

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RA UNALNIŠTVO IN INFORMATIKA

Andrej Štrajhar

**Uporabniško prijazen sistem za upravljanje s  
spletnimi vsebinami za skavte**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE STOPNJE  
RA UNALNIŠTVO IN INFORMATIKA

MENTOR: dr. Marko Robnik-Šikonja

Ljubljana, 2012



Št. naloge: 00141/2011

Datum: 01.09.2011

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

Kandidat: **ANDREJ ŠTRAJHAR**

Naslov: **UPORABNIŠKO PRIJAZEN SISTEM ZA UPRAVLJANJE S SPLETNIMI  
VSEBINAMI ZA SKAVTE**

**A USER-FRIENDLY WEB CONTENT MANAGEMENT SYSTEM FOR  
SCOUTS**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:

Obstaja vrsta splošnonamenskih sistemov za upravljanje s spletnimi vsebinami, ki pokrivajo široko množico potreb različnih uporabnikov. V posameznih primerih pa so zaradi specifične ciljne populacije potrebne tudi specializirane rešitve. Analizirajte slabosti obstoječih rešitev pri objavljanju spletnih vsebin za skavtsko organizacijo in predlagajte ustrezno rešitev. Izdelajte prototip uporabnikom prijazne aplikacije in delovanje praktično ovrednotite.

Mentor:

  
prof. dr. Marko Robnik Šikonja

Dekan:

  
prof. dr. Nikolaj Zimic



# IZJAVA O AVTORSTVU

## diplomskega dela

Spodaj podpisani Andrej Štrajhar,

z vpisno številko 63060374,

sem avtor diplomskega dela z naslovom:

Uporabniško prijazen sistem za upravljanje s spletnimi vsebinami za skavte

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom dr. Marko Robnik-Šikonja
- 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 27.2.2012

Podpis avtorja:

## **Zahvala**

Za pomoč pri izdelavi diplomske naloge se zahvaljujem mentorju, dr. Marko Robnik-Šikonji in ekipi skavtskega spletnega servisa – zavod SkavtNET pri podpori in nasvetih med samim programiranjem.

# KAZALO

Povzetek .....	1
Abstract.....	2
1. UVOD .....	3
2. SISTEM ZA UPRAVLJANJE Z VSEBINAMI .....	5
2.1. Kaj omogoajo sistemi CMS? .....	6
2.2. Slabosti sistema CMS .....	7
2.3. Odprtokodni CMS sistemi .....	8
2.3.1. WordPress.....	9
2.3.2. Drupal .....	9
2.3.3. Joomla! .....	10
2.4. Primerjava CMS-sistemov .....	11
3. UPORABA CMS SISTEMA JOOMLA! V PRAKSI .....	12
3.1. Razlogi za izbiro Joomla, kot najprimernejši CMS sistem:.....	12
3.2. Spletna stran kamniških skavtov.....	12
4. OPIS Skavt AROVNIKA .....	16
4.1. Vidni del .....	16
4.2. Administrativni del .....	18
4.2.1. Dodajanje, urejanje, brisanje novic .....	19
4.2.2. Dodajanje, urejanje, brisanje strani .....	19
4.2.3. Dodajanje, urejanje, brisanje kategorij .....	19
4.2.4. Nalaganje, urejanje, brisanje slik in drugih datotek .....	19
4.2.5. Zaklepanje dostopa vseh vsebin .....	20
4.2.6. Urejanje strukture strani .....	21
4.2.7. Preprosto urejanje izgleda strani .....	22
4.2.8. Nalaganje lastnega CSS.....	23
4.2.9. Osnovne nastavitve strani.....	23
4.3. Zakaj smo se odločili ali razviti Skavt AROVNIKA?.....	24
4.4. Prednosti in slabosti Skavt AROVNIKA glede na druge CMS sisteme .....	24
4.5. Predstavitev uporabljene tehnologije pri Skavt AROVNIKU .....	24

