na kontrolni plošči. Tipično je to na primer spletna trgovina, forum, vmesnik za upravljanje z uporabniki, vmesnik za vnos novice.

Moduli so polja, vidna na ospredju strani, ki navadno prikazujejo podatke iz komponent. Položaj modulov je določen v predlogi strani (angl. Template), ki skrbi za razporeditev in izgled celotne spletne strani. Modul na primer prikazuje zadnjih nekaj novic na prvi strani ali pa število trenutnih obiskovalcev.

Vtičniki so delčki programske kode nameščeni na določena mesta v ogrodje sistema, ki dodajajo ali pa spreminjajo njegovo funkcionalnost. Lahko se na primer uporabljajo znotraj teksta članka, za vključevanje zunanjih večpredstavnostnih vsebin.

Administrator upravlja s sistemom preko nadzorne plošče v ozadju sistema. Ukaze posreduje preko menijev ali gumbov na nadzorni plošči.



Slika 8: Nadzorna plošča sistema Joomla.

Pravice dostopa se urejane preko sistema skupin, ki že obstajajo v sistemu in imajo določene pravice. Razdeljene na dva tipa [10], [11]:

- skupine uporabnikov ospredja – ti se lahko prijavijo samo v ospredje:
 - Registrirani uporabnik (angl.Registered)
 - Avtor (angl. Author)
 - Urednik (angl.Editor)
 - Odgovorni urednik (angl.Publisher)
- skupine uporabnikov ozadja – ti se lahko prijavijo v ozadje in imajo (omejen)dostop do nadzorne plošče:
 - Ravnatelj (angl.Manager)
 - Administrator
 - Super Administrator

4.2 Plone 3.1.7

Odprtokodni sistem Plone temelji na skriptnem jeziku Python in ima vgrajen spletni strežnik Zope. Tudi shranjevanje je vgrajeno, vendar ne v obliki klasične relacijske podatkovne baze, temveč kot predmetna zbirka Zope Object Database.

Uporabniški vmesnik sistema PLONE je sestavljen iz štirih glavnih elementov:

1. Glava strani (angl. Page Header)
2. Vsebina (angl. Content Area)
3. Stranskih orodnih vrstic (angl. Sidebars)
4. Noga strani (angl. Page Footer)

Slika 9 prikazuje vmesnik sistema Plone. Navigacija po hierarhiji vsebine mogoča preko menijev in zavihkov. Vsaka stran ima lahko več t.i. portletov, ki se navadno nahajajo v eni od stranskih orodnih vrstic. Portleti ponujajo različne funkcionalnosti, na primer: povzemajo informacije strani, vsebujejo določene vsebine, vsebujejo povezave, omogočajo navigacijo po strukturi strani, prikazujejo koledar dogodkov in podobno. Lahko bi jih primerjali z moduli, ki jih poznajo drugi CMS sistemi.

The screenshot shows the Plone 3.1.7 user interface. At the top right, there are links for 'Site Map', 'Accessibility', and 'Contact'. Below these is a search box labeled 'Search Site' with a 'Search' button and a checkbox for 'only in current section'. A navigation menu includes 'Home', 'Users', 'Predstavitev', 'News', and 'Events'. A breadcrumb trail shows 'You are here: Home'. On the left, there is a 'Log in' portlet with fields for 'Login Name' (admin) and 'Password' (masked), and a 'Log in' button. Below it is a 'Forgo your password?' link. A 'News' portlet shows a recent article 'Začetek sezone' dated 'Mar 13, 2009'. The main content area features a 'Welcome to Plone' message with a congratulatory note and a 'Get started' section with a list of three steps. On the right, there is a calendar for April 2009. The footer contains copyright information for Plone CMS and links for 'Powered by Plone', 'Valid XHTML', 'Valid CSS', 'Section 508', and 'WCAG'.

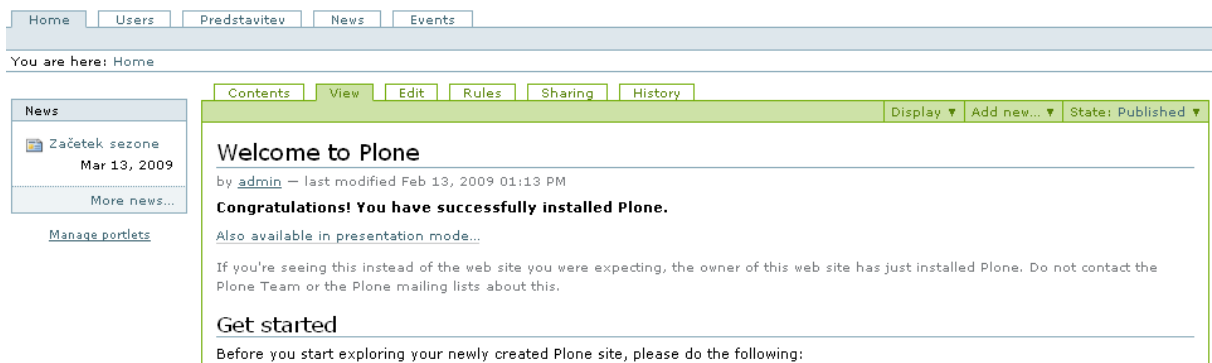
Slika 9: Izgled vmesnika Plone

[12], [13]

Vsebina je v sistemu Plone predstavljena v tako imenovanimi vsebinskimi elementi (angl. Content Items). Plone ponuja več različnih tipov vsebinskih elementov (angl. Item Types).

- Strani (angl. Pages)
- Novice in dogodki (angl. News item and events)
- Slike in datoteke (angl. Images and files)
- Povezave(angl. Links and favorites)
- Mape in zbirke (angl. Folders and collections)

Plone za vsakega od tipov elementov zagotavlja več različnih zaslonskih pogledov (angl. Displays). Vsakič, ko uporabnik dostopa do enega od elementov, bo dostopal preko enega od možnih pogledov, v odvisnosti od tega, kaj želi z nekim elementom delati, na primer gledati, spreminjati, dodajati, premikati po spletni strani. Osnovni pogledi »View«, »Edit«, »Sharing« so skupni vsem tipom vsebine, vsak pa ima lahko še svoje. Pogledi so vidni kot zavihki (angl. Tab) nad vsebinskim delom strani .



Slika 10: Prikaz "pogledov" v sistemu Plone

Plone omogoča registracijo uporabnikov. Vsakemu registriranemu se lahko dodeli vsaj ena vloga. Sam sistem že ima določenih 7 različnih vlog, s katerimi se določa pravice in odgovornosti uporabnikov. Uporabnike lahko združujemo v skupine, tem pa lahko pripišemo določeno vlogo.

The screenshot shows a web interface for managing sharing permissions. At the top, there are tabs for 'View', 'Edit', 'Sharing', and 'History'. The main heading is 'Sharing for "Predstavitev"'. Below this, a message states: 'You can control who can view and edit your item using the list below.' There is a search box labeled 'Search for user or group' with a magnifying glass icon and a 'Search' button. Below the search box is a table with the following structure:

User/Group	Can add	Can edit	Can view	Can review
Logged-in users	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Below the table, there is a checked checkbox labeled 'Inherit permissions from higher levels'. A small text block explains: 'By default, permissions from the container of this item are inherited. If you disable this, only the explicitly defined sharing permissions will be valid. In the overview, the symbol indicates an inherited value. Similarly, the symbol indicates a global role, which is managed by the site administrator.' At the bottom, there are 'Save' and 'Cancel' buttons.

Slika 11: Vmesnik za določanje pravic dostopa v sistemu Plone.

Plone ima zelo prilagodljiv in zmogljiv sistem za predloge imenovan ZPT (angl. Zope Page Template). Sistem temelji na za Zope specifičnih jezikih TAL (Template Attribute Language) in METAL (Macro Expansion Template Attribute Language), ki znata elementa sistema poiskati v objektni podatkovni bazi in jih z njihovimi lastnostmi prikazati v HTML obliki. Ločevanje izgleda od vsebine je zelo podrobno razdelano in natančno določeno. Spreminjanje izgleda je mogoče opraviti tudi preko uporabniškega vmesnika.

4.3 Drupal 6.9

Drupal je odprtokodni sistem za upravljanje s spletnimi vsebinami. Temelji na jeziku PHP in podatkovnih bazah MySQL, PostgreSQL ali Oracle.

Vsaka Drupalova vsebina se imenuje »node«, ki pa nima nič skupnega z vozliščem v omrežju. Predstavljamo si jo kot dokument povezan v sistem. [14] Med te vsebine štejemo naslednje tipe: strani (angl. Pages), ankete (angl. Polls), zgodbe (angl. Stories), vsebino foruma (angl. Forum Topic), objave na blogu (angl. Blog Entry). Vsebine imajo nekatera skupna podatkovna polja kot so: zaporedna št. vsebine, vsebina, tip vsebine, naslov, avtor, status, čas nastanka in čas spremembe. Vsakemu tipu vsebine lahko dodajamo nova polja (npr.: ocene, priponke, komentarje).

S pomočjo modula CCK (angl. Content Construction Kit) lahko tipe vsebine razširjamo. Obstaja veliko razširitev za ta modul, s katerimi dodajamo nova polja (npr.: polja za slike, datume, polja z izračunanimi vrednostmi)

Kako je vsebina razporejena na strani, je določeno z bloki (angl. Blocks), tem pa postavitev določajo teme sistema. Bloki so nekakšni vsebniki (angl. Containers), ki prikazujejo vsebino, pogosto vsebujejo sezname »nodov«. Predstavljeni so kot stolpci na levi in desni strani spletne strani.

Navigacijo omogočajo meniji, ki se prikazujejo v blokkih. Elemente menija lahko poljubno dodajamo in hierarhično strukturiramo.



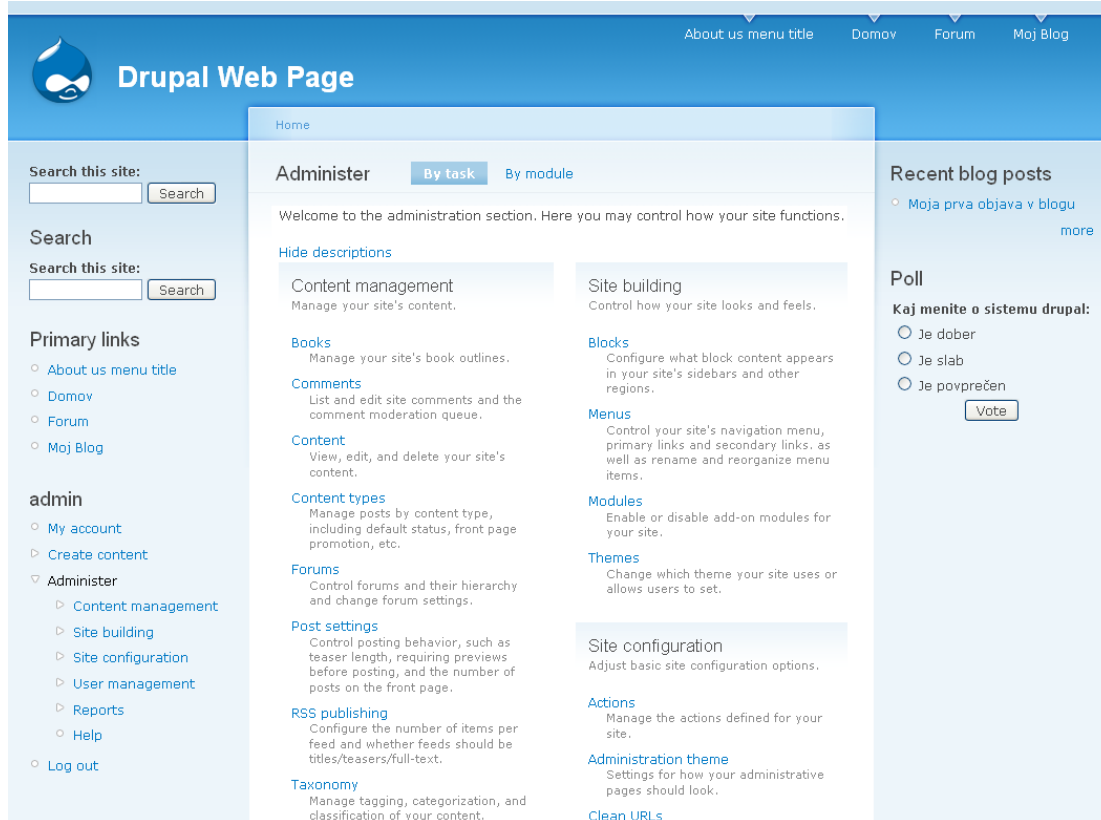
Slika 12: Osrednje sistema Drupal.

Administrator upravlja s sistemom preko administrativnega področja. Za razliko od nekaterih drugih CMS-jev, Drupal nima ločenega ozadja. Slika 13 prikazuje administrativno področje v sistemu Drupal. Trenutno je izbran pogled na opravila »By Task«. Administratorska opravila so združena v pet področij: urejanje vsebine, gradnja strani, nastavitve strani, upravljanje z uporabniki, poročila in pomoč. Obstaja pa tudi pogled »By module«, ki prikazuje opravila za vsak modul posebej.

Dovoljenja za dostop določajo vloge, ki jih dodajamo uporabnikom. Vsak ima lahko tudi več kot eno vlogo. Osnovna namestitvev sistema že predvideva 2 vloge [15], [16]:

- »authenticated users« - uporabniki, ki se prijavijo v sistem
- »anonymous users« - uporabniki brez uporabniškega računa ali neprijavljeni uporabniki.

Vloge je mogoče tudi poljubno dodajati.



Slika 13: Administrativno področje v sistemu Drupal.

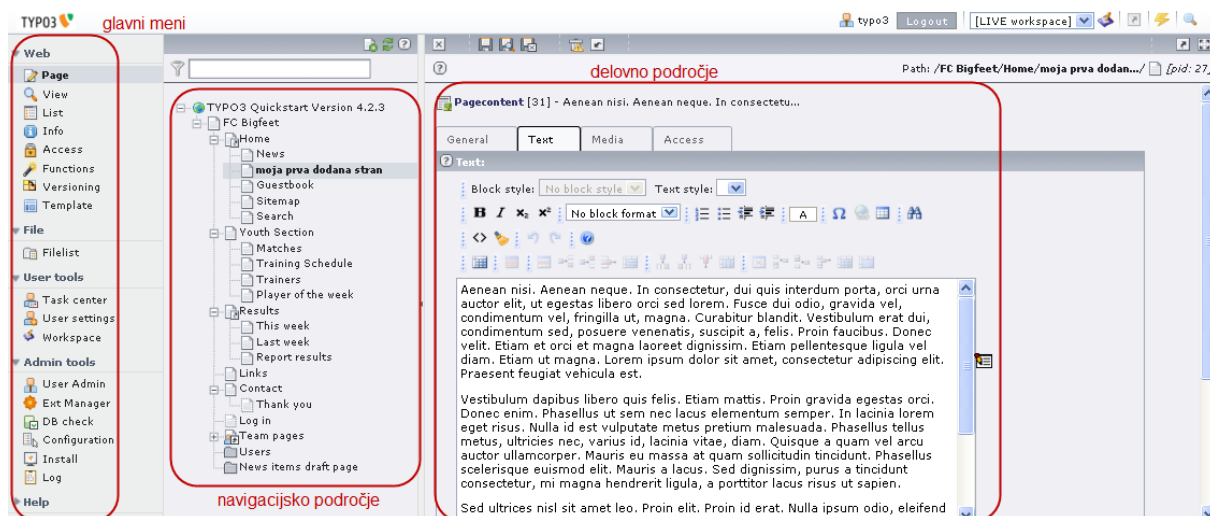
4.4 Typo3 4.2.3

Typo3 je odprtokodni sistem za upravljanje s spletnimi vsebinami. Temelji na jeziku PHP, podpira relacijske baze MySQL, PostgreSQL, Oracle in MSSQL in je neodvisen od operacijskega sistema.

Za Typo3 je značilen njegov jezik TypoScript, s katerim lahko določamo praktično vsako lastnost gradnikov. Ne gre za tipičen programski ali skriptni jezik, ampak gre za neke vrste nastavitveni jezik[17].

Typo3 je razdeljen na ospredje in ozadje. Slednje je seveda namenjeno administraciji spletne strani. Vmesnik je razdeljen na tri področja. Na levi strani je glavni meni, tako imenovani »module bar«, na sredini je navigacijsko področje, kjer je prikazana struktura spletne strani na

desni je delovno področje, »detail view«, kjer se prikaže vsebina glede na to, kaj je izbrano oziroma s čim delamo.

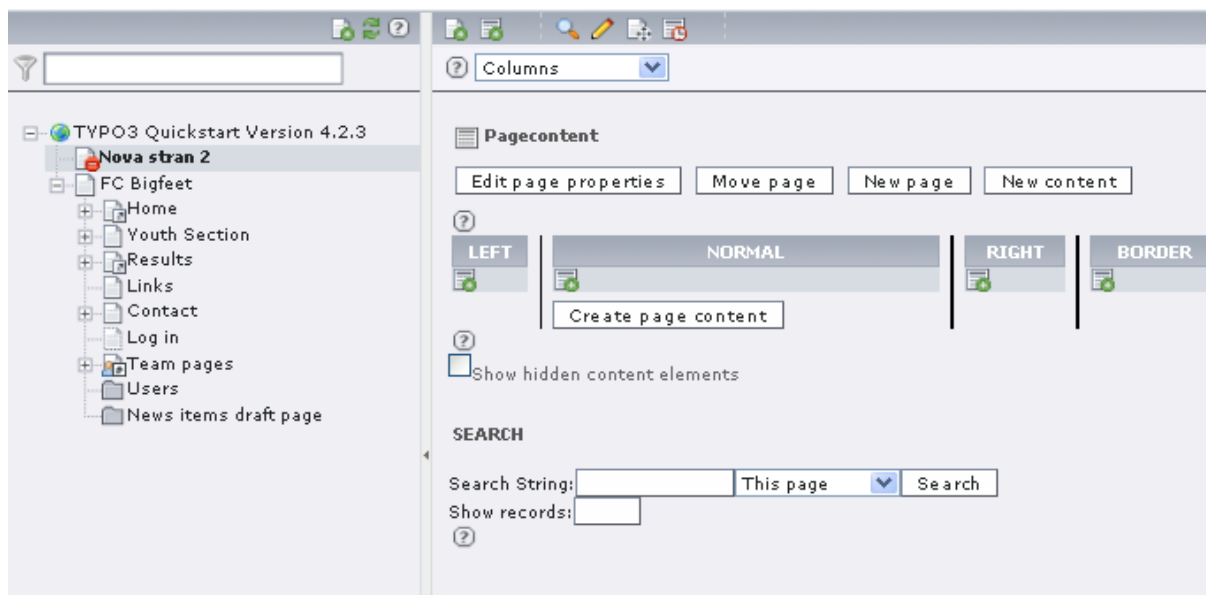


Slika 14: Ozadje sistema Typo3 - dodajanje nove strani in vsebine.