4.5.1.	Jezik HTML .....	24
4.5.2.	Jezik JavaScript.....	25
4.5.3.	Jezik PHP .....	25
4.5.4.	Podatkovna baza MySQL .....	26
4.5.5.	CSS.....	27
5.	IMPLEMENTACIJA KOMENTARJEV PRI SKAVT AROVNIKU.....	28
5.1.	CodeIgniter.....	28
5.1.1.	MVC arhitektura .....	29
5.2.	Na rtovanje komentarjev .....	30
5.2.1.	Na rtovanje podatkovne baze .....	31
5.3.	Implementacija in opis kode za komentarje .....	32
5.3.1.	Preverjanje, e je uporabnik urednik spletne strani .....	32
5.3.2.	Uporabnik je prijavljen/ni prijavljen.....	33
5.3.3.	Preverjanje možnosti komentiranja novice .....	33
5.3.4.	Vstavljanje komentarjev v podatkovno bazo .....	34
5.3.5.	Izpis vseh komentarjev pod dolo eno novico .....	34
5.3.6.	Brisanje komentarjev .....	35
5.3.7.	Prijava neprimerne komentarja .....	36
6.	ZAKLJU EK .....	38
7.	VIRI.....	39

## Seznam slik:

Slika 3.1 Ospredje spletne strani kamniških skavtov [8].....	13
Slika 3.2 Nadzorna ploš a sistema Joomla! [9].....	14
Slika 4.1 Skavt AROVNIK [10]. ....	16
Slika 4.2 Vidni del Skavt AROVNIKA [11]. ....	17
Slika 4.3 Administrativni del Skavt AROVNIKA [12]. ....	18
Slika 4.4 Nalaganje datotek preko modula. ....	20
Slika 4.5 Urejanje in iskanje datotek – medijska knjižnica .....	20
Slika 4.6 Navigacija na spletni strani. ....	22
Slika 4.7 Urejanje izgleda strani Skavt AROVNIKA.....	23
Slika 5.1 Logotip programksega orodja CodeIgniter [19].....	28
Slika 5.2 MVC arhitektura [21].....	30
Slika 5.3 Relacijski podatkovni model pretvorjen iz konceptualnega modela.....	31
Slika 5.4 Tabela cms_news_comments v podatkovni bazi. ....	32
Slika 5.5 Primer poslane e-pošte uredniku za neprimeren komentar. ....	37

## Seznam preglednic:

Tabela 2.1 Primerjava odprtokodnih CMS sistemov. ....	11
-------------------------------------------------------	----

## Seznam uporabljenih kratic in simbolov:

CMS - sistem za upravljanje z vsebinami (angl. Content Management Systems),  
WYSIWYG – to kar vidiš, to dobiš (angl. What you see is what you get),  
HTML - ozna evalni jezik (angl. Hyper Text Markup Language),  
CSS - kaskadni stilski slog (angl. Cascadung Style Sheet),  
PDF – odprt standard za izmenjavo elektronskih dokumentov (angl. Portable Document Format),  
RAR – sistem za stiskanje datotek (angl. Roshal Archive),  
RSS – zares preprosto zlaganje (angl. Really Simple Syndication) ,  
PHP - orodja za osebno spletno stran (angl. Personal Home Page Tool),  
LAMP – Linux, Apache, MySQL, PHP,  
MySQL – moj strukturalni poizvedbeni jezik (angl. My Structured Query Language),  
URL - enoli ni naslov vira (angl. Uniform Resource Locator).  
MVC – model (M), pogled (V), krmilnik (C) (angl. Model, View, Controller)  
HTTP - protokol za prenašanje hiperteksta (angl. HyperText Transfer Protocol)  
ODBC - odprto povezovanje podatkovne baze (angl. Open DataBase Connectivity)



## **Povzetek**

Diplomska naloga obsega opis sistema za upravljanje s spletnimi vsebinami, njegove prednosti in slabosti. Opisal sem tri najbolj pogosto uporabljene CMS sisteme WordPress, Drupal in Joomla. Te sisteme sem primerjal med seboj. Izkazalo se je, da so najbolj obiskane spletne strani izdelane s pomojo CMS sistema WordPress.

Glavni del moje diplomske naloge je opis sistema za upravljanje z vsebinami Skavt AROVNIK, ki smo ga razvili v letu 2010 in 2011 pri ekipi SkavtNET. Podrobneje sem opisal funkcionalnosti, uporabljene tehnologije za razvoj aplikacije in implementacijo komentarjev. Implementacija komentarjev in prikaz komentarjev na spletni strani je bil moj prispevek pri razvoju Skavt AROVNIKA. Opisal sem tudi programsko orodje CodeIgniter, s pomojo katerega sem sprogramiral komentarje, ter na rtovanja podatkovne baze in delovanje funkcij, ki sem jih sprogramiral.

### **Ključne besede:**

Sistem za upravljanje z vsebinami, Skavt AROVNIK, spletna stran, implementacija komentarjev.