»Module bar« vsebuje funkcionalne enote oziroma module s katerimi delamo v ozadju:

- Stran (angl. Web) - modul je namenjen ustvarjanju in upravljanju vsebine
- Datoteka (angl. File) – modul za upravljanje z datotekami, ki jih stran vsebuje
- Dokument (angl. Doc) - modul, ki omogoča hiter dostop do odprtih dokumentov
- Uporabniška orodja (angl. User Tool) – modul za upravljanje z uporabniki in njihovimi pravicami
- Administratorska orodja (angl. Admin Tool) – modul, ki vsebuje operacije za vzdrževanje sistema
- Pomoč(angl. Help) – modul, ki nudi povezave do pomoči v zvezi s sistemom

Strani so lahko poljubno organizirane v drevesno strukturo, ki predstavlja le ogrodje za vsebino, na posamezno stran pa lahko dodajamo poljubno število elementov vsebine. Tako stran vsebine ne vsebuje in je samo okvir za elemente, ki se ji pripisujejo in se v njej prikazujejo. Znotraj strani je vsebina razdeljena na kolone, ki jih prikazuje Slika 15 (»left«, »normal«, »right«, »border«). V vsako od njih lahko dodamo različne vsebinske elemente. S kolonami tako lahko razdelimo stran na obrobni in glavni del.



Slika 15: Dodajanje vsebine v kolone

Naprej se dodajanje vsebine naredi preko čarovnika, kjer izberemo tip vsebine (tekst, tekst s slikami, slike, seznam, tabela, datoteke, obrazci, skriptpe, HTML,...) in vnesemo vsebino preko »WYSIWYG urejevalnika«.

Typo3 pozna dve vrsti uporabnikov:

- Uporabnik ospredja – uporabnik oziroma obiskovalec, ki je registriran na lastno željo
- Uporabnik ozadja – uporabnik, ki deluje v ozadju (administrator)

Pravice uporabnikov se določa preko skupin uporabnikov. Vsaki skupini se lahko natančno določi do katerih modulov, tabel, tipov strani in dostop in kakšne vrste je ta dostop. Skupine se posebej določi tako za uporabnike ospredja kot tudi za uporabnike ozadja [18], [19].

4.5 DotNetNuke 3

DotNetNuke je odprtokodni sistem za upravljanje s spletnimi vsebinami, ki temelji na Visual Basic .NET. ogrodju ASP .NET. Za delovanje potrebuje strežnik Internet Information Services in podatkovno bazo Microsoft SQL Server [20], [21].

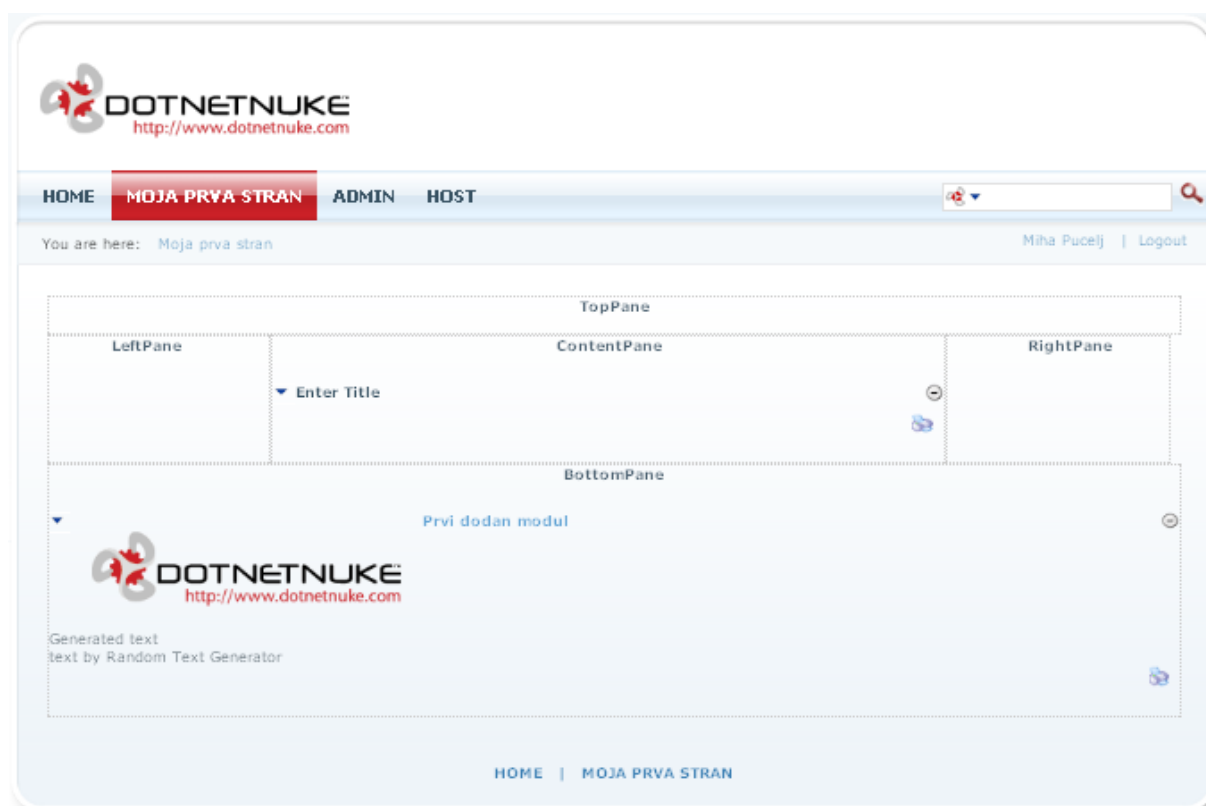
DotNetNuke v osnovni namestitve podpira kar nekaj portalskih funkcionalnosti, ki pokrivajo področje diskutiranja, dogodke, povezave, novice, kontakte, pogosta vprašanja, objave.

Kot večino drugih CMS-jev ima tudi DotNetNuke ločene izgled strani, vsebino strani in aplikacijsko logiko. Za prezentacijo uporablja tako imenovan mehanizem preoblek ali »skinning«.

Sistem DotNetNuke podpira gostovanje več strani. To pomeni, da lahko da z enim samim sistemom, ustvarimo več različnih spletnih portalov, ki pa imajo skupno podatkovno bazo in jih upravljamo na enem mestu. [22]

DotNetNuke za prikazovanje vsebine uporablja dinamično generiranje. Tako gre pri tem sistemu smo za eno fizično spletno stran, ki na zahtevo zahtevano vsebino povleče iz podatkovne baze in jo prikaže na določenem mestu.

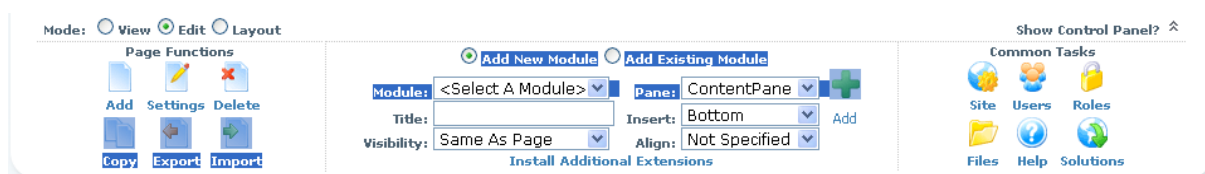
Vsaka stran je sestavljena iz petih področij v katere lahko dodajamo posamezne module. Slednje si lahko predstavljamo kot gradnike strani, vsak pa ima določeno funkcionalnost. Nekateri so na primer namenjeni prikazovanju besedila, pisanju novic, zagotavljanju povezav, drugi shranjevanju kontaktnih podatkov, prikazovanju pozdravnih sporočil. Na ta način vsebino poljubno razporejamo po spletni strani



Slika 16: Razdelitev DotNetNuke strani na 5 področij.

Za DotNetNuke je značilna delitev administratorskih vlog na skrbnika sistema (angl. Host) in in skrbnika portala (angl. Admin). Prvi ima vse pravice na spletnem CMS-ju, slednji pa ima pravice omejene na posamezni portal [23].

DotNetNuke ima razporejene ukaze v menije in kontrolno ploščo, ki je na vrhu strani. Slednja je razdeljena na tri dele: na levi strani so na voljo ukazi za delo s trenutno stranjo, v sredini so polja za vpis osnovnih lastnosti modulov in ukaz za dodajanje modula, na desni pa so bližnjice do pogostih opravil.



Slika 17: Kontrolna plošča sistema DotNetNuke.

Ko uporabnik ustvari novo spletno stran, se poleg tega ustvari tudi navigacijski meni. Tako lahko izgradi poljubno hierarhično strukturo strani.



Vsak modul ima svoje nastavitve, omogoča uvoz in izvoz vsebine, ponujanje vsebine. Obstaja možnost nastavitve koliko časa ostane modul v predpomnilniku. Module je mogoče premikati in jih seveda brisati.


Pravice in odgovornosti uporabnikov so določene z vlogami. DotNetNuke ima v osnovni namestitvi že tri vgrajene vloge, in sicer:

- Administrator
- Registrirani uporabnik (angl. Registered User)
- Naročnik (angl. Subscriber)

Vloge je možno poljubno dodajati in jih hierarhično združevati v skupine. Posameznemu uporabniku lahko dodelimo več vlog.

Pravice dostopa uporabnikov določamo tako, da v nastavitvah modulov in strani, za vsako vlogo, določimo ali ima pravico pregledovanja oziroma urejanja. Podobno kot vlogam, lahko pravice dostopa določimo tudi posameznim uporabnikom.

	View Module	Edit Module
Administrators		
All Users	<input type="checkbox"/>	<input type="checkbox"/>
Home Page Admin	<input type="checkbox"/>	<input type="checkbox"/>
Registered Users	<input type="checkbox"/>	<input type="checkbox"/>
Subscribers	<input type="checkbox"/>	<input type="checkbox"/>
Unauthenticated Users	<input type="checkbox"/>	<input type="checkbox"/>

Username:  Add

Inherit **View** permissions from **Page**

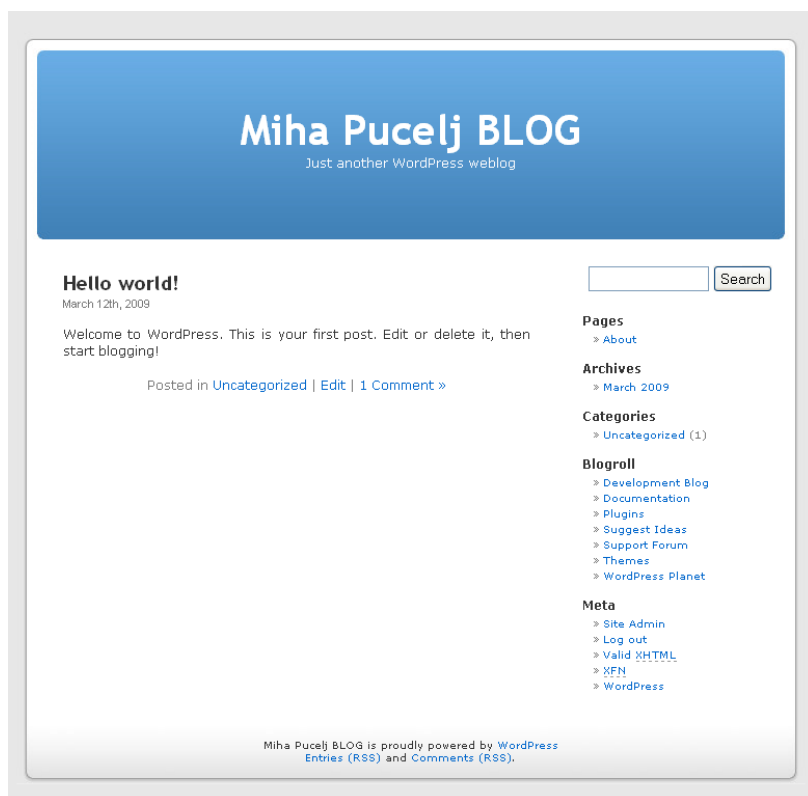
Slika 18: Določanje pravic vlog nad modulom v sistemu DotNetNuke.

4.6 WordPress 2.7.1

WordPress je odprtokodni sistem za upravljanje s spletnimi vsebinami, ki temelji na programskem jeziku PHP in podatkovni bazi MySQL.

Namenjen je predvsem pisanju blogov. Tako se WordPress od ostalih klasičnih CMS sistemov nekoliko odstopa, saj imajo blog sistemi precej značilen zunanji izgled – »neskončno dolga« prva stran objav, ter podpirajo manj različnih tipov vsebine. Čeprav ni tipičen CMS predstavnik, je bil v seznam dodan kot sistem, ki naj bi bil predvsem primeren za profil individualnega uporabnika.

Osnovni oziroma glavni tip vsebine objava (angl. Post). Namenjene so vnosu člankov, spisov, diskusij, premišljevanj avtorjev in pravzaprav razlog za obstoj nekega bloga. Za objave je značilno, da so časovno orientirane in se navadno prikazujejo od najnovejše proti starejšim. Pripadajo eni od določenih kategorij (angl. Category) ali podkategorij, ki so namenjene klasifikaciji vsebine v skupine. To omogoča lažjo navigacijo. Drugi koncept iskanja vsebine pa so oznake (angl. Tags). Oznake so ključne besede, ki jih avtor pripiše objavi in za razliko od kategorij niso urejene v nobeno hierarhijo. Objave imajo tudi možnost dodajanja komentarjev.



Slika 19: Osrednje sistema WordPress.

Drugi tip vsebine v WordPress-u pa so statične strani. Namenjene so objavljanju vsebin, ki se časovno ne spreminjajo, kot so na primer kontaktne informacije in strani »o nas«.

Upravljanje sistema poteka preko administrativne plošče, ki jo prikazuje Slika 20. Razdeljena je na glavo strani, na vrhu, navigacijo na levi strani, delovno površino na sredini in nogo.

Slika 20: Administrativna plošča sistema WordPress.

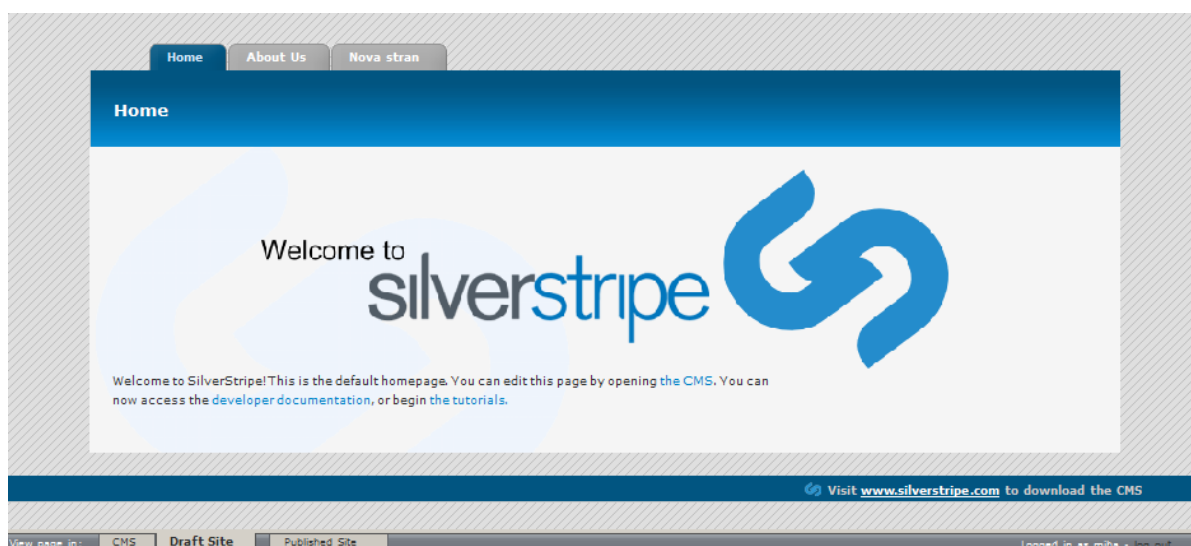
Navigacijski meni prikazuje vse možne administrativne funkcionalnosti, ki jih lahko izvajamo. Glede na izbrano povezavo v meniju, se na delovni površini prikazujejo specifične informacije. Nadzorna plošča pa prikazuje pred kratkim uporabljene aktivnosti.

Razširljivost funkcionalnosti je mogoča preko vtičnikov (angl. Plug-in) [24]. Za prikazovanje določenih delcev informacij v stranskih orodnih vrsticah so na voljo gradniki (angl. Widgets)

ovoljenja uporabnikov so določena z vlogami. WordPress ima že vključenih pet vlog: »Administrator«, »Editor«, »Author«, »Contributor«, »Subscriber«. Vsaka ima že določeno katere naloge lahko opravlja [25].

4.7 Silverstripe 2.2.3

Silverstripe je odprtokodni sistem za upravljanje s spletnimi vsebinami. Napisan je v PHP jeziku, za shranjevanje podatkov pa uporablja podatkovno bazo MySQL. Sistem vsebuje zelo prilagodljivo razvojno ogrodje imenovano »Sapphire«. CMS omogoča popolno kontrolo nad kodo, ki je generirana v označevalnih jezikih. To predstavlja približevanje semantiki in standardom XHTML. Posledično pa to pomeni, da je za gradnjo spletne strani potrebno nekaj tehničnega znanja iz razvoja spletnih strani [26].



Slika 21: Osredje sistema SilverStripe.

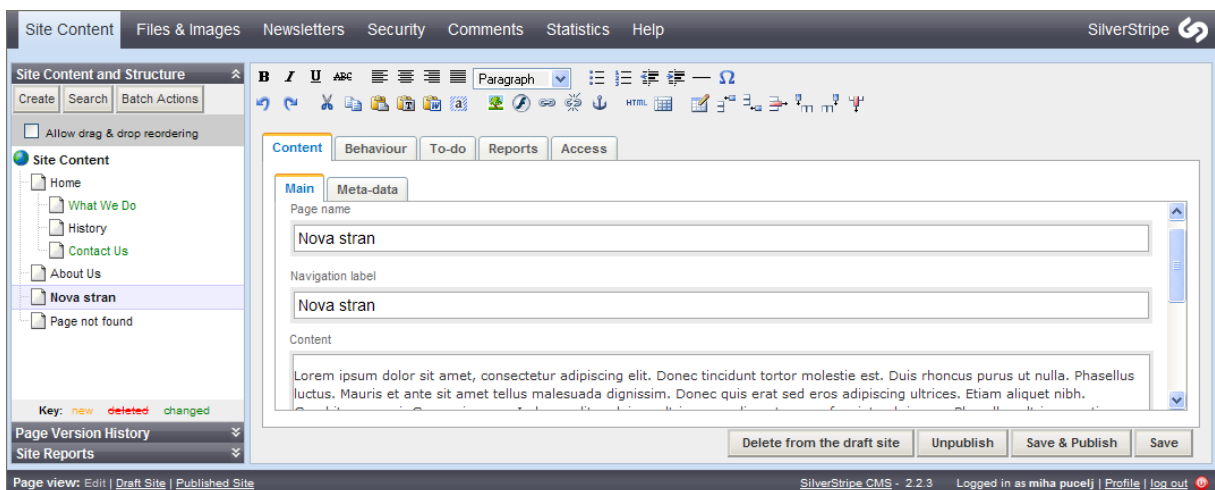
Sistem SilverStripe pozna tri različne poglede na spletno stran: »Published«, »Draft« in pa »CMS«. Prvi prikazuje stran v stanju in obliki, ki jo lahko vidijo neregistrirani obiskovalci. Drugi pogled prikazuje osnutek strani, kar pomeni da prikazuje tudi tiste dele vsebine ki še niso objavljeni. Tretji pogled pa predstavlja ozadje sistema, ki je namenjen upravljanju.

Spletno strani in vsebino se upravlja preko administratorske plošče, ki je razdeljena na sekcije:

- Vsebina strani (angl Site Content)
Sekcija je namenjena ustvarjanju in upravljanju spletnih strani.
- Datoteke in slike (angl. Files and Images)
Sekcija je namenjena upravljanju z datotekami in slikami, ki jih naložimo v sistem.
- Komentarji (angl. Comments)
Sekcija je namenjena upravljanju s komentarji.

- Poročila (angl. Reports)
Sekcija namenjena pregledovanju poročil o spletni strani.
- Varnost (angl. Security)
Sekcija je namenjena upravljanju z uporabniki in skupinami.
- Pomoč (angl. Help)
Povezava do spletnih strani pomoči sistema.

Osnovni gradniki oziroma bloki strani so tako imenovani tipi strani (angl. Page Types). Vsak tip ima eno ali več predlog za prikazovanje podatkov, določeno katera polja podatkov prikazuje in ureja, ter način delovanja – ali gre za navadno tekstovno stran, ali gre za stran, ki pošilja neke podatke, ali gre za formo, ki jo bo določil uporabnik. Slika 22 prikazuje administratorsko ploščo in uporabniški vmesnik za dodajanje nove spletne strani.

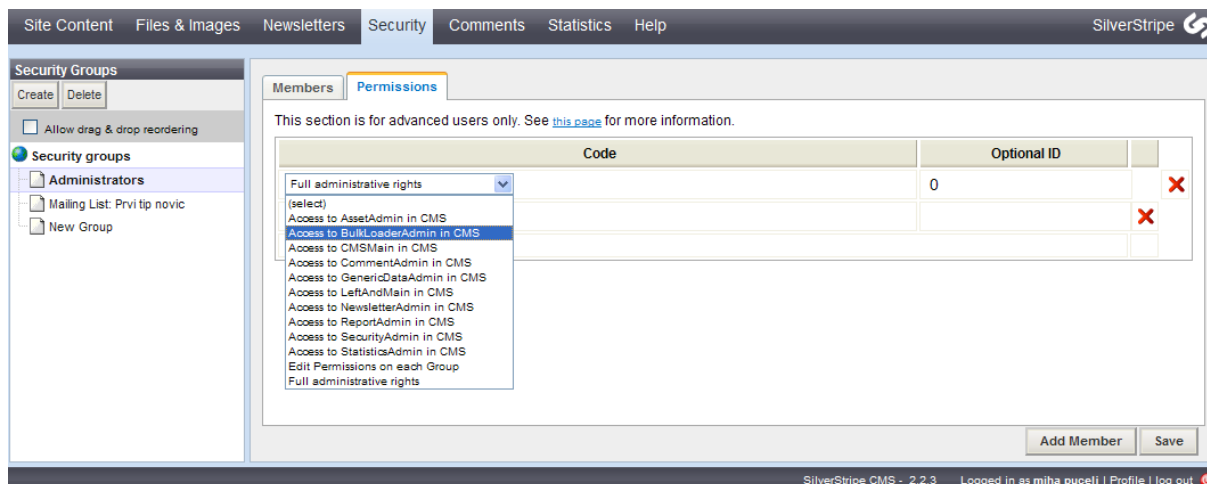


Slika 22: Administratorska plošča sistema SilverStripe.

Spletne strani se lahko strukturira poljubno hierarhično, urejanje pa je mogoče po principu »povleci in spusti«.

Sistem je razširljiv z dodatnimi moduli gradniki (angl. Widgets). Moduli so razširitve, ki predstavljajo nek nov tip strani s svojo predlogo in načinom delovanja. Tukaj gre za različne forume, bloge, galerije in podobno. Gradnik pa predstavlja nek majhen delček funkcionalnosti nekega modula. Vsak modul ima tako vsaj en gradnik [27].

Kontrola dostopa je urejena preko skupin uporabnikov. Skupine lahko poljubno ustvarjamo in hierarhično strukturiramo.



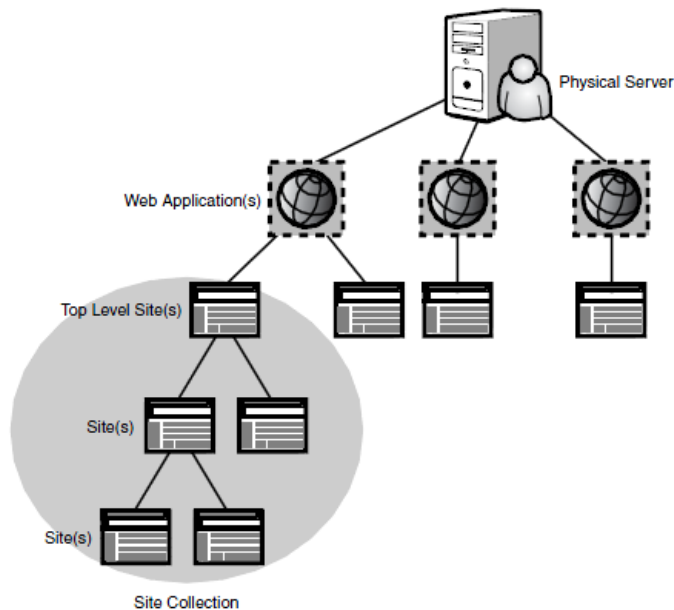
Slika 23: Vmesnik za ustvarjanje skupin in določanje pravic.

4.8 Microsoft Office SharePoint Server 2007

Microsoft Office SharePoint Server 2007 (v nadaljevanju MOSS) je integriran strežniški program, namenjen upravljanju z vsebin– ne samo spletnih . Temelji na Windows SharePoint Services (v nadaljevanju WSS) in ogrodju .NET. Za shranjevanje podatkov uporablja podatkovno bazo Microsoft SQL Server 2005. WSS je zgrajen na tehnologijah in storitvah, ki jih ponuja operacijski sistem Windows Server 2003. WSS zagotavlja storitve shranjevanje vsebine, upravljanja s stranmi, postavljanja spletnih aplikacij, organiziranja spletnih strani in sodelovanja (angl. collaboration) . Storitve sodelovanja v MOSS-u zagotavljajo predvsem funkcionalnosti kot so skupno upravljanje z dokumenti, wikiji in blogi, podpora vodenju projektov, urejanje kontaktov, koledarjev dogodkov in opravil, integracija elektronske pošte in integracija z orodji Microsoft Office [28], [29].

SharePoint je zgrajen v hierarhijo nekakšnih vsebnikov. To prikazuje Slika 24. Na vrhu je strežnik, ki vsebuje spletne aplikacije (angl. Web Applications). Te pa vsebujejo strani (angl. Sites) ali zbirke strani (angl. Site Collections), do katerih dostopajo končni uporabniki.

Zbirka strani je hierarhična množica strani pod skupnim upravljanjem. Te imajo navadno skupno podatkovno bazo in funkcionalnosti kot so na primer deljene pravice, galerija za predloge, tipi vsebine in navigacijo. V primeru portala je najvišja stran v zbirki strani domača stran.



Slika 24: Hierarhija SharePoint vsebnikov [28].

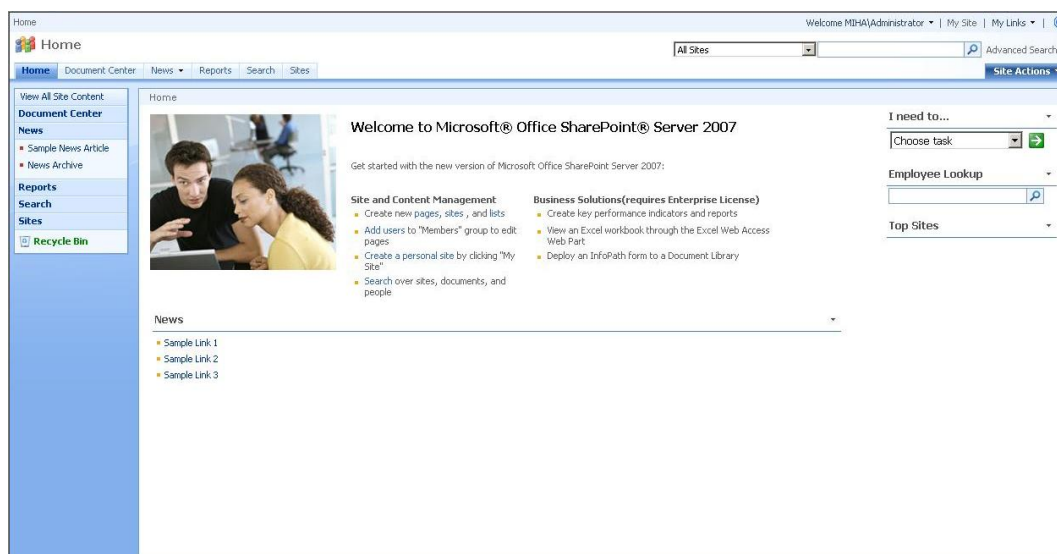
Posamezna spletna stran (angl. Web Site) sestoji iz več strani (angl. Web Pages). Vsaka ima podatkovni repozitorij, vizualne elemente, administracijo in ostale uporabniške funkcionalnosti.

Pomemben koncept v SharePoint-u so tudi sezname (angl. Lists). To so repozitoriji, ki shranjujejo stolpce podatkov ali pa dokumente. Predstavljamo si jih lahko kot analogijo delovnim zvezkom programa Excel.

Funkcionalnosti in način delovanja vsake spletne strani v SharePoint-u določajo predloge (angl. Templates), ki so nekakšen skupek zbirke seznamov, strani in funkcionalnosti imenovanih »Web Parts«. V osnovi SharePoint definira štiri skupine predlog:

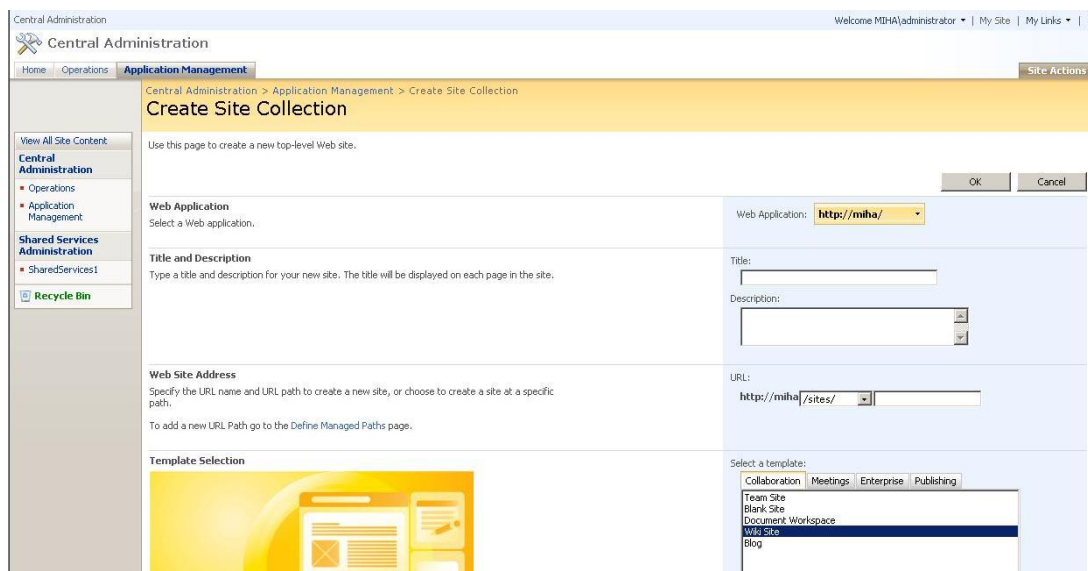
- »Collaboration« - podpora deljenju datotek, wiki-jem in blogom
- »Mettings« – podpora koledarjem, dogodkom, odločanju in opraviom
- »Enterprise« - podpora upravljanju z dokumenti, sledenju projektom, ključnim kazalnikom poslovanja
- »Publishing« - podpora upravljanju spletnih strani in dokumentov, delovnemu toku objav in upravljanju z različicami.

Upravljanju s spletnimi vsebinami je namenjena predvsem predloga »Publishing«.



Slika 25: Osrednje sistema SharePoint 2007.

Administracija SharePoint-a je mogoča na preko centralne administracije (angl. Central Administration), ki je namenjena bolj globalnim nastavitvam sistema, kot je na primer ustvarjanje novih spletnih aplikacij na stražniku, ter administracija na nivoju posamezne spletne strani, ki omogoča upravljanje z nastavitvami zbirke strani, z izgledom strani, z pravicami na nivoju strani ter seznama in knjižnicami.



Slika 26: Centralna administracija – izbira predloge strani za novo spletno aplikacijo.

4.9 Xoops 2.3.2

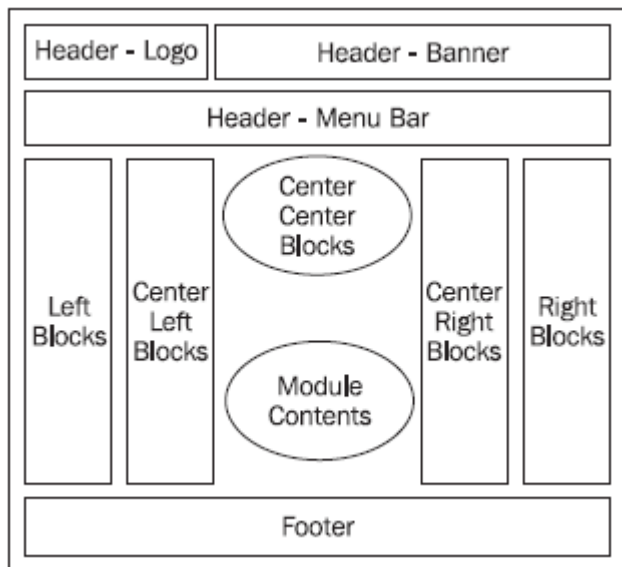
Xoops je kratica za razširljiv objektno orientiran sistem za upravljanje s spletnimi vsebinami, ki izhaja iz angleške besedne zveze eXtensible Object Oriented Portal System. Za podatkovno bazo uporablja MySQL, ima modularno zgradbo in je napisan v programskem jeziku PHP. Deluje na strežniku Apache. Omogoča personalizacijo posameznih delov portala.



Slika 27: Osprejje sistema Xoops.

Vsebina in navigacijski elementi so postavljeni s pomočjo blokov. Vsebina je prikazana preko modulov, ti pa so postavljeni v enega izmed blokov sistema.

Izgled, strukturo celotne spletne strani in skupnih elementov posameznih podstrani v sistemu (npr. glava, noga širina stolpcev, barve,...)določa tema (angl. Theme) oziroma preobleka. Poleg teme pa obstajajo tudi predloge, ki kontrolirajo izgled in razporeditev elementov strani, kot so bloki (glavni meni, uporabniški meni, zadnje objave, ...) in moduli. Slika 28 prikazuje osnovno razporeditev elementov strani, ki je določena s temo. Prilagoditve predlog so mogoče preko administratorskega vmesnika, medtem, ko je za preoblikovanje teme potreben zunanji urejevalnik HTML datotek.



Slika 28: Osnovna razdelitev strani v Xoops-u [30].

Pravice dostopa in administratorske pravice se uporabnikom določa preko prilagodljivega sistema pravice, ki temelji na skupinah. Vsaka skupina ima lahko določeno katere bloke lahko vidi, do katerih modulov lahko dostopa ali jih ureja, in do katerih aspektov administracije ima dostop [30] , [31].

Title	Block Description	Module	Side Left-Center-Right	Visible in	Cache lifetime	Visible	Weight	Action
User Menu	User Menu	System	<input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	All Pages Top Page	No Cache	<input checked="" type="radio"/> Yes <input type="radio"/> No	0	Edit Clone
Login	Login	System	<input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	All Pages Top Page	No Cache	<input checked="" type="radio"/> Yes <input type="radio"/> No	0	Edit Clone
Search	Search	System	<input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	All Pages Top Page	No Cache	<input type="radio"/> Yes <input checked="" type="radio"/> No	0	Edit Clone
Waiting Contents	Waiting Contents	System	<input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	All Pages Top Page	No Cache	<input type="radio"/> Yes <input checked="" type="radio"/> No	0	Edit Clone
Main Menu	Main Menu	System	<input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	All Pages Top Page	No Cache	<input checked="" type="radio"/> Yes <input type="radio"/> No	0	Edit Clone
Site Info	Site Info	System	<input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	All Pages Top Page	No Cache	<input type="radio"/> Yes <input checked="" type="radio"/> No	0	Edit Clone

Slika 29: Ozadje sistema - administracija blokov.

4.10 dotCMS 1.7

Sistem dotCMS je odprtokodni sistem za upravljanje s spletnimi vsebinami, ki temelji na programskem jeziku Java. Podpira poljubne relacijske podatkovne baze, pogoje je le ustrezen JDBC gonilnik. Kot večina ostalih CMS ima ločen izgled od podatkov, dotCMS je vsebuje veliko funkcionalnosti kot so na primer navidezno gostovanje (angl Virtual Hosting), gručenje (angl. Clustering) [24], strukturirana vsebina, predpomnenje, predloge, koledar dogodkov, poročila in še veliko drugih.