## Abstract

The work describes content management systems, their advantages and disadvantages. I described the three most commonly used systems: WordPress, Drupal and Joomla. These systems were compared. The main part of the work is the description of the content management system Skavt AROVNIK, which was developed in 2010 and 2011 by the SkavtNET team. I have described the functionalities and technologies used for application development and implementation of comments. Implementation of the comments is my contribution to the Skavt AROVNIK. I describe a software tool CodeIgniter, which was used for for implementation of comments. I also describe the database design and explain the functionality I have contributed.

### **Keywords:**

Content Management Systems, Skavt AROVNIK, website, implementation comments.

# 1. UVOD

V vedno bolj dinami nem, globalnem in kompleksnem okolju, ki nas obdaja, je dostop do aktualnih, ažurnih in preglednih informacij nujen in potreben. Zaradi ostre konkurence in boja za vsakega novega obiskovalca spletne strani, postajajo le-te vedno bolj tehnološko izpopolnjene in vsebinsko bogatejše. Uporabniki želijo pregledne in ažurne spletne informacije in prijazne spletne servise. Spletnim stranem podjetij, ki tega ne omogo ajo, se uporabniki izogibajo.

Potreba po vsakodnevnem objavljanju ali obnavljanju ve jega števila spletnih strani je problem, s katerim se soo ajo mnogi lastniki spletnih strani. as, ki pote e od nastanka informacij do njene objave in stroški, ki so s tem procesom povezani, lahko predstavljajo eno klju nih prednosti ali slabosti spletnega nastopa. e so postopki za objavo informacij na spletni strani preve zapleteni in tudi predragi (kar je še posebej pogost pojav pri kompleksnih in vsebinsko zelo obsežnih spletnih mestih), lahko privedejo do tega, da se vsebina na spletni strani preprosto ne obnavlja, oziroma se ne obnavlja v dovolj pogostih asovnih intervalih. Na daljši rok se kaže takšen odnos do vsebine v njeni vedno manjši aktualnosti, relevantnosti in verodostojnosti, kar pripelje do zmanjšanega obiska spletne strani. Za odpravo teh problemov obstaja na trgu ogromno aplikacij - sistemov za upravljanje s spletnimi vsebinami, kot so Drupal, Joomla!, Wordpress. Namenjeni so samostojnemu ustvarjanju, urejanju in vzdrževanju spletnih vsebin, ter upravljanju s celotnim spletiš em brez posebnega programerskega znanja.

Tega se zavedamo tudi pri ekipi skavtNET, kjer skrbimo za ažurnost informacij združenja slovenskih katoliških skavtinj in skavtov, zato smo za naše uporabnike, ki se soo ajo s takšnimi in podobnimi težavami, razvili lasten CMS (angl. Content Management Systems), ki se imenuje Skavt AROVNIK.

Skavt AROVNIK je bil zelo obširen projekt, zato nas je pri tem projektu sodelovalo ve razvijalcev programske kode. Moj prispevek je bil razvoj možnosti komentiranja novic in prikaz komentarjev k posamezni novici na spletni strani.

V drugem poglavju sem opisal CMS, njegove prednosti in slabosti, nato odprtokodne CMS sisteme (WordPress, Drupal, Joomla!). V tretjem poglavju sem bolj podrobno opisal CMS sistem Joomla!, ki smo ga kamniški skavti uporabili za izdelavo lastne spletne strani. V etrtem poglavju sem opisal Skavt AROVNIK. Podrobno sem opisal administrativni del in predstavil uporabljene tehnologije pri implementaciji komentarjev. Skavt AROVNIK sem primerjal z Joomla. Peto poglavje zajema opis programskega orodja CodeIgniter, na rtovanje komentarjev in implementacijo programske kode za komentarje. V zaključ u sem opredelil namen razvoja Skavt AROVNIKA s poudarkom na mojem prispevku pri tem projektu in

možnosti razširitve sistema. Na koncu sem dodal nekaj komentarjev uporabnikov Skavt AROVNIKA [1].

## 2. SISTEM ZA UPRAVLJANJE Z VSEBINAMI

Zaradi vedno večjega povpraševanja po programskih rešitvah, ki odpravljajo probleme obnavljanja spletnih strani, beležimo v zadnjih letih pravo ekspanzijo **komercialnih** (plačljivih) in **odprtokodnih** (brezplačnih) sistemov za upravljanje s spletnimi vsebinami.

Reševajo so sistemi CMS (angl. Content Management Systems pomeni sistem za upravljanje z vsebinami) namenjeni različnim tipom spletnih strani in reševajo pokrivajo zelo raznolik spekter funkcionalnosti, lahko rečemo, da imajo vsaj v osnovi enoten skupen cilj: na enostaven način (brez programiranja in urejanja HTML kode) omogočiti objavo vsebin na spletni strani, ne da bi imel uporabnik pri tem težave z načinom in obliko prikaza.

Kljudo uspešne predstavitve in prodaje na spletu je vsekakor vsebina spletne strani in ne toliko njen izgled ali tehnične zmožnosti, ki jih stran omogoča. Vsebine spletnih strani predstavljajo tekst, slike, avdio in video datoteke, pravzaprav vse, kar je v digitalni obliki in je možno prikazati. Večja je kot je podjetje, večja je potreba po pravočasnosti informacij na spletni strani, zato je tudi frekvenca dodajanja in spreminjanja spletne vsebine večja.

Uporabnost sistemov CMS se kaže v tem, da omogočajo preprosto in celovito obvladovanje spletnega mesta tudi tehnično nepodkovanim uporabnikom. Spreminjanje vsebine na spletni strani namreči pogojeno s poznavanjem programskih jezikov in internetnih tehnologij. Urejanje spletnih vsebin s pomočjo sistema CMS je zelo enostavno. Podobno je uporabi enega izmed urejevalnikov besedil, kot sta npr. Word ali Notepad. Poleg enostavne uporabe je ena izmed pglavitnih prednosti uporabe sistema CMS tudi ta, da lahko uporabnik prek spletnega brskalnika kadarkoli in od kjerkoli ureja vsebino spletnega mesta.

Ne glede na vrsto spletne prisotnosti (predstavitvena stran, trgovina, portal, itd.) se lahko z uporabo ustreznega sistema CMS spremembe na spletni strani opravijo hitro, enostavno in takrat, ko uporabnik to želi. V skrb za spletne predstavitve lahko vključimo tudi zaposlene. Dostop do sistema CMS ima lahko namreč več oseb znotraj podjetja, kar pomeni, da se lahko opravilo vzdrževanja spletnih strani porazdeli.

Poleg urejanja teksta, kjer je možnost upravljati z vrsto, velikostjo, barvo in s stilom pisave, lahko s CMS sistemom vplivamo tudi na poravnavo besedila, dodajanje, brisanje in urejanje slik, tabel, grafov, izdelkov v spletni trgovini, hipertekstualnih povezav, fotografij in podobno. Pri urejanju vsebine je za uporabnike pomembno, da je videz vsebinskih posodobitev, ki jih opravimo znotraj sistema CMS, enak končnemu videzu na spletni strani (princip WYSIWYG – What you see is what you get, kar pomeni: to kar vidiš, to dobiš). Sistemi CMS pa ne omogočajo le urejanja vsebine, temveč tudi celovito upravljanje spletnega mesta, vključno z njegovo strukturo (navigacijo), oblikovno podobo, pravicami uporabnikov in z vsemi moduli, ki so vključeni v spletno mesto (ankete, forum, novice, ipd). Tako lahko z

uporabo sistema CMS uporabniki npr. popravijo in dodajajo novice, ki se prikazujejo na osrednji strani njihove spletne strani kar v urejevalniku, ki se nahaja v sistemu, ne da bi se ukvarjali z oblikovanjem novice v HTML (angl. Hyper Text Markup Language pomeni ozna evalni jezik) kodi ali z njenim prenosom na spletni strežnik.

Kljub vsem funkcionalnostim, ki jih omogo ajo sistemi CMS, se je treba zavedati, da ni vsak CMS primeren za vse tipe spletnih strani. Nekateri sistemi puš ajo malo kreativne svobode, kar je posledica uporabe že vnaprej pripravljenih predlog, istega tipa navigacije ter dodatnih modulov, ki delujejo po istem principu in imajo zelo podoben izgled. eprav so enostavnost rokovanja in možnost spreminjanja in nastavljanja oblikovnih predlog zaželjene lastnosti vsakega sistema CMS, pa te prednosti niso edino, kar ho emo. Izdelovalci spletnih strani še posebej opozarjajo na nevarnost, do katere pride, ko po zaklju ku projekta prepustijo spletno stran, ki so jo izdelali, v samostojno upravljanje naro niku. Takrat oblikovalec nima ve kontrole nad obliko vnesenih vsebin in lastnik strani lahko za ne pretiravati s spreminjanjem strani in lahko se zgodi, da jih popolnoma zmali i.