The screenshot shows the dotCMS 1.7 starter site homepage. At the top left is the dotCMS logo with the text 'dotCMS starter site.' and 'Java CMS'. To the right are links for 'Login/Register' and 'dotCMS.org'. Below the logo is a navigation bar with links: HOME, GETTING STARTED, NEWS, CALENDAR, BLOGS, STORE (BETA), and a search bar with a 'go' button. The main content area is divided into several sections:

- Welcome Message:** 'Thank you for choosing dotCMS™. Welcome to the dotCMS Core starter site. Whether you are building your 1st web site in the dotCMS or your 100th, the templates provided in the Core starter site are designed to give you a jump start on your next web project.' A 'Let's Get Started »' button is present.
- Need Help?:** A section providing help and support information, including a mailing list subscription link and archives.
- Four Macros You Need to Know!:** A list of macros:
 - Pull Content:** Dynamically pull content from the content database and display it on your website.
 - Menu Generator:** Dynamically build your web site navigation.
 - Displaying Resized Images:** Automatically resize images with predefined widths, heights, or both.
 - Photo Gallery:** Create and display photos in a gallery format by just pointing to a folder on the tree.
- Recent News:** A sidebar section with two news items: 'Headline News Test 2' (Nov 09, 2008) and 'Headline News Test 1' (Oct 07, 2008).

The footer contains a grid of links organized into columns: Home, dotCMS, Products, Services, onDemand, Community Tools, and Company.

dotCMS	Products	Services	onDemand	Community Tools	Company
Demo	dotCMS Core	Support	dotCMS onDemand	Documentation	dotCMS, Inc.
Download	Starter Sites	Training	Professional	api	News
Documentation	Comparison	Web Design	Enterprise	Mailing List	Jobs
Contact Us		Consulting	Custom Hosted	RoadMap	RSS Feeds
		Implementation	Appliance	Other dotCMS Sites	

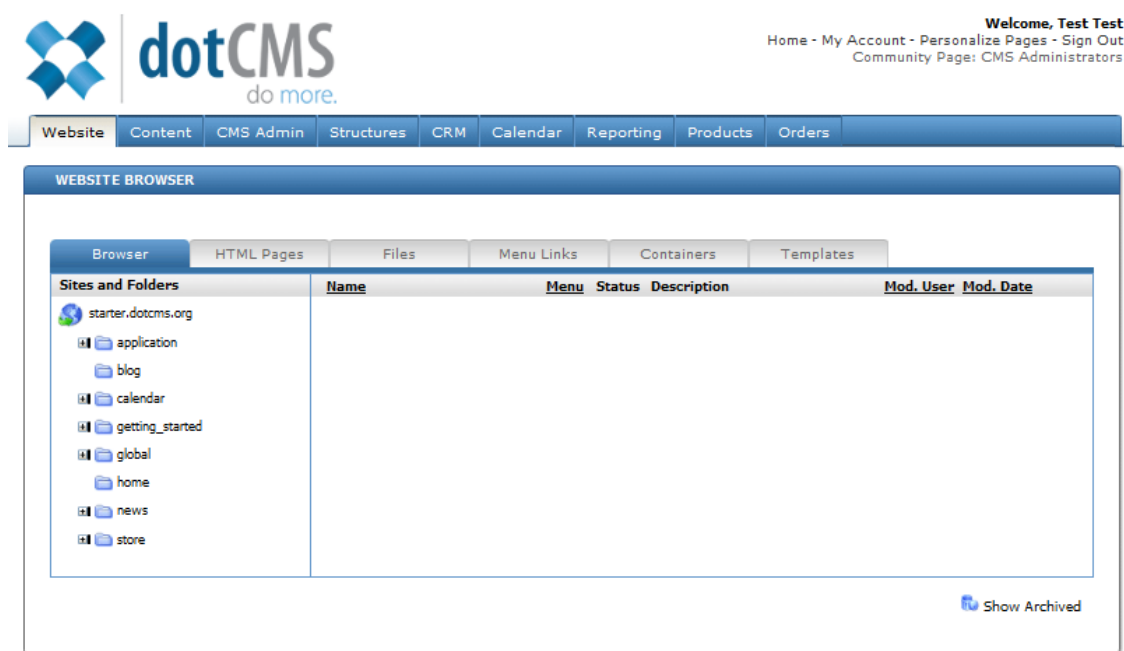
© 2009 dotCMS, Inc. | Miami FL 33133 305.858.1422 | Contact Us

Slika 30: Osprede sistema dotCMS.

Pomembni koncepti v sistemu dotCMS so predloge (angl. Templates), vsebniki (angl. Containers) [24], strukture (angl. Structures) in vsebina (angl. Content).

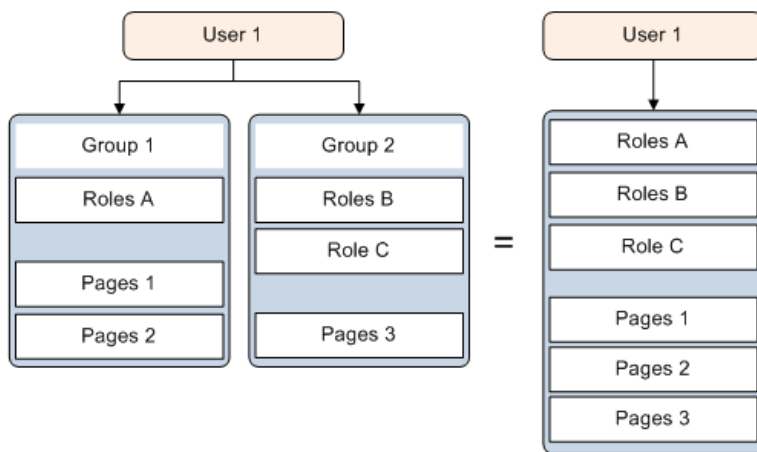
Predloge so načrt razporeditve strani. Gre za XML datoteke, ki vsebujejo in določajo položaj vsebnikov na strani. Slednji so tako namenjeni prikazovanju različnih delčkov informacij kot s glava in noga strani, navigacijski elementi, slike in seveda vsebina. Vsebniki tudi določajo izgled vsebine, ki se preko njega prikazuje.

Strukture so namenjene organiziranju informacij. Določajo polja preko katerih se bo shranjevala vsebina. Polja so na primer lahko: naslov, podnaslov, opis, telo. Administrator s pomočjo struktur določi tipe vsebine, ki se prikazuje preko vsebnikov. Nova vsebina je vedno dodana tako, da se izbere neko strukturo in nato izpolni predpisana polja. Strukture se lahko med seboj povezujejo in združujejo v nove [32], [33]



Slika 31: Ozadje sistema dotCMS [32]

dotCMS ima kompleksen sistem dodeljevanja dovoljenj, s katerim lahko uporabnikom omejuje dostop do vseh objektov v ozadju, vključno do struktur, map, strani, kategorij. Uporabnikom se omejuje in dovoljuje dostop z vlogami, ki se med seboj izključujejo, kar pomeni, da ima lahko vsak uporabnik lahko vlog. Poleg vlog, pa lahko uporabnikom pripisujemo tudi skupine. Te združujejo različne vloge in strani. Top pomeni, da ima uporabnik, ki pripada določeni skupini, vse pravice določene z vsebovanimi vlogami ter dostop do vseh strani, do katerih ima ta skupina dovoljenje. Slika 32 prikazuje omenjen koncept dodajanja pravic [32].



Slika 32: Shema dodeljevanja pravic v sistemu dotCMS [32].

5 Uporaba teorije odločanja pri izbiri CMS

Glavni cilj diplomske naloge je primerjava sistemov za upravljanje spletnih vsebin. V ta namen sem izdelal odločitveni model, na podlagi katerega bom določil kateri sistem je primeren za različne profile uporabnikov. Zato se bom v tem poglavju posvetil nekaterim osnovnim konceptom teorije odločanja.

5.1 Odločanje

Odločanje razumemo kot proces, s katerim želimo doseči zastavljene cilje, in to tako, da izbiramo med več alternativnimi potmi ali rešitvami. Z izbiro dajemo prednost določenim rešitvam in sicer tistim za katere predvidevamo, da nas bodo najhitreje, najlažje ali najbolj zanesljivo popeljale do cilja ali pa vsaj bližje tega cilja.

Odločamo se, da bi rešili odločitveni problem. Problem skušamo rešiti tako, da izberemo eno izmed različnih rešitev, alternativ, in to tako, da bi dosegli zastavljene cilje. Odločevalec proučuje in tehta različne lastnosti alternativ, ki opredeljujejo njihovo kvaliteto ali zmožnost doseganja ciljev. Govorimo o kriterijih, na podlagi katerih odločevalec določenim alternativam daje prednost, preferenco, na koncu odločitvenega procesa pa se odloči za eno izmed alternativ – sprejme odločitev.

Pri odločanju imamo le redko popolne podatke o odločitvenem problemu in popoln nadzor nad odzivom okolja, zato odločitveni procesi pogosto potekajo v pogojih negotovosti, pri čemer mora odločevalec sprejeti določeno tveganje [34].

5.2 Elementi odločitvenega procesa

Osnovni elementi odločitvenega procesa so množica variant, preferenčna relacija in funkcija koristnosti.

5.2.1 Množica variant

Z izrazi alternative, variante, različice ali možnosti označujemo različne izbire, ki jih imamo na voljo pri odločanju. Najpogosteje izbiramo med seboj primerljivimi objekti, dogodi ali aktivnostmi.

5.2.2 Preferenčna relacija

Pri določanju dajemo nekaterim alternativam prednost pred drugimi. Pravimo da ki damo prednost ali preferenco. V teoriji odločanja se preferenco izraža s preferenčnimi relacijami:

- *Stroga preferenčna relacija*: med alternativama a in b velja takrat, kadar imamo a rajši od b . Zapišemo $a > b$.
- *Indeferenčna relacija*: med alternativama a in b velja takrat, kadar sta obe alternativni enakovredni. Zapišemo $a \sim b$.
- *Šibka preferenčna relacija*: med alternativama a in b velja takrat, kadar nam je a vsaj tako všeč kot b . Zapišemo $a \geq b$.

Odločitvena teorija predpisuje preferenčnim relacijam določene lastnosti.

Za strogo preferenčno relacijo zahteva naslednje lastnosti:

- *Asimetričnost*: če za alternativni a in b velja $a > b$, potem ne velja $b > a$.
- *Tranzitivnost*: če za alternative a, b in c velja $a > b$ in $b > c$, potem velja tudi $a > c$.

Za indeferenčno relacijo veljajo lastnosti:

- *Refleksivnost*: za vsako alternativo a , velja $a \sim a$.
- *Simetričnost*: če za alternativni a in b velja $a \sim b$, potem velja $b \sim a$.
- *Tranzitivnost*: če za alternative a, b in c velja $a \sim b$ in $b \sim c$, potem velja tudi $a \sim c$.

Za šibko preferenčno relacijo velja:

- *Stroga sovisnost*: za vsak par alternativ a in b velja ena od možnosti: $a \geq b$, $b \geq a$.
- *Tranzitivnost*: če za alternative a, b in c velja $a \geq b$, $b \geq c$, potem velja tudi $a \geq c$.

5.3 Funkcija koristnosti

Funkcija koristnosti je element odločanja, ki skladno s strukturo odločitvenega modela, izračunava vrednost alternativ pri vrednosti osnovnih parametrov. Pri večparametrskem modelu združuje te vrednosti v vrednost nadrejenih modelov.

Vzemimo, da imamo množico variant $A: a_1, a_2, a_3, \dots, a_n, \dots$ in preferenčno relacijo P .

Eden ključnih pristopov v teoriji odločanja je, da preferenčne relacije prevedemo na primerjave med realnimi števili. V ta namen vpeljemo vrednostno funkcijo $v(a)$, ki preslika množico alternativ v realna števila in sicer tako, da za vsak par alternativ a in b velja

$$a P b \Leftrightarrow v(a) > v(b)$$

[34].

5.4 Faze odločitvenega procesa

Odločitveni proces je proces, v katerem se sistematično zbira in urejanja znanja. V tem procesu naj bi pridobili dovolj informacij za primerno odločitev, zmanjšali možnost, da bi kaj bistvenega spregledali ter se zavedali tveganj in posledic odločitve. Proces naj bi potekal hitro in poceni, odločitev pa naj bi bila čim boljša.

Sistematični pristop reševanja odločitvenih problemov proces razdeli na posamezne fazi, v okviru katerih potekajo aktivnosti strukturiranja in razgradnje odločitvenega problema na manjše in lažje obvladljive podprobleme, analiza podatkov o odločitvenem problemu in alternativah ter aktivnosti uporabo vseh teh informacij za čim boljše doseganje ciljev odločanja [34].

5.4.1 FAZA 1: Identifikacija odločitvenega problema

Rezultat spoznanja, da je nastopil tako zahteven odločitveni problem, da ga je smiselno reševati na sistematičen način, nam daje prvo fazo odločitvenega procesa - identifikacija odločitvenega proces. V tej fazi poskušamo definirati problem ter opredeliti cilje in zahteve. Oblikujemo odločitveno skupino, katere jedro sestavljajo odločevalci, ki so v zadnji fazi odgovorni za odločitev. V tej fazi pogosto odločimo tudi za okvirni potek nadaljnjih faz postopa in po potrebi izberemo računalniška podporna orodja za izgradnjo modelov ter vrednotenje in analizo alternativ.

5.4.2 FAZA 2: Identifikacija kriterijev

V tej fazi določimo kriterije, na osnovi katerih bomo ocenjevali variante ter zasujemo strukturo odločitvenega modela. Z modelom se potem ovrednoti alternative in jih med seboj primerja. Pomembno je, da ne spregledamo kriterijev, ki bistveno vplivajo na odločitev. Pri oblikovanju modela poizkušamo izpolniti tudi nekatere druge zahteve, kot so:

- **Struktura odločitvenega modela:** Tu se vprašamo, ali je možno odločitveni model razgraditi na manjše in po možnosti lažje obvladljive odločitvene podprobleme. Rezultat nadgradnje lahko potem opišemo na primer z hierarhijo spremenljivk v večparametrskem modelu ali pa z zaporedjem odločitvenih vozlišč v odločitvenem drevesu.
- **Neredundantnost:** »Redundantnost« pomeni odvečno, nepotrebno. V modelu naj ne bi bilo parametrov, ki ne vplivajo na odločitev.
- **Ortogonalnost:** Gre za zahtevo, po kateri naj bi bilo vsi parametri med seboj čim bolj neodvisni. Vsak pomemben odločitveni dejavnik naj bi bil zastopan samo z enim parametrom, tako da ne bi bilo dveh ali več parametrov, ki bi »vlekli« v isto smer in jo s tem pretirano ojačili.
- **Operativnost:** Parametri morajo biti izbrani tako, da se jih da uporabiti v praksi, kar pomeni, da jih je mogoče razumljivo opredeliti oziroma definirati ter pojasniti. Izbrani morajo biti tako, da se jih da pri vseh alternativah izmeriti ali izraziti z neko zahtevano stopnjo točnosti ali zanesljivosti.

Postopek identifikacije kriterijev navadno poteka po naslednjih korakih:

- Spisek kriterijev – oblikujemo nestrukturiran seznam kriterijev, ki jih bomo upoštevali pri odločanju
- Strukturiranje kriterijev – kriterije hierarhično uredimo, upoštevajoč medsebojne odvisnosti in vsebinske povezave. Nepomembne kriterije in tiste, ki so izraženi z ostalimi, zavržemo in po potrebi oblikujemo nove.
- Merske lestvice – vsem kriterijem v drevesu določimo merske lestvice, to je zalogo vrednosti, ki jih zavzemajo pri vrednotenju, ter morebitne druge lastnosti (npr. urejenost).

5.4.3 FAZA 3: Definicija funkcije koristnosti

V tej fazi definiramo funkcije, ki opredeljujejo vpliv nižjenivojskih kriterijev na tiste, ki so v drevesni strukturi višje, vse do korena, ki predstavlja končno oceno variant. Oblika funkcij in način njihovega zajemanja je odvisen od uporabljene metode. Najpogosteje se uporabljajo preproste funkcije kot so utežena vsota in razna povprečja, srečamo pa tudi zahtevnejše funkcije, ki imajo večjo izrazno moč, vendar pa so zahtevnejše za praktično uporabo: funkcije zvezne logike, funkcije osnovi Bayesovega pravila, odločitvena pravila.

5.4.4 FAZA 4: Opis variant

Vsako varianto opišemo z vrednostmi osnovnih kriterijev, to je tistih, ki ležijo na listih drevesa. Do opisa pridemo z bolj ali manj zahtevnim proučevanjem variant in zbiranjem podatkov o njih. Pri tem sem pogosto srečamo s pomanjkljivimi ali pa nezanesljivimi podatki.

5.4.5 FAZA 5: Vrednotenje in analiza variant

V tej fazi uporabimo odločitveni model, s katerim vrednotimo variante. Za vsako tako pridobimo oceno kvalitete ali koristnosti glede na zastavljene cilje odločitvenega modela. Vrednotenje poteka od spodaj navzgor, v skladu s strukturo kriterijev in funkcijami koristnosti. Varianta, ki dobi najvišjo oceno, je praviloma najboljša.

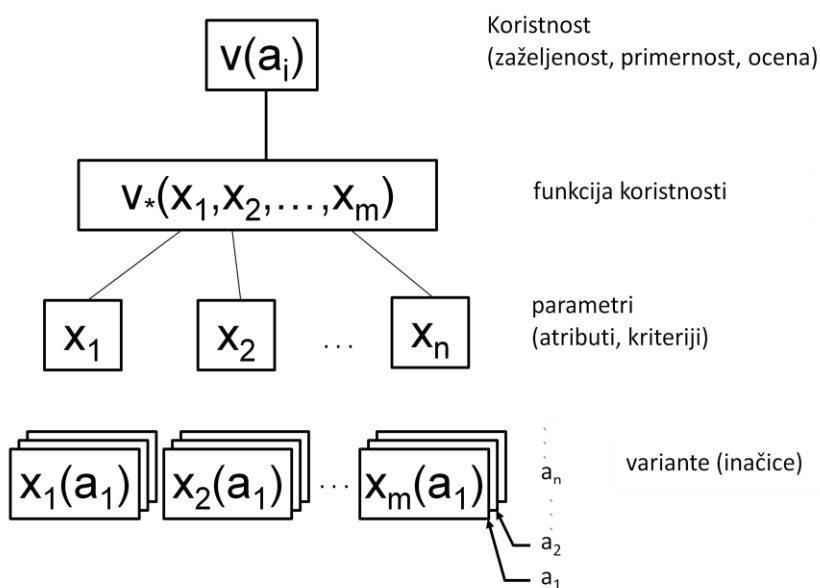
Ocena je lahko do neke mere negotova in nenatančna. Razumeti moramo, pod kakšnimi pogoji se lahko spremeni in kaj to pomeni za našo odločitev. Pri tem si navadno zastavljamo vprašanja, kot so:

- Kako bi bila izračunana končna ocena – na osnovi katerih vrednostih katerih funkcij? Ali so podatki o alternativah in uporabljane metode ustrezne?
- Zakaj je končna ocena takšna kot je? Ali je v skladu s pričakovanji ali odstopa in zakaj? Katere lastnosti kriterijev so najbolj prispevale k takšni oceni?
- Katere so bistvene prednosti in pomanjkljivosti posamezne variante?
- Kakšna je občutljivost odločitve? Kako spremembe vrednosti kriterijev vplivajo na končno oceno? Ali je mogoče in kako variante izboljšati? Katere spremembe povzročijo bistveno poslabšanje ocen?
- V čem se variante bistveno razlikujejo med seboj?

5.5 Večparametersko odločanje

Značilnost večparameterskega odločanja je, da hkrati spremljamo in primerjamo več kot le eno lastnost neke variante. Temelji na razgradnji odločitvenega problema na manjše podprobleme. Variante razgradimo na posamezne parametre (kriterije, attribute) in jih ločeno

ocenimo glede na vsak parameter. Končno oceno variante dobimo z nekim postopkom združevanja. Tako izpeljana vrednost je potem osnova za izbor najustreznejše variante.



Slika 33: Večparameterski odločitveni model [35].

Vrednotenje variant pri večparametrskem odločanju tako poteka na osnovi večparametrskega odločitvenega modela, ki je v splošnem sestavljen iz treh komponent (slika 10). Vhod v model predstavljajo parametri (atributi, kriteriji) x_i . To so spremenljivke, ki ponazarjajo podprobleme odločitvenega problema, to je tiste dejavnike, ki opredeljujejo kvaliteto variant. *Funkcija koristnosti* v^* je predpis, po katerem se vrednosti posameznih parametrov združujejo v $v(a_i)$, ki ponazarja končno oceno ali koristnost variante.

Variante opišemo po osnovnih parametrih z vrednostmi a_i . Na osnovi teh vrednosti funkcija koristnosti določi končno oceno vsake variante. Varianta, ki dobi najvišjo oceno, je praviloma najboljša.

5.5.1 Metoda večparameterskega odločanja

MAUT je pomembna skupina metod večparameterskega odločanja, osnovana na teoriji večparameterske koristnosti (angl. Multi-Attribute Utility Theory)[34].

Osnovna načela te skupine so:

- Strukturo odločitvenih parametrov opisujejo z drevesom ali hierarhijo parametrov

- Notranja vozlišča strukture oziroma izpeljani parametri so zvezne spremenljivke, ki uporabljajo enotno preferenčno mersko lestvico. Gre za lestvico, postavljeno na nekem intervalu 0 – 10 ali 0 – 100. Spodnja meja intervala ustreza nezaželeni vrednosti parametra, zgornja pa najbolj zaželeni.
- Vhodni parametri (osnovni parametri) so zvezne spremenljivke, ki jih izražamo in merimo v »naravnih« enotah (npr.: ceno v denarnih enotah)
- Vrednotenje alternativ poteka z združevanjem vrednosti od spodaj navzgor. To pomeni od osnovnih parametrov proti korenu strukture modela. Združevanje poteka v skladu s funkcijami koristnosti, ki so dveh vrst:
 - Koristnost vsakega posameznega parametra določimo z osnovnimi funkcijami koristnosti. Gre za preslikavo, ki »pravo« vrednost osnovnega parametra preslika v stopnjo njegove zaželenosti. Funkcijo imenujemo osnovna, ker se nanaša na osnovne parametre modela.
 - Združevanje preferenc od spodaj navzgor opravimo s funkcijami združevanja. Pri tem najpogosteje uporabimo kar uteženo vsoto.

5.5.2 Utežena vsota

Pri večparameterskih metodah je potrebno pogosto določiti uteži parametrov. Poznamo dva načina:

1. Dogovorimo so, da ima najpomembnejši parameter neko najvišjo konstanto, na primer 100, potem pa parametre določimo relativno glede na najpomembnejši parameter.
2. Utež 100% razdelimo med vse parametre, to pomeni, da je vsota vseh uteži enaka 100. Pravimo da utež normiramo na 100%.

Normiranje

Naj bodo $w_1, w_2, w_3, \dots, w_n$ poljubne nenegativne uteži, ki jih želimo normirati na 100%. Za vsak parameter k je potem ustrezna normirana utež enaka ω_k :

$$\omega_k = \frac{100w_k}{\sum_{i=1}^n w_i}$$

Za določanje uteži bom v diplomski nalogi izbral 2. možnost

6 Odločitveni model za izbiro CMS

6.1 Identifikacija odločitvenega problema

Izdelava odločitvenega modela, ki bo različnim profilom uporabnikov pomagal izbrati sistem za upravljanje s spletnimi vsebinami.

6.2 Identifikacija kriterijev

6.2.1 Osnovna zmogljivost

Osnovno zmogljivost sistema razumemo kot množico tistih funkcionalnosti posameznega sistema, ki so vključene že v njegovi osnovni namestitvi. Tako lahko vidimo, kaj lahko od nekega sistema za urejanje spletnih vsebin pričakujemo že v osnovi. Sicer velja, da bi bilo mogoče vsak sistem tako prilagodi, da bi lahko vseboval katerokoli funkcionalnost, vendar kljub temu posamezni produkti dajejo poudarek na različnih osnovnih zmogljivostih. To je pomemben podatek za ocenjevanje prilagodljivosti, saj velja, da naj bi sistem imel dovolj osnovnih funkcionalnosti, ki tako zagotavljajo dobro osnovo za prilagoditev [36].

Funkcionalnost	Opis
Podpora verzijam	Zmožnost sistema, da spremlja vse spremembe neke vsebine in tako za vsako spremembo ustvari novo različico.
Podpora lokalizaciji za vsako stran	Ali sistem podpira možnost, da lahko uporabnik vsako objavo napiše v več različnih jezikih. (angl. Multi-lingual Content)
Podpora lokalizaciji za vsak vsebinski objekt	Ali sistem podpira več-jezičnost za vsak objekt na strani. To pomeni, da se je jezil za prikazovanje odvisen od splošnih nastavitvev samega sistema. (angl. Multi-lingual Content Integration)
Časovno odvisno objavlanje	Zmožnost sistema, da vsebino samodejno objavlja ali odstrani na podlagi datumskih nastavitvev. (angl. Content Scheduling)
Podpora delovnemu toku	Zmožnost sistema, da podpira delovni tok vsebine ali po možnosti tudi drugim poslovnim procesom. (angl. Workflow)
Navigacijski elementi (meniji)	Splošna ocena že vgrajenih elementov za navigacijo po spletni strani. Ali so meniji dovolj pregledni, da jim lahko sledimo in jih uporabljamo?

Metapodatki	Zmožnost sistema, da zagotavlja različne dodatne tipe polj, polja sama oziroma dodatne oznak za vnos meta podatkov.
Sočasno urejanje vsebin	Zmožnost sistema, da podpira sočasno urejanje vsebin več uporabnikov (checkout).
Iskalnik vsebine	Modul, ki omogoča iskanje po vsebini.
Gostovanje več strani	Funkcionalnost sistema, ki omogoča gostovanje več spletnih strani (angl. Multi-Site Deployment). To pomeni, da sistem enkrat postavimo, potem pa lahko na enostaven način ustvarimo več različnih spletnih strani.
Prispevki uporabnikov	Zmožnost sistema, da uporabniki na enostaven način lahko prispevajo vsebino.
Upravljanje s povezavami	Modul, ki omogoča enostavno upravljanje z povezavami.
Upravljanje z datotekami	Modul, ki omogoča enostavno upravljanje z datotekami (nalaganje, brisanje, ustvarjanje map)
Upravljanje s kontakti	Modul, ki omogoča enostavno upravljanje s kontaktnimi naslovi.
Upravljanje z dokumenti	Modul, ki omogoča nalaganje, shranjevanje in upravljanje z dokumenti.
Naročanje	Zmožnost sistema, ki uporabnikom omogoča naročanje na prejemanje različnih vrst obvestil. Navadno gre za prejemanje novic, novih objav ali obvestil o spremembi vsebine. (angl. Subscriptions)
Podpora prikazovanju na mobilnih napravah	Zmožnost sistema, da lahko prilagodi izgled in strukturo uporabniškega vmesnika primerno za mobilne naprave

Tabela 1: Seznam parametrov za ocenjevanje osnovne zmogljivosti.

6.2.2 Razširljivost z dodatnimi funkcionalnostmi

Raziskal bom katere in koliko dodatnih modulov, komponent ali vtičnikov za posamezen sistem še obstaja. Nekateri sistemi lahko ponujajo določene funkcionalnosti že v osnovni namestitvi, drugi pa kot dodatne module. Predvsem za odprtokodne sisteme je značilno, da je večina modulov prosto dostopnih, nekateri pa so pod komercialno licenco in je za njih potrebno odšteti nekaj denarja. Kot dodatne funkcionalnosti bom štel tudi nekatere module in komponente, ki so sicer mogoče že vključene v osnovni namestitvi CMS sistema, vendar pa presegajo nabor osnovnih zmogljivosti, ki sem jih določil v prejšnjem razdelku.

Parameter	Opis
Preglednost obstoječih modulov	Preglednost spletnih strani, ki ponujajo dodatne module.
Število modulov	Splošna ocena števila modulov, ki jih je mogoče dobiti na trgu.
Forum	Aplikacija oziroma modul, ki omogoča postavitev foruma.
Blog	Aplikacija oziroma modul, ki omogoča pisanje bloga.
Galerija slik	Aplikacija oziroma modul, ki omogoča postavitev galerije slik.
Pogosto zastavljena vprašanja	Modul, ki omogoča enostavno upravljanje z najbolj pogostimi vprašanji in odgovori.
Koledar dogodkov	Modul, ki omogoča grafično prikazovati in upravljati z dogodki.
Časovno sledenje uporabnikov	Modul, ki omogoča spremljanje uporabnikove prisotnosti na sistemu. Modul je lahko uporabljen v podjetjih za spremljanje prisotnosti na delu.
Ankete	Modul, ki omogoča ustvarjanje kratkih spletnih anket.
Ankete z vprašalnikom	Modul, ki omogoča enostavno ustvarjanje in upravljanje z anketami, ki vsebujejo več vprašanj.
Zaslonska maska za elektronsko pošto	Modul, ki omogoča uporabnikom enostavno pošiljanje elektronske pošte.
Zaslonska maska za vnos podatkov	Ali obstaja modul, ki vnos podatkov v poljubno definirana podatkovna polja?
Poročilo o podatkovni bazi	Modul, ki omogoča generiranje poročil o stanju podatkovne baze
Poročilo o stroških	Modul, ki omogoča spremljanje stroškov.
Oglaševanje (angl. Classifieds)	Modul, ki omogoča upravljanje z oglasnimi sporočili.
Knjiga gostov	Modul, ki omogoča uporabnikom enostaven vnos njihovih mnenj.
Vodenje projektov	Modul, ki omogoča spremljanje in upravljanje z nalogami projekta.
Prijava napak	Modul, ki omogoča zbiranje in obravnavanje prijav uporabnikov o napakah, vprašanjih...
Podpora skupinskemu delu	Modul, ki omogoča enostavno komuniciranje med uporabniki sistema, omogoča napredno urejanje njihovih profilov, razporejanje nalog, podporo delovnemu toku.

Kvizi	Modul, ki podpira ustvarjanje in upravljanje s kvizi.
Klepetalnica	Modul, ki omogoča komunikacijo v realnem času.
Objava delovnih mest	Modul, ki omogoča objavlanje in upravljanje z seznamom delovnih mest.
HTTP Proxy	Modul, ki zagotavlja podporo posredovalnemu strežniku za zrcaljenje
Prikaz prisotnosti	Modul, ki prikazuje podatke o trenutno prisotnih uporabnikih.
Grafikoni	Modul, ki omogoča izrisovanje grafikonov na podlagi danih podatkov.
Wiki	Modul, ki omogoča »wiki« funkcionalnost. To pomeni, da uporabniki na preprost način preko spletnega tekstovnega urejevalnika prispevajo vsebino. Pomembna lastnost takšnega modula je označevanje pojmov in posledično povezovanje do drugih strani v povezavi s tem pojmom.
Zemljevid strani	Modul, ki samodejno gradi in prikazuje drevo hierarhično urejenih vseh spletnih strani CMS sistema.
Pregled zaloge	Modul, ki prikazuje stanje zalog.
Vreme	Modul za prikazovanje vremenskih informacij.

Tabela 2: Seznam parametrov za ocenjevanje dodatne funkcionalnosti.

6.2.3 Enostavnost za uporabo z vidika uporabnika

Katere so tiste lastnosti, moduli, komponente, funkcionalnosti, ki zagotavljajo uporabniku enostavno uporabo sistema pri njegovem delu.

Funkcionalnost	Opis
Urejevalnik WYSIWYG	Spletni urejevalnik besedila, ki omogoča uporabnikom oblikovati besedilo brez potrebnega znanja HTM in CSS. (angl. What You See Is What You Get Editor)
»Nazaj« (UNDO)	Možnost razveljavitve sprememb narejene v nekem spletnem dokumentu ali vsebini spletne strani. (angl. Undo)
Povleci in spusti	Zmožnost sistema, ki uporabniku omogoča določiti postavitev neke vsebine na spletni strani na način »povleci in spusti«. (angl. Drag and Drop)
Črkovalnik	Zmožnost sistema, da sproti označuje nepravilno zapisane besede. (angl. Spell Checker)
Nalaganje in spreminjanje velikosti slik	Ali sistem zagotavlja uporabniški vmesnik, ki omogoča enostavno nalaganje slik v sistem. (angl. Image Upload) Nekateri sistemi podpirajo zmožnost, da slikam, ob nalaganju, samodejno spremeni velikost in kvaliteto. (angl. Resize Images)
Vstavljanje slik v besedilo	Zmožnost, da uporabnik lahko sliko vstavi na poljubno mesto v besedilo. Poleg tega nekateri sistemi podpirajo vzorčne slikice (angl. Thumbnail); to so samodejno pomanjšane slike, ki se ob nalaganju strani hitro naložijo, nato pa je mogoče odpreti tudi sliko v osnovni velikosti. (angl. Image Insert)
Paketno nalaganje vsebine	Zmožnost sistema, da uporabnik lahko naloži dokumentov na enkrat. (angl. Mass Upload)
Nalaganje vsebine v stisnjeni obliki (ZIP)	Zmožnost sistema, da lahko uporabnik vsebino naloži v obliki stisnjene datoteke (npr. v ZIP format-u), nato pa sistem to vsebino, navadno gre za statično vsebino, objavi v razpakirani. (angl. Zip Archives)
Uporabniški vmesnik za ustvarjanje in upravljanje vsebine	Splošna ocena uporabniškega vmesnika, preko katerega uporabniki ustvarjajo, objavljajo in tako upravljajo z vsebino.
Nadzorna plošča	Zmožnosti sistema, da uporabniku nudi nadzorno ploščo. Le-ta navadno vsebuje vse tiste elemente, komponente, ali module, za do katerih ima uporabnik dostop oziroma je odgovoren. (angl. Dashboard)
Pozabljeno geslo	Zmožnost, da uporabniku, v primeru da je pozabil geslo, sistem sam posreduje novo ali pa pozabljeno geslo.

Tabela 3: Seznam parametrov za ocenjevanje enostavnosti za uporabo za uporabnika.

6.2.4 Enostavnost za uporabno z vidika administratorja

Katere so tiste lastnosti, moduli, komponente, funkcionalnosti, ki zagotavljajo administratorju enostavno uporabo sistema pri njegovem delu.

Funkcionalnost	Opis
Administratorski vmesnik	Sistem administratorju nudi poseben uporabniški vmesnik.
Vmesnik za enostavno nadgrajevanje sistema	Ali obstaja uporabniški vmesnik, ki administratorju omogoča enostavno nadgradnjo ali pa namestitev popravkov sistema.
Enostavno dodajanje novih funkcionalnosti	Ali obstaja uporabniški vmesnik preko katerega se na enostaven način namesti nove module, komponente ali vtičnike in z njimi upravlja.
Statistika spletne strani	Možnost prikazovanja podatkov kot so število ogledov posamezne strani , število obiskov uporabnikov na časovno periodo.
Lokalizacija administrativnega vmesnika	Ali sistem omogoča krajevno prilagoditev administrativnega uporabniškega vmesnika.

Tabela 4: Seznam parametrov za ocenjevanje enostavnosti uporabe za administratorja.

6.2.5 Oblikovanje izgleda

Pri oblikovanju izgleda bom ocenjeval kako enostavno je spreminjati in oblikovati zunanji izgled spletne strani. Kako lahko reorganiziramo postavite. Ali je mogoče lažje napisati svojo lastno predlogo kot samo zamenjati? Kakšen jezik uporablja predloga? Ali se ga lahko hitro naučimo? Ali je predloga strani napisana v takem formatu, da ga je enostavno razumeti?

Parameter/Lastnost	Opis parametra
Število osnovnih tem	Splošna ocena števila tem je na voljo že v osnovni namestitvi sistema.
Ugnezdene teme	Zmožnost sistema, da je na isti strani mogoče za različne dele strani uporabiti različne teme.
Poljubno določanje položaja vsebine	Zmožnost poljubnega določanja postavitve neke vsebine.
Vmesnik za dodajanje nove teme	Ocena uporabniškega vmesnika za nameščanje novih tem.
Vmesnik za oblikovanje teme	Zmožnost sistema, da nudi uporabniški vmesnik, ki omogoča spreminjanje teme kar v sistemu samemu, na primer tekstovnem oknu
Struktura datotek	Splošna ocena strukture in preglednosti datotek, ki določajo izgled strukturirane in pregledne.
Posebni jezik	CMS sistem ima lahko specifičen jezik za oblikovanje izgleda. Splošna ocena jezika in uporabnosti le tega.