Prednosti sistemov CMS poleg lastnikov spletnih strani s pridom izkoriš ajo tudi podjetja, ki se ukvarjajo z izdelavo spletnih strani, saj jim lahko uporaba sistema CMS zagotavlja poleg ve je preglednosti nad celotnim spletnim mestom tudi velike asovne in stroškovne prihranke na ra un hitrejše izdelave spletnega mesta (predvsem v primerih izdelave tipskih spletnih strani), kar se nenazadnje pozna tudi na ve jem zaslužku na posameznem projektu [1].

## 2.1. Kaj omogo ajo sistemi CMS?

Kljub temu, da deluje ve ina sistemov za upravljanje z vsebinami po podobnem principu, pa se med seboj najbolj razlikujejo po uporabnosti in po številu ter namembnosti razli nih funkcionalnosti, ki jih omogo ajo. Med najbolj pogosto ponujene funkcionalnosti sistemov CMS bi lahko prišteli:

- možnost integracije zunanjih CSS(angl. Cascadung Style Sheet pomeni kaskadni stilski slog) stilov, kar je z vidika samostojnega oblikovanja podobe spletne strani zelo pomembno,
- dodajanje in urejanje slik, tabel, grafov, povezav,
- samostojno urejanje strukture strani spletnega mesta – gradnja menujev in podmenujev,
- vklju evanje že vnaprej izdelanih modulov ali obrazcev za vnos podatkov na spletne strani (npr. aktualne novice, anketa, forum, galerija, poštna lista),
- možnost dodajanja dokumentov na spletno stran (PDF - angl. Portable Document Format pomeni odprt standard za izmenjavo elektronskih dokumentov, RAR datoteke - angl. Roshal Archive, pomeni sistem za stiskanje datotek,...),

- osnovno vodena objava vsebine (določimo lahko, kdaj se naj določena vsebina objavi in kako dolgo se naj prikazuje na spletni strani),
- možnost uvoza in izvoza podatkov (v Word, Excel, Access,...),
- kreiranje podatkov za RSS(angl. Really Simple Syndication pomeni zares preprosto zlaganje) sisteme vsebin,
- kreiranje meta oznak, ki so pomembne za pozicioniranje spletne strani v iskalnikih,
- možnost dodeljevanje pravic upravljanja (ve uporabniški sistem z reguliranim dostopom področnih urednikov do posameznih vsebin),
- možnost upravljanja z uporabniki,
- podrobna statistika obiskanosti spletnega mesta,
- večjezična podpora, ki omogoča postavitev strani v različnih jezikovnih variantah,
- arhiviranje podatkov,
- enostavno oblikovanje vsebine na podoben način, kot v večini urejevalnikov besedil,
- urejanje in posodabljanje spletne strani iz različnih lokacij,
- povezanost s podatkovnimi skladišči,
- administracija spletnih pasic na strani, z možnostjo merjenja prikazov pasice in klikov uporabnikov na pasico,
- administracija baze partnerjev in avtomatizirana korespondenca z njimi, v smislu obveščanja partnerjev o aktivnostih podjetja (angl. Partner Management),
- kategoriziranje vsebine v številne kategorije in podkategorije, z namenom lažjega iskanja informacij za uporabnika.

Večina sistemov CMS je zasnovanih modularno, kar pomeni, da so fleksibilni v smislu dodajanja novih in spreminjanja obstoječih zmožnosti glede na potrebe in zahteve naročnika. S posameznim modulom lahko uporabniki aplikacije urejajo vsebino določene spletne strani ali upravljajo s spletnimi servisi, kot so npr. ankete, novice, katalog izdelkov[1].

## 2.2. Slabosti sistema CMS

Na prednosti in slabosti, ki jih imajo CMS rešitve, gleda vsak uporabnik drugače, saj ima vsak lastnik spletišča svoje specifične potrebe in želje. Kar je prednost za nekoga, je lahko za drugega slabost in obratno. Težave s katerimi se soočajo uporabniki CMS sistemov lahko nastanejo tudi kot posledica izbire med komercialno in odprtokodno verzijo sistema. Poleg osnovnih prednosti, ki jih uporabnikom prinašajo komercialni sistemi CMS (ponudnik nudi podporo uporabnikom, omogoča nadaljnji razvoj sistema, sistem lahko proti doplatilu prilagodi isto specifičnim željam uporabnikov), pa je lahko njihova uporaba povezana tudi z določenimi slabostmi.

V prvi vrsti velja omeniti stroške nakupa programske opreme in nadaljnjega vzdrževanja, ki se

jim z uporabo odprtokodne rešitve lahko izognemo. Veliko dodatnih stroškov povzročijo tudi nove