Tabela 5: Seznam parametrov za ocenjevanje oblikovanje izgleda

6.2.6 Interoperabilnost (angl. Interoperability)

Kako je pri posameznih sistemih poskrbljeno za interoperabilnost oziroma med-obraovalnost. V splošnem lahko poteka med-obraovalnost med sistemi na različnih nivojih:

- **Sistemska medsebojno povezovanje (angl. System Interconnection)**
To je najnižji nivo medsebojne povezanosti. Vključuje skupno infrastrukturo, komunikacijske in varnostne protokole. Za primer naj navedem podpora uporabe Aktivnega imenika (angl. Active Directory) in LDAP (Lightweight Directory Access Protocol), ki se uprabljata za avtentikacijo uporabnikov.
- **Integracija in izmenjava podatkov (angl. Data Integration and Interchange)**
Tukaj gre za zmožnost izmenjave podatkov preko skupnih komunikacijskih protokolov. Pomembno vlogo pri tem imajo spletne storitve (angl. Web Services), ki predstavljajo zbirko komunikacijskih specifikacij in protokolov. Za prenos podatkov se uporablja XML format.
- **Aplikacijska integracija (angl. Application Integration)**
Tukaj gre za nivo na katerem lahko posamezni CMS sistemi med seboj delijo vsebino, strukturo in metapodatke. Kot primer naj navedem protokola RSS (angl. Really Simple Syndication) in NewsML za ponujanje in izmenjavo vsebine, ki sta uporabna predvsem na področju novic. Splošnih standardov za izmenjavo vsebine med različnimi CMS sistemi ni, razvijajo se predvsem po potrebi za določena področja (npr.: za potrebe objavljanja novic, standard za področje e-učenja, standardizirane metapodatkov,...) [37].

Parameter	Opis
WebDAV	Ali sistem podpira nalaganje vsebine in datotek preko WebDAV (angl. Web-based Distributed Authoring and Versioning). Gre za množico razširitev protokola http, ki tako uporabnikom omogoča ustvarjanje, spreminjanje in upravljanje z datotekami na oddaljeni lokaciji. Podpira tudi verzioniranje dokumentov [38].
Podpora FTP	Ali sistem podpira nalaganje vsebine in datotek preko FTP protokola?
Ponujanje vsebine	Ali sistem podpira ponujanje vsebine drugim preko protokolov RSS protokola in v XML formatu. RSS (angl. Really Simple Syndication) je skupina protokolov za ponujanje vsebine kot so novice, avdio in video vsebine v standardiziranih formatih.
Podpora XHTML	Zmožnost sistema, da podpira XHTML standard, ki ga določa organizacija W3C. XHTML (angl. Extensible HyperText Markup Language) je kombinacija HTML in XML standarda. Podpira idejo jezika, ki ga je mogoče razčleniti (angl. parse) kot XML, sintaktično pa je še vedno podoben HTML [39] [40].
Podpora uporabi spletnih storitev	Zmožnost sistema, da zagotavlja vmesnik za uporabo različnih spletnih storitev.
Prikazovanje ponujene spletne vsebine (RSS)	Zmožnost sistema, da prikazuje vsebino, ki jo pridobi preko RSS protokola.

Tabela 6: Seznam parametrov za ocenjevanje interoperabilnosti.

6.2.7 Varnost

Varnost v spletnih rešitvah ima veliko različnih delov, glavni cilj pa je zaupnost in zasebnost, celovitost in dosegljivost informacij, ki jih neka organizacija v lasti ali pa jih nadzira. Kršitev katerekoli od teh komponent lahko postavi organizacijo v nevarnost iz odgovornostnega vidika ter vodi k izgubi ugleda. [41]

Za varnost v CMS sistemih poskrbijo različni mehanizmi. Uporabljajo različne sisteme za avtentifikacijo uporabnikov kot so LDAP avtentifikacija, NTLM avtentifikacija, Kerberos avtentifikacija, NIS(Network Information Service).

Sistemi morajo omogočati tudi avtorizacijo uporabnikov. Pravice, ki jih uporabnik nad vsebino ali administracijo CMS sistema ima se določajo z vlogami. Tipično so to vloge uporabnik, avtor, urednik, skrbnik portala, skrbnik sistema. Posamezne vloge uporabnikov lahko pripadajo tudi različnim skupinam, ki lahko so lahko ali pa ne hierarhično urejene. Sistemi imajo sistem vlog in pravic različno podrobno razčlenjen. Temu pravimo tudi granulacija pravic.

Med mehanizmi, ki skrbijo za večjo varnost sistema so tudi možnost uporabe varne povezave preko SSL(angl. Secure Socet Layer) ali TSL (angl. Transport Layer Security) protokola, podpora upravljanju s sejami, beleženje dnevnika prijav, verifikacija naslovov elektronske

pošte, potrjevanje vsebine, omogočena nadzorna sled (angl. Audit Trial) in sistem validiranja vhodnih podatkov. Sistemi podpirajo različne varnostne mehanizme. Ti so lahko vključeni že v osnovnem naboru funkcionalnosti ali pa na voljo kot dodatni moduli.

Komponenta	Opis
LDAP avtentikacija	Ali sistem podpira LDAP (angl. Lightweight Directory Access Protocol). To je preprost omrežni protokol za dostop do imenika. LDAP določa mehanizme za pregled, iskanje in popravljanje vsebine v imenikih. To so univerzalni imeniki, ki lahko hranijo široko paleto različnih vrst informacij, za CMS sisteme pa so največkrat potrebne informacije o uporabnikih. [42]
NTLM avtentifikacija	Zmožnost prijave v sistem z Windows avtentikacijo.
KERBOS avtentifikacija	Omrežni protokol za sporazumevanje dveh vozlišč po nezavarovanem omrežju.
Odobritev objave	Zmožnost sistema, da za vsako ustvarjeno vsebino zahteva odobritev pred objavo.
Nadzorna sled	Zmožnost sistema, da sledi vsem spremembam podatkov.
Peskovnik	Zmožnost sistem, da zagotavlja okolje, ki dovoljuje poizkusno ustvarjanje vsebine brez da bi spremembe vplivale na ostalo vsebino.
Zgodovina prijav	Zmožnost zapisovanje zgodovine prijav uporabnikov.
Upravljanje z sejami	Zmožnost sistema, da nadzoruje seje uporabnikov.
Varne povezave(SSL in TSL)	Zmožnost sistema, da se uporabniki nanj povezujejo preko varnih povezav.
Preverjanje naslova elektronske pošte	Zmožnost sistema, da ob registraciji uporabnika preverja pravilnost elektronske pošte.
Captcha	Zmožnost sistema, da robotom prepreči izvajanje uporabniku namenjenih ukazov.
Dnevnik	
Varnostna kopija	Ali obstaja vmesnik, ki omogoča varnostno ----kopiranje repozitorija?
Kontrola dostopa na nivoju skupin	Administrator lahko določa kontrolo dostopa posameznim skupinam uporabnikov. Posamezni skupini pripadajo lahko uporabniki z določeno vlogo. Skupine se lahko ustvarja poljubno.
Kontrola dostopa z vlogami	Administrator lahko dostop posameznim uporabnikom omeji z njegovo vlogo. Sistemi imajo lahko vloge že v naprej določene. Obstajati bi morala tudi možnost ustvarjanja novih vlog. Vloge so lahko hierarhično urejene, tako da ima vloga na višjem nivoju poleg svojih pravic tudi vse tiste pravice, ki jih imajo vlogo pod njo.
Podrobno določanje dostopa	Ali lahko uporabniku določamo pravice dostopa na nivoju posamezne strani, na nivoju tipa vsebine ali pa se uporabniku določi dostop samo do določenih funkcionalnosti.

Tabela 7: Seznam parametrov za ocenjevanje varnosti.

6.2.8 Skalabilnost

Skalabilnost CMS sistema bom v tukaj ocenjeval po tem, kolikšen obseg prometa posamezni sistem zmore, in ne kot možnost razširljivosti portala, v smislu dodajanja dodatnih funkcionalnosti. Temu bi lahko rekel tudi skalabilnost uporabniškega vmesnika. Pri sistemih za upravljanje spletnih vsebin vse delo z sistemom poteka preko spletnega uporabniškega vmesnik. To pomeni da mora uporabniški vmesnik zagotoviti vse te zahtevne kontrole, ki jih sistem nudi za enostavnejšo uporabo. Za primer vzemimo samo dodajanje ali odstranjevanje dodatnih programskih modulov, ustvarjanje novih uporabnikov ali na primer nalaganje velike količine podatkov. Spletni brskalniki so v primerjavi s klasičnimi odjemalci zelo omejeni kar se tiče podpore različnim tipom programskih kontrol, poleg tega pa hitrost prenosa podatkov od odjemalca do aplikacije omejuje omrežje.

Posledica premajhne skalabilnosti so lahko sesuvanje sistema, počasno odzivanje, kar pripelje do neučinkovitosti v poslovnem procesu.

Kolikšen obseg posamezen sistem zmore je predvsem odvisno od njegove systemske arhitekture in tudi načina organizacije dela s sistemom. Pri slednjem gre za to, na kakšen način lahko uporabnik nalaga večje količine podatkov, koliko podatkov na primer nek uporabnik vidi na enkrat, kakšne so možnosti ponovnega iskanja, da se uporabniku določi dostop samo do vsebine s katero dela in dovoli izvajati le tiste operacije, ki jih v svojem delovnem procesu mora [43].

Določeni zahtevki, ki prihajajo iz http strežnika se pogosto ponavljajo. Ravno zaradi tega, ker so nekateri časovno potratni (recimo klic zunanega podatkovnega stražnika za pridobivanje določenih podatkov), je smiselno, da vključimo za te zahtevke predpomnenje. Kako se bo to predpomnenje obnašalo je odvisno od nastavitvev – ponavadi nastavimo za določen zahtevk maksimalni interval, v katerem se vračajo predpomnjeni podatki. Po tem času se podatki avtomatsko osvežijo iz originalnega vira [41].

Določil bom seznam funkcionalnosti, ki pripomorejo k boljši zmogljivosti sistema. Za vsak sistem bom preveril vsako od teh podpira ali ne.

Funkcionalnost	Opis
Predpomnenje	Ali sistem podpira predpomnenje?
Napredno predpomnenje	Ali sistem podpira napredno predpomnenje, tako, da je mogoče nastavljati lastnosti predpomnenja za posamezne komponente strani.
Statična vsebina	Ali sistem podpira izvoz vsebine v statične strani (HTML), tako da se potem pri ogledovanju vsebine dostopa le do spletnega strežnika ali zunanjega strežnika, ki skrbi za predpomnenje.
Izenačevanje obremenitve	Zmožnost sistema, da sodeluje z izenačevalnikom obremenitve, ki zna zahteve porazdeliti na več strežnikov. Tako ima sistem na razpolago več resursov, kar omogoča hiter odzivni čas tudi v primeru velikega števila zahtev in podatkov. (angl. Load Balancing)

Tabela 8: Seznam parametrov za ocenjevanje skalabilnosti.

Merjenje odzivnosti

Poleg tega bom za vsak sistem opravil meritve odzivnosti posameznega sistema. Za to bom uporabil program ab, ki je del spletnega strežnika Apache.

Vsakemu sistemu bom pošiljal 10000 zahtev. Izvedel bom več ponovitev za vsak sistem. Pri tem bom meril koliko zahtev na sekundo je sistem uspel obdelati.

Rezultate bom za vsak sistem podal v tabelarični obliki.

CMS	1.meritev [št.zaht/s]	2.meritev [št.zaht/s]	3.meritev [št.zaht/s]	Povprečje

6.2.9 Tehnične značilnosti

Tehnične značilnosti posameznega sistema predstavljajo tehnologije, na katerih sistem sloni oziroma jih potrebuje za svoje delovanje. Kriterij v odločitveni model ne bo vključen za ocenjevanje, saj ne more vplivati na oceno variante. Gre za izključitveni kriterij, ki nam pride v poštev samo, če se želimo pri odločanju omejiti le na sisteme z določenimi tehničnimi lastnostmi.

Funkcionalnost	Opis
Aplikacijski strežnik	Kateri aplikacijski strežnik sistem potrebuje za delovanje.
Operacijski sistem	Kateri operacijski sistem potrebuje za delovanje.
Spletni strežnik	Kateri spletni strežnik potrebuje za delovanje.
Programski jezik	Kateri programski jezik potrebuje za delovanje.
Podatkovna baza	Katero podatkovno bazo potrebuje za delovanje.

Tabela 9: Seznam tehničnih lastnosti sistemov.

6.2.10 Popularnost sistema

Popularnost sistema oziroma priljubljenost bom preveril z več parametri.

- **Število prenosov**

Eden izmed glavnih je pokazatelj je gotovo število prenosov posameznega sistema, vendar ti podatki niso najbolj zanesljivi. Lahko se zgodi, da za posamezen sistem sploh ne najdemo podatkov, nekatere strani za prenos sistema so lahko zrcaljene in to na statistiko prenosov ne vpliva, podatki za posamezen sistem so lahko zajeti v različnih obdobjih, poleg tega pa se število prenosov najbolj povečuje ob objavi novih verzij, to pa navadno ne sovпада pri vseh sistemih. Število prenosov je dober pokazatelj, vendar moramo imeti v mislih zgoraj našete težave.

- **Stopnja podpore tretje strani**

Pokazatelj priljubljenosti je tudi stopnja podpore s tretje strani. To lahko preverimo na spletni strani, ki ponuja mehanizme za iskanje prostovoljnih strokovnjakov na trgu, ki se ukvarjajo s programiranjem spletnih tehnologij, www.elance.com.

- **Število napisnih publikacij**

Poleg števila strokovnjakov, ki se ukvarjajo s CMS sistemi, lahko preverimo tudi število napisanih publikacij o posameznih sistemih. It tega lahko vidimo tudi kam nekatera podjetja vlagajo denar.

Število publikacij za posamezni sistem bom preveril na Amazon.com, ki velja za eno največjih spletnih knjigarn.

- **Število vhodnih povezav** (angl. Inbound connections)

Kako so posamezni CMS Sistemi vidni iskalnim robotom bom lahko določimo s številom vhodnih povezav, to pomeni iz koliko različnih spletnih strani obstaja povezana na primer uradno domačo nekega spletnega portala. To število pomembno vpliva na rang in relevantnost spletne strani. Povezav na neko spletno stran ni nobeden prisilen dodajati. Vedno, ko je povezava na neko stran dodana, je to zato, ker lastnik tega spletišča vidi v tem, da je njegova stran povezana z nekom projektom, neko vrednost.

Število Vhodnih povezav bom preveril s pomočjo spletnega iskalnika Google.

Kako dobro se posamezni projekti odrežejo pri uvrščanju med rezultati iskanja je prav tako pomemben podatek, kako priljubljen posamezen projekt je.

- **Mesto v rezultatu iskanja po ključnih besedah** (angl. Search Engine Rankings)

S spletnim iskalnikom Google bom preveril kako se posamezni sistemi po straneh uvrščajo med rezultate iskanja, če iščem po nekaj ključnih besedah in besednih zvezah, ki so povezane s sistemi za upravljanje spletnih vsebin. Uporabil bom besedno zvezo:

- *Content management system.*

- **Število nagrad**

Nagrade so ponavadi podeljene na podlagi glasovanja javnosti oziroma uporabnikov. To pomeni da nek projekt s tem, ko prejme nagrado, dobi neko potrditev priljubljenosti. Če pa pogledamo zgodovino nagrad pa lahko ugotovimo, kateri sistemi so bili mogoče nekdam priljubljeni, pa sedaj niso več ali obratno. Na spletni strani www.packtpub.com je objavljeno, kateri sistemi v zadnjih nekaj letih prejeli nagrade.

- **Rating na www.OpenSourceCMS.org**

Spletna stran OpenSourceCMS.com omogoča da uporabniki tudi glasujejo. Preveril bom ratinge posameznih sistemov.

6.3 Definicija funkcije koristnosti

V skladu z četrtem načelom metode MAUT, po katerem vrednotimo ocene od spodaj navzgor, moramo definirati dve vrsti funkcij koristnosti: osnovne in funkcije združevanja.

6.3.1 Osnovne funkcije koristnosti

Pri osnovnih funkcijah koristnosti gre za to, da pravo vrednost kriterija preslika v preferenčno. Za vsak kriterij bom z ocenami od 0 do 5 podal subjektivno njegovo oceno. Ničla predstavlja v splošnem slabo oceno, pet pa najboljšo možno.

Natančnejša razlaga pomena ocen:

- Ocena 0 - nek kriterij sistemu ni zastopan.
- Ocena 1 – kriterij je zastopan, vendar ne le komaj zadovoljuje pričakovanja, je ocena v bolj negativnem pomenu
- Ocena 2 – kriterij je zastopan, vendar le malo bolje zadovoljuje pričakovanja
- Ocena 3 – kriterij je zastopan, negativnih ali pozitivnih učinkov glede na pričakovanja ni mogoče določiti
- Ocena 4 – kriterij je zastopan, zadovoljstvo nad pričakovanji, ocena je priporočilo
- Ocena 5 – kriterij zastopan, zadovoljstvo zelo presega pričakovanja, zelo priporočljivo

6.3.2 Funkcije združevanja

Za funkcijo združevanja parametrov na višjih nivojih bom uporabil linearno funkcijo

$$\sum_{i=1}^n w_i * v_i$$

, kjer je

- n število osnovnih kriterijev ali kriterijev na nižjem nivoju,
- w_i normirana utež kriterija i na 100%
- V_i vrednost funkcije koristnosti za parameter i.

6.4 Opis variant

Variante bom opisal s pomočjo kriterijev, ki sem jih določil v predhodnih razdelkih. Za vsak izpeljani kriterij bom opise podal v ločenih tabelah. Vrednosti v tabelah so vrednosti osnovnih funkcij koristnosti.

6.4.1 Osnovna zmogljivost

CMS	Podpora verzijam	Podpora lokalizaciji za vsako stran	Podpora lokalizaciji za vsak vsebinski objekt	Časovno odvisno objavlanje	Podpora delovnemu toku	Navigacijski elementi (meniji)	Metapodatki	Sočasno urejanje vsebin
Drupal	4	4	3	3	2	5	5	2
Joomla	0	3	3	3	3	4	4	0
Plone	5	5	3	3	3	3	3	4
Wordpress	2	2	3	4	0	3	3	0
Silverstirpe	2	0	3	0	2	2	2	0
Sharepoint	2	0	3	4	2	3	3	3
DotNetNuke	4	3	3	4	2	4	3	0
dotCMS	5	3	3	3	4	5	3	4
Xoops	3	3	3	3	2	4	3	0
Typo3	5	5	3	4	5	4	5	5
CMS	Iskalnik vsebine	Gostovanje več strani	Prispevki uporabnikov	Upravljanje s povezavami	Upravljanje z datotekami	Upravljanje s kontakti	Upravljanje z dokumenti	Naročanje
Drupal	4	2	4	5	3	3	3	3
Joomla	4	0	3	4	2	4	4	2
Plone	4	4	3	3	3	4	3	2
Wordpress	3	0	3	3	3	2	0	3
Silverstirpe	3	0	2	1	3	0	0	0
Sharepoint	3	5	3	4	4	4	5	3
DotNetNuke	3	5	3	3	3	3	3	3
dotCMS	3	3	3	3	3	4	4	3
Xoops	3	0	3	3	2	2	2	3
Typo3	5	5	4	5	5	3	3	3

Tabela 10: Ocene osnovne zmogljivosti

6.4.2 Razširljivost z dodatnimi funkcionalnostmi

CMS	Preglednost obstoječih modulov	Število modulov	Forum	Blog	Galerija slik	Pogosto zastavljena vprašanja	Koledar dogodkov	Časovno sledenje uporabnikov	Ankete	Ankete z vprašalnikom	Zaslonska maska za elektronsko pošto	Zaslonska maska za vnos podatkov
Drupal	3	5	5	4	3	4	3	2	4	3	3	3
Joomla	3	5	2	2	3	4	3	2	4	3	3	1
Plone	3	5	2	5	4	1	0	2	4	3	2	0
Wordpress	3	3	2	5	3	1	2	0	4	2	2	0
Silverstirpe	3	4	1	1	2	2	2	0	0	0	0	5
Sharepoint	1	2	0	0	2	3	5	0	4	3	2	0
DotNetNuke	4	5	1	2	0	4	1	3	4	5	3	1
dotCMS	4	5	4	0	0	4	4	4	2	3	3	3
Xoops	1	4	1	4	5	3	2	4	5	0	3	4
Typo3	4	3	3	3	3	3	3	4	5	3	4	4

Tabela 11: Ocene dodatnih funkcionalnosti 1

CMS	Poročilo o podatkovni bazi	Poročilo o stroških	Oglaševanje (angl. Classifieds)	Knjiga gostov	Vodenje projektov	Podpora skupinskemu delu	Kvizi	Klepetalnica	Objava delovnih mest	HTTP Proxy	Prikaz prisotnosti	Grafikoni
Drupal	3	3	3	5	3	3	3	2	3	3	0	3
Joomla	2	2	4	3	2	4	2	2	1	3	0	2
Plone	0	3	2	4	4	1	2	5	5	3	2	2
Wordpress	0	0	2	0	0	0	2	2	0	3	3	2
Silverstirpe	0	0	0	0	0	0	0	0	0	3	2	2
Sharepoint	1	2	2	3	0	3	2	1	3	3	2	3
DotNetNuke	1	0	0	4	3	5	4	2	5	3	2	4
dotCMS	1	4	4	4	1	0	4	2	1	3	4	4
Xoops	4	4	4	4	1	0	3	3	1	3	3	5
Typo3	4	3	3	3	3	3	5	3	3	3	3	3

Tabela 12: : Ocene dodatnih funkcionalnosti 2

6.4.3 Enostavnost uporabe za uporabnika

	Urejevalnik WYSIWYG	»Nazaj« (UNDO)	Povleci in spusti	Črkovalnik	Nalaganje in spreminjanje velikosti slik	Vstavljanje slik v besedilo	Paketno nalaganje vsebine	Nalaganje vsebine v stisnjeni obliki (ZIP)	Uporabniški vmesnik za ustvarjanje in	Nadzorna plošča	Pozabljeno geslo
Drupal	0	1	0	1	3	2	3	3	2	1	3
Joomla	2	1	2	4	1	0	4	3	3	4	2
Plone	3	2	3	2	4	4	3	2	1	2	5
Wordpress	3	3	0	2	5	1	1	0	5	4	2
Silverstirpe	4	4	4	2	3	1	3	1	4	4	0
Sharepoint	3	4	3	4	3	2	2	5	0	3	4
DotNetNuke	3	4	1	1	2	1	2	3	4	2	3
dotCMS	3	1	2	1	4	0	2	3	4	0	3
Xoops	0	3	3	3	4	1	3	3	0	0	5
Typo3	0	1	5	1	0	5	2	4	2	4	2

Tabela 13: : Ocene enostavnosti uporabe za uporabnika

6.4.4 Enostavnost uporabe za administratorja

CMS	Administratorski vmesnik	Vmesnik za enostavno nadgrajevanje sistema	Enostavno dodajanje novih funkcionalnosti	Statistika spletne strani	Lokalizacija administrativnega vmesnika	Upravljanje z datumi ustvarjanja in spreminjanja vsebine
Drupal	4	0	5	0	4	1
Joomla	2	5	2	4	1	1
Plone	2	4	4	3	2	4
Wordpress	0	1	0	2	1	1
Silverstirpe	2	3	4	2	3	1
Sharepoint	1	2	1	4	2	3
DotNetNuke	3	0	0	2	2	4
dotCMS	4	1	4	1	3	3
Xoops	3	4	1	4	1	2

Typo3	5	2	1	0	4	4
	1	4	1	1	1	2

Tabela 14: Ocene enostavnosti uporabe za administratorja

6.4.5 Oblikovanje izgleda

CMS	Število osnovnih tem	Ugnezdene teme	Poljubno določanje položaja vsebine	Vmesnik za dodajanje nove teme	Vmesnik za oblikovanje teme	Struktura datotek	Posebni jezik
Drupal	3	3	3	3	0	3	1
Joomla	4	3	2	3	2	3	1
Plone	2	3	4	3	3	3	1
Wordpress	3	3	4	3	3	3	1
Silverstirpe	1	3	3	3	1	3	1
Sharepoint	1	3	1	3	1	3	1
DotNetNuke	3	3	3	3	3	3	1
dotCMS	2	3	2	3	3	3	1
Xoops	3	3	3	3	3	3	1
Typo3	5	3	4	3	5	3	5

Tabela 15: Ocene parametrov za oblikovanje izgleda

6.4.6 Interoperabilnost

	WebDAV	Podpora FTP	Ponujanje vsebine	Podpora XHTML	Podpora uporabi spletnih storitev	Prikazovanje ponujene vsebine (RSS)
Drupal	0	3	3	3	3	3
Joomla	3	3	3	3	4	3
Plone	4	3	3	3	3	3
Wordpress	5	2	3	3	3	3
Silverstirpe	4	2	3	3	3	3
Sharepoint	2	2	3	3	3	3
DotNetNuke	1	3	3	3	3	3
dotCMS	1	3	3	3	3	3
Xoops	1	3	3	3	4	3
Typo3	3	3	3	5	4	3

Tabela 16: Ocene parametrov za ocenjevanje interoperabilnosti

6.4.7 Varnost

	Nadzorna sled	Peskovnik	Zgodovina prijav	Upravljanje z sejami	Varne povezave(SSL in TSL)	Preverjanje naslova elektronske pošte	Captcha	Dnevnik	Varnostna kopija	Kontrola dostopa na nivoju skupin	Kontrola dostopa z vlogami	Podrobno določanje dostopa
Drupal	3	3	0	4	4	0	3	3	3	5	2	3
Joomla	3	3	3	0	1	3	4	4	1	4	1	1
Plone	3	3	3	1	2	3	1	5	1	2	4	4
Wordpress	5	4	0	3	5	2	3	0	2	4	4	3
Silverstirpe	1	1	3	1	4	2	2	4	2	1	5	4
Sharepoint	1	1	1	3	3	1	2	3	2	0	3	3
DotNetNuke	0	3	0	4	1	2	2	2	3	1	2	1
dotCMS	4	2	3	0	5	3	5	1	1	3	1	3
Xoops	4	4	2	2	2	4	3	0	1	1	2	4
Typo3	3	3	3	5	5	5	5	3	3	3	3	5

Tabela 17: Ocene parametrov za ocenjevanje varnosti

6.4.8 Skalabilnost

CMS	Predpomnenje	Napredno predpomnenje	Statična vsebina	izenačevanje obremenitve	Strežba zahtev
Drupal	3	4	0	3	2
Joomla	3	3		3	4
Plone	3	3	2	2	1
Wordpress	3	2	0	0	4
Silverstirpe	3	3	0	0	1
Sharepoint	4	4	4	4	5
DotNetNuke	3	3	3	3	5
dotCMS	3	3	3	3	2
Xoops	3	0	0	3	3
Typo3	5	5	3	4	1

Tabela 18: Ocene parametrov za kriterije skalabilnosti

CMS	1.meritev (z/s)	2.meritev (z/s)	3.meritev (z/s)	Povprečje (z/s)
Sharepoint	2399,55	2470,84	2444,60	2438,33
DotNetNuke	1447,19	1501,60	1443,01	1463,93
Joomla	1406,21	1402,26	1367,69	1392,05
Wordpress	1267,02	1280,02	1275,11	1274,05
Xoops	873,63	844,09	850,56	856,09
dotCMS	731,55	611,49	687,71	676,92
Drupal	545,28	543,62	526,03	538,31
Silverstirpe	200,40	205,18	205,86	203,81
Typo3	30,78	35,86	33,45	33,36
Plone	6,92	5,12	5,09	5,71

Tabela 19: Rezultati meritev zmogljivosti sistema z programom ab

6.4.9 Popularnost

CMS	število prenosov	Stopnja podpore tretje strani	število napisanih publikacij	število vhodnih povezav	mesto v rezultatu iskanja po ključ. besedah	število nagrad ²	rating ³
Drupal	0	3	4	4	3	4	2
Joomla	4	5	3	3	4	4	4
Plone	0	5	4	3	0	3	4
Wordpress	5	4	5	5	0	1	5
Silverstirpe	0	4	1	1	0	2	3
Sharepoint	0	2	5	5	0	n.p. ⁴	n.p.
DotNetNuke	0	2	3	2	2	2	n.p.
dotCMS	0	3	1	1	1	2	n.p.
Xoops	3	1	2	4	0	0	2
Typo3	0	1	2	2	5	0	1

Tabela 20: Ocene parametrov za kriterije popularnosti

² Vir podatkov o nagradah je spletna stran www.packtpub.com, ki ocenjuje samo odprtokodne sisteme

³ Podatki za rating so na voljo le za odprtokodne sisteme napisane v PHP jeziku

⁴ n.p. pomeni ni podatka

CMS	število prenosov	Vri
Plone	ni podatka	ni podatka
Drupal	ni podatka	ni podatka
Typo3	ni podatka	ni podatka
DotNetNuke	ni podatka	ni podatka
Silverstripe	ni podatka	ni podatka
SharePoint	ni podatka	ni podatka
dotCMS	ni podatka	ni podatka
Wordpress	4.543.436	wordpress.org
Joomla	218.306	joomla.org
Xoops	25.000	SourceForge.net

Tabela 21: Podatki o prenosih sistemov

CMS	Število razvijalcev sistema ⁵
Plone	2,463
Joomla	1,595
Wordpress	389
Silverstripe	203
dotCMS	69
Drupal	34
Xoops	34
DotNetNuke	10
SharePoint	2
Typo3	2

Tabela 22: Podatki za oceno stopnje podpore tretje strani

CMS	število napisanih publikacij ⁶
SharePoint	1.472

Wordpress	626
Drupal	221
Plone	203
Joomla	141
DotNetNuke	70
Xoops	56
Typo3	55
Silverstripe	1
dotCMS	0

Tabela 23: Podatki o številu napisanih publikacij

CMS	Št. strani v rezultatih iskanja po besedni zvezi ⁷
Typo3	1
Joomla	3
Drupal	7
DotNetNuke	14
dotCMS	18
Plone	>20
Wordpress	>20
Silverstripe	>20
SharePoint	>20
Xoops	>20

Tabela 24: Podatki za oceno mesta nahajanja v rezultatu iskanja po ključni besedah

⁵ Vir podatkov: www.elance.com (april 2009)⁶ Vir podatkov: spletna knjigarna amazon.com (april 2009)⁷ Uporabljen je bil spletni iskalnik Google

6.5 Vrednotenje in analiza variant

Odločitveni model naj bi uporabnikom pomagal pri izbiri CMS sistema tako, da izbere sistem, ki najbolj ustreza njegovim željam in potrebam.

6.5.1 Opis delovanja odločitvenega modela

Odločitveni model je bil razvit v programu Excel. Razlog za to je v široki dostopnosti in enostavnosti uporabe.

Delček modela prikazuje Slika 34. Parametri so v dveh nivojih strukturirani navpično, variante pa vodoravno.

Uporabnik ima na voljo prej določene in ovrednotene kriterije, svoje potrebe pa opiše z ustrezno določitvijo uteži. Uteži lahko izbira med 0 in 100. Rezultat se izračunava sproti, za vsako varianto. Tista varianta, ki ima največ skupnih točk je za določen profil uporabnika najprimernejša.

	normirane uteži na 1. nivoju	utež	normirane uteži na 2. nivoju	utež	Joomla		Plone	
					ocena*utež	ocena	ocena*utež	ocena
Končna ocena	100,00					29199,00		30644,83
Osnovna zmogljivost	2,17	10	100,42		282,50	614,13	357,50	777,17
Podpora verzijam			6,25	50	12,50	2	31,25	5
Podpora lokalizaciji za vsako stran			6,25	50	18,75	3	31,25	5
Podpora lokalizaciji za vsak vsebinski objekt			6,25	50	18,75	3	18,75	3
Časovno odvisno objavlanje			6,67	50	20,00	3	20,00	3
Podpora delovnemu toku			6,25	50	18,75	3	18,75	3
Navigacijski elementi (meniji)			6,25	50	25,00	4	18,75	3
Metapodatki			6,25	50	25,00	4	18,75	3
Sočasno urejanje vsebin			6,25	50	0,00	0	25,00	4
Iskalnik vsebine			6,25	50	25,00	4	25,00	4
Gostovanje več strani			6,25	50	0,00	0	25,00	4
Prispevki uporabnikov			6,25	50	18,75	3	25,00	4
Upravljanje s povezavami			6,25	50	25,00	4	12,50	2
Upravljanje z datotekami			6,25	50	12,50	2	18,75	3
Upravljanje s kontakti			6,25	50	25,00	4	25,00	4
Upravljanje z dokumenti			6,25	50	25,00	4	31,25	5
Naročanje			6,25	50	12,50	2	12,50	2
Podpora za PDA in mobilne naprave			6,25	50	18,75	3	25,00	4
Razširljivost	13,04	60	100,00		264,29	3447,20	257,14	3354,04
Enostavnost za uporabnika	8,70	40	100,00		209,09	1818,18	254,55	2213,44
Enostavnost za administratc	1,09	5	100,00		316,67	4130,43	216,67	2826,09

Slika 34: Primer definiranja funkcije koristnosti

V nadaljevanju bom definiriral nekaj tipičnih profilov uporabnikov CMS sistemov. V odločitvenim modelu bom za vsakega določil njemu primerno funkcijo koristnosti. Varianta oziroma sistem, ki bo prejel najvišjo oceno, bo za določen profil uporabnika najbolj primeren.

Definirani profili bodo služili samo za prikaz delovanja odločitvenega sistema, tako da posameznim uporabnikom odločitvenega modela seveda dopuščam spremembe.

6.5.2 Profili uporabnikov

6.5.2.1 Individualni portal

V tabeli 10 je s pomočjo uteži definiran profil uporabnika »individualni portal«. Uteži na nižjih nivojih so bile v modelu enakomerno razporejene definirane.

	Normirana utež	utež
Osnovna zmogljivost	18,18	70
Razširljivost	12,99	50
Enostavnost za uporabnika	20,78	80
Enostavnost za administratorja	23,38	90
Oblikovanje izgleda	12,99	50
Interoperabilnost	3,90	15
Varnost	5,19	20
Skalabilnost	1,30	5
Popularnost sistema	1,30	5

Tabela 25: Definicija profila "individualni portal"

Odločitveni model ja za najprimernejše sisteme izbral:

- DotNetNuke
- Drupal
- WordPress
- dotCMS

6.5.2.2 Portal za manjše skupine ljudi in društva

V tabeli 11 je s pomočjo uteži definiran profil uporabnika »individualni portal«. Uteži na nižjih nivojih so bile v modelu enakomerno razporejene definirane.

	Normirana utež	utež
Osnovna zmogljivost	10,87	50
Razširljivost	13,04	60
Enostavnost za uporabnika	15,22	70
Enostavnost za administratorja	6,52	30
Oblikovanje izgleda	15,22	70
Interoperabilnost	15,22	70
Varnost	13,04	60
Skalabilnost	6,52	30
Popularnost sistema	4,35	20

Tabela 26: Definicija profila za "Portal za manjše skupine ljudi in društva"

Odločitveni model ja za najprimernejše sisteme izbral:

- DotNetNuke
- Drupal
- dotCMS

6.5.2.3 Podjetje in večje organizacije

V tabeli 12 je s pomočjo uteži definiran profil uporabnika »Podjetje in večje organizacije«.

Uteži na nižjih nivojih so bile v modelu enakomerno razporejene definirane.

	Normirana utež	utež
Osnovna zmogljivost	2,17	10
Razširljivost	13,04	60
Enostavnost za uporabnika	8,70	40
Enostavnost za administratorja	1,09	5
Oblikovanje izgleda	15,22	70
Interoperabilnost	15,22	70
Varnost	21,74	100
Skalabilnost	21,74	100
Popularnost sistema	1,09	5

Tabela 27: Definicija profila "Podjetje in večje organizacije"

Odločitveni model ja za najprimernejše sisteme izbral:

- Typo3
- Drupal
- dotCMS

6.6 Splošne ugotovitve

Odločitveni model je za profil »individualnega uporabnika« predlagal DotNetNuke, Drupal, WordPress in dotCMS. Pomembne lastnosti za ta profil so predvsem enostavnost za uporabo, tako za administratorja, kot tudi za navadnega uporabnika. Po pomembnosti sledi osnovna zmogljivost in možnost razširitve. Varnost ni tako pomembna, saj navadno take spletne strani ne vsebujejo zelo pomembnih podatkov. Skalabilnost tudi ni ključnega pomena, saj se ne pričakuje velikega prometa.

Odločitveni model je za profil »portal za manjše skupine ljudi in društva« predlagal Drupal, DotNetNuke in dotCMS. Za ta profil ni izrazito pomembnih lastnosti. Nekoliko bolj pomembno je, da je portal enostaven za uporabo predvsem za uporabnike, saj se predvideva, da ima neka manjša skupina ljudi ali društvo posebnega administratorja. Za razliko od prejšnjega profila je pri tem varnost pomembnejša, saj je zaradi večjega števila potrebno urediti pravice dostopa. Zaradi pričakovanega nekoliko večjega obiska je priporočljiva tudi nekoliko boljša skalabilnost.

Odločitveni model je za profil »podjetje in večje organizacije« predlagal sisteme Typo3, Drupal in dotCMS. Varnost in skalabilnost sta za ta profil najpomembnejša. Predvideva se namreč da s portalom oziroma s podatki upravlja veliko število ljudi z različnimi odgovornostmi. Velika količina dnevnega prometa zahteva visoko skalabilnost. Enostavnost uporabe ni tako pomembna, saj se predvideva da portal uporabljajo tehnično bolj izobraženi uporabniki. Oblikovanje izgleda ni zanemarljiva lastnost, saj je za podjetja lepo in kvalitetno oblikovana zunanja podoba tudi pomembna stvar.

Opaziti je da pri vseh profilih izstopajo sistemi, ki dajejo vtis, da so »NET« orientirani. Razlog je v tem, da je uporabnik, ki je subjektivno podajal ocene za variante, ter kasneje tudi določal profile, tipičen »Windows uporabnik«, ki so mu blizu koncepti omenjenega operacijskega sistema.

7 Zaključek

Izdelani odločitveni omogoča uporabnikom, da s poljubnim določanjem uteži posameznim lastnostim na višjem nivoju in parametrom na nižjem nivoju izberejo CMS sistem, ki kar najbolj ustreza njegovim potrebam.

Definiranje posameznih parametrov, ki določajo posamezno lastnost, se je izkazalo za težavno, saj v svetu CMS sistemov ni še definiranih kakšnih standardov, ki bi natančno določevali katere funkcionalnosti mora v osnovi nek CMS sistem zagotavljati.

Izkazalo se je, da zasnovani odločitveni model, na podlagi trenutnih ocen variant, sprejema precej podobne odločitve, ne glede na različne profile uporabnikov. Razlog je predvsem v ocenah variant saj so precej subjektivne in namenjene predvsem prikazovanju delovanja odločitvenega modela, ki je nastal na podlagi uporabi teorije odločanja pri izbiri CMS sistemov. Glede subjektivnost velja poudariti to, da so bile variante ocenjene predvsem z vidika uporabnika, ki je tipičen »Windows uporabnik«.

Predvidevanja so bila, da bi bili sistemi WordPress, Joomla, Drupal primerni za individualne uporabnike, za manjše skupine ljudi ali društva bi bili namenjeni Plone, Drupal, Joomla, Xoops, DotNetNuke, za podjetja in večje organizacije pa bi bili najbolj ustrezni SharePoint Server, dotCMS in Typo3.

Možica parametrov za ocenjevanje je precej velika, zato bi bilo smiselno razmisliti o postavitvi nekakšnega spletišča, kjer bi lahko določene skupine strokovnjakov na podlagi definirane sistema ocenjevanja, podajali ocene posameznim parametrom. Na tak način bi bila zagotovljena tudi ažurnost ocen, saj se sistemi vseskozi razvijajo in izboljšujejo.

Možne so seveda spremembe in nadaljnji razvoj modela, predvsem v smislu dodajanja in spreminjanja odločitvenih kriterijev ter dodajanja novih variant in spreminjanje obstoječih ocen.

Dodatki

Priložena je slika odločitvenega modela za izbiro CMS sistema.

	normirane uštede na 1. nivoju	ušte	normirane uštede na 2. nivoju	ušte	Joomla		Ploone		Drupal		Typo3		DotNetNuke		Wordpress		SilverStripe		Jare Point Server 20		Xoops		dotCMS		
					kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena	kena*ute ocena
Končna ocena	100,00				28953,35	53589,41	83500,56	143991,79	89436,22	138488,44	20636,58	32233,50	80636,20	28158,51											
Osnovna zmogljivost	4,72	20	100,00		252,94	1193,12	329,41	1553,83	323,53	1526,08	405,88	1914,54	288,24	1359,60	200,00	943,40	117,65	554,94	288,24	1359,60	211,76	998,89	302,00	1424,53	
Podpora verzijam	5,88	50	0,00	0	0,00	0,00	29,41	23,53	4	29,41	5	23,53	4	11,76	2	0	2	0	2	0	3	2	0	5	
Podpora lokalizacij za vsako stran	5,88	50	17,65	3	17,65	3	29,41	5	23,53	4	29,41	5	17,65	3	11,76	2	0,00	0	0,00	0	17,65	3	17,65	3	
Podpore lokalizacij za vsak vsebinski objekt	5,88	50	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65
Časovno odvisno objavljanje	5,88	50	17,65	3	17,65	3	17,65	3	23,53	4	23,53	4	23,53	4	23,53	4	0,00	0	23,53	4	17,65	3	17,65	3	17,65
Podpora delovnemu toku	5,88	50	17,65	3	17,65	3	11,76	2	29,41	5	11,76	2	0,00	0	11,76	2	11,76	2	11,76	2	11,76	2	23,53	4	
Navigacijski elementi (meni/ii)	5,88	50	23,53	4	17,65	3	29,41	5	23,53	4	23,53	4	23,53	4	17,65	3	11,76	2	17,65	3	23,53	4	29,41	5	
Metapodatki	5,88	50	23,53	4	17,65	3	29,41	5	29,41	5	17,65	3	17,65	3	17,65	3	11,76	2	17,65	3	17,65	3	17,65	3	
Sočasno urejanje vsebin	5,88	50	0,00	0	23,53	4	11,76	2	29,41	5	0,00	0	0,00	0	0,00	0	17,65	3	17,65	3	0,00	0	23,53	4	
Iskalnik vsebine	5,88	50	23,53	4	23,53	4	23,53	4	29,41	5	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65
Gostovanje več strani	5,88	50	0,00	0	23,53	4	11,76	2	29,41	5	29,41	5	29,41	5	0,00	0	0,00	0	29,41	5	0,00	0	17,65	3	17,65
Prispevki uporabnikov	5,88	50	17,65	3	17,65	3	23,53	4	23,53	4	17,65	3	17,65	3	17,65	3	11,76	2	17,65	3	17,65	3	17,65	3	17,65
Upravljanje s povezavami	5,88	50	23,53	4	17,65	3	29,41	5	29,41	5	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65
Upravljanje z datotekami	5,88	50	11,76	2	17,65	3	17,65	3	29,41	5	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65
Upravljanje s kontakti	5,88	50	23,53	4	23,53	4	23,53	4	17,65	3	17,65	3	17,65	3	11,76	2	0,00	0	23,53	4	11,76	2	23,53	4	23,53
Upravljanje z dokumenti	5,88	50	23,53	4	23,53	4	17,65	3	17,65	3	17,65	3	17,65	3	0,00	0	0,00	0	29,41	5	11,76	2	23,53	4	23,53
Naročanje	5,88	50	11,76	2	11,76	2	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65	3	17,65
Podpora za PDA in mobilne naprave	5,88	50	17,65	3	23,53	4	17,65	3	29,41	5	17,65	3	17,65	3	17,65	3	0,00	0	17,65	3	11,76	2	23,53	4	23,53
Razširitev	11,56	49	100,00		287,86	3995,52	271,43	3136,79	317,86	3673,35	335,71	3879,72	282,14	3260,61	185,71	2146,23	125,00	1444,58	207,14	2393,87	303,57	3508,25	289,29	3343,18	
Preglednost obstoječih modulov	3,57	50	10,71	3	10,71	3	10,71	3	14,29	4	14,29	4	10,71	3	10,71	3	10,71	3	3,57	1	3,57	1	14,29	4	14,29
Število modulov	3,57	50	17,65	3	17,65	3	17,65	3	10,71	3	17,65	3	10,71	3	10,71	3	14,29	4	7,14	2	14,29	4	17,65	3	17,65
Blizum	3,57	50	7,14	2	7,14	2	17,65	3	10,71	3	3,57	1	7,14	2	3,57	1	3,57	1	0,00	0	3,57	1	14,29	4	14,29
Blizum	3,57	50	7,14	2	17,65	3	14,29	4	10,71	3	7,14	2	17,65	3	3,57	1	3,57	1	0,00	0	14,29	4	0,00	0	0,00
Galerija slik	3,57	50	10,71	3	14,29	4	10,71	3	0,00	0	0,00	0	10,71	3	7,14	2	7,14	2	17,65	3	0,00	0	0,00	0	0,00
Pogosto zastavljena vprašanja	3,57	50	14,29	4	3,57	1	14,29	4	10,71	3	14,29	4	3,57	1	7,14	2	10,71	3	10,71	3	10,71	3	14,29	4	14,29
Kalendar dogodkov	3,57	50	10,71	3	0,00	0	10,71	3	10,71	3	3,57	1	7,14	2	17,65	3	7,14	2	17,65	3	7,14	2	14,29	4	14,29
Časovno sledenje uporabnikov	3,57	50	7,14	2	7,14	2	7,14	2	14,29	4	10,71	3	0,00	0	0,00	0	0,00	0	0,00	0	14,29	4	14,29	4	14,29
Ankete	3,57	50	14,29	4	14,29	4	14,29	4	17,65	3	14,29	4	14,29	4	14,29	4	14,29	4	0,00	0	17,65	3	7,14	2	7,14
Ankete z vprašanjskimi	3,57	50	10,71	3	10,71	3	10,71	3	10,71	3	17,65	3	7,14	2	0,00	0	10,71	3	0,00	0	0,00	0	10,71	3	10,71
Zaslonka maska za elektronsko pošto	3,57	50	10,71	3	7,14	2	10,71	3	14,29	4	10,71	3	7,14	2	0,00	0	7,14	2	10,71	3	10,71	3	10,71	3	10,71
Zaslonka maska za vnos podatkov	3,57	50	3,57	1	0,00	0	10,71	3	14,29	4	3,57	1	0,00	0	17,65	3	0,00	0	14,29	4	10,71	3	10,71	3	10,71
Poročilo o podatkovni bazi	3,57	50	7,14	2	0,00	0	10,71	3	14,29	4	3,57	1	0,00	0	0,00	0	3,57	1	14,29	4	14,29	4	3,57	1	3,57
Poročilo o stroških	3,57	50	7,14	2	10,71	3	10,71	3	10,71	3	0,00	0	0,00	0	0,00	0	7,14	2	14,29	4	14,29	4	14,29	4	14,29
Oglaševanje (angl. Classifieds)	3,57	50	14,29	4	7,14	2	10,71	3	10,71	3	0,00	0	7,14	2	0,00	0	7,14	2	14,29	4	14,29	4	14,29	4	14,29
Knjiga gostov	3,57	50	10,71	3	14,29	4	17,65	3	10,71	3	14,29	4	10,71	3	0,00	0	10,71	3	14,29	4	14,29	4	14,29	4	14,29
Vodenje projektov	3,57	50	7,14	2	14,29	4	10,71	3	10,71	3	0,00	0	0,00	0	0,00	0	0,00	0	3,57	1	3,57	1	14,29	4	14,29
Podpora skupinskemu delu	3,57	50	14,29	4	3,57	1	10,71	3	10,71	3	17,65	3	17,65	3	0,00	0	10,71	3	0,00	0	0,00	0	0,00	0	0,00
Kvizi	3,57	50	7,14	2	7,14	2	10,71	3	17,65	3	14,29	4	14,29	4	14,29	4	14,29	4	14,29	4	10,71	3	14,29	4	14,29
Klepatelnica	3,57	50	7,14	2	17,65	3	7,14	2	10,71	3	7,14	2	7,14	2	0,00	0	3,57	1	10,71	3	10,71	3	7,14	2	7,14
Objava delovnih mest	3,57	50	3,57	1	17,65	3	10,71	3	17,65	3	17,65	3	17,65	3	0,00	0	10,71	3	3,57	1	3,57	1	14,29	4	14,29
HTTP Proxy	3,57	50	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71
Prikaz prisotnosti	3,57	50	0,00	0	14,29	4	10,71	3	10,71	3	7,14	2	10,71	3	7,14	2	7,14	2	10,71	3	17,65	3	14,29	4	14,29
Grifikoni	3,57	50	7,14	2	7,14	2	10,71	3	10,71	3	14,29	4	7,14	2	7,14	2	10,71	3	17,65	3	14,29	4	14,29	4	14,29
Wiki	3,57	50	10,71	3	7,14	2	10,71	3	10,71	3	10,71	3	3,57	1	3,57	1	3,57	1	14,29	4	14,29	4	14,29	4	14,29
Zemljevid strani	3,57	50	14,29	4	14,29	4	14,29	4	14,29	4	14,29	4	14,29	4	14,29	4	14,29	4	14,29	4	14,29	4	14,29	4	14,29
Pregled vrednosti borznih indeksov	3,57	50	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71
Vreme	3,57	50	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71	3	10,71
Samostojnost za uporabnika	7,08	30	100,00		1157,80	172,70																			

Viri

1. Wikipedia. [Elektronski] www.wikipedia.org.
2. **Boiko, Bob.** *Content Management Bible, 2nd Edition*. Indiana : Wiley Publishing, Inc., 2005.
3. **Kampffmeyer, Ulrich.** Enterprise Content Management. *Project Consult*. [Elektronski] 2006. http://www.project-consult.net/Files/ECM_White%20Paper_kff_2006.pdf.
4. **Jon Kee.** Open Source CMS Vs Commercial CMS - It's On. *EzineArticles*. [Elektronski] EzineArticles.com. <http://ezinearticles.com/?Open-Source-CMS-Vs-Commercial-CMS---Its-On&id=2003706>.
5. **Sean Michael Kerner.** Choose between a commercial, open source, or customized CMS. *TechRepublic*. [Elektronski] CBS Interactive Inc. http://articles.techrepublic.com.com/5100-10878_11-5054863.html.
6. *The CMS Matrix*. [Elektronski] <http://www.cmsmatrix.org/>.
7. **Shreves, Ric.** *The 2008 Open Source CMS Market Share Report*. Kerobokan, Indonesia : water&stone, 2008.
8. **Boiko, Bob.** *Top Level CMS Evaluation Criteria, A CM Domain White Paper*. s.l. : Metatorial Services Inc. & HungryMinds Inc., 2002.
9. **Cummings, David.** Push Vs. Pull - The Battle for the Best CMS. *Sitepoint*. [Elektronski] 15. februar 2005. <http://www.sitepoint.com/article/push-pull-best-cms/>.
10. **Graf, Hagen.** *Building Websites with Joomla! 1.5 Beta 1*. Birmingham : Packt Publishing Ltd., 2007.
11. *Joomla!* [Elektronski] www.joomla.org.
12. *Plone*. [Elektronski] Plone Foundation. <http://www.plone.org/>.
13. **Thomas Lotze, Jan Ulrich Hasecke.** *A User's Guide to Plone*. Hamburg, Germany : gocept gmbh & co. kg, 2009.

14. **Mav, Jure.** *Multimedijska spletna skupnost o2z2 na odprtokodnem spletnem sistemu za upravljanje vsebin Drupal (diplomsko delo).* Ljubljana : Inštitut in akademija za multimedije, Višja šola za multimedije, 2008.
15. *Drupal.org.* [Elektronski] <http://drupal.org>.
16. **Mercer, David.** *Building powerful and robust websites with Drupal 6.* Birmingham, UK : Packt Publishing Ltd., 2008.
17. **Banjac, Vitomir.** *Povezovanje CMS in CRM sistemov v malih podjetjih (diplomsko delo).* [ured.] doc. dr. Mojca Ciglarič. Ljubljana : Univerza v Ljubljani, Fakulteta za računalništvo in informatika, 2007.
18. Documentation - Learning TYPO3. *TYPO3.* [Elektronski] <http://typo3.org/documentation/document-library/>.
19. **Peacock, Michael.** *Building Websites with TYPO3.* Birmingham, UK : Packt Publishing Ltd., 2007.
20. *DotNetNuke.* [Elektronski] <http://www.dotnetnuke.com>.
21. Easy DotNetNuke Installation Guide. *Best Web Sites.* [Elektronski] <http://www.bestwebsites.co.nz/dotnetnuke/easy-dotnetnuke-installation-guide/>.
22. **Egan, Daniel N., Washington, Michael A. in Valenzula, Steve.** *Building Websites with VB.NET and DotNetNuke 4.* Birmingham,UK : Packt Publishing, 2006.
23. **Ličen, Peter.** *Odprtokodni spletni sistemi za urejanje vsebin (diplomsko delo).* [ured.] doc. dr. Mojca Ciglarič. Ljubljana : Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, 2007.
24. *islovar.* [Elektronski] Slovensko društvo Informatika. www.islovar.org.
25. Codex. *wordpress.org.* [Elektronski] <http://codex.wordpress.org>.
26. SilverStripe. *Wikipedia.* [Elektronski] <http://en.wikipedia.org/wiki/SilverStripe>.
27. *SilverStripe Documentation.* [Elektronski] <http://doc.silverstripe.com>.

28. **Jamison, Scott, Cardarelli, Mauro in Hanley, Susan.** *ESSENTIAL SHAREPOINT2007*. s.l. : Addison-Wesley, 2007.
29. **Williams, Vanessa L.** *Microsoft SharePoint 2007 for Dummies* . Indianapolis : Wiley Publishing, Inc., 2007.
30. **Atwal, Steve.** *Building Websites with XOOPS*. Birmingham : Packt Publishing Ltd., 2006.
31. A Visual Introduction to XOOPS. *Scribd*. [Elektronski] <http://www.scribd.com/doc/7301152/A-Visual-Introduction-to-XOOPS>.
32. *dotCMS*. [Elektronski] dotCMS, Inc. www.dotcms.org.
33. dotCMS Administrator Manual. *dotCMS*. [Elektronski] 2007. http://marketing.dotmarketing.com/community/documentation/documentation_repository/manuals/dotCMS_1.2_Admin_Manual.pdf.
34. **Bohanec, Marko.** *Odločanje in modeli*. Ljubljana : DMFA - ZALOŽINIŠTVO, 2006. ISBN 961-212-190-7.
35. **Rajkovič, prof. dr. Vladislav.** *Odločitveni sistemi*. Ljubljana : s.n.
36. **Smith, Elliot.** A review of open source content management systems. <http://www.openadvantage.org>. [Elektronski] maj 2005. [Navedeno: 5. marec 2009.] <http://www.openadvantage.org/articles/oadocument.2005-04-19.0329097790/view>.
37. **Robertson, James.** CMS interoperability? [Elektronski] 4. april 2005. http://www.steptwo.com.au/papers/kmc_interoperability/index.html.
38. WebDAV. *Wikipedia*. [Elektronski] <http://en.wikipedia.org/wiki/WebDAV>.
39. XHTML Compliance. *LassoSoft*. [Elektronski] <http://www.lassosoft.com/Documentation/TotW/index.lasso?8991>.
40. XHTML2 Working Group Home Page. [Elektronski] W3C. <http://www.w3.org/MarkUp/>.
41. **Bartolj, Primož.** *Spletni CMS sistemi in distribucija spletnih (diplomsko delo)*. [ured.] prof. dr. Borut Robič. Ljubljana : Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, 2005.

42. **Sodnik, Jaka.** *Preprost protokol za dostop do imenika (diplomsko delo univerzitetnega študija).* [ured.] prof. dr. Sašo Tomažič. Ljubljana : Univerza v Ljubljani, Fakulteta za elektrotehniko, 2002.

43. **Freund, Jeff.** Interface Scalability. *www.cmswatch.com.* [Elektronski] 16. julij 2003.
[http://www.cmswatch.com/Feature/92-Scalability.](http://www.cmswatch.com/Feature/92-Scalability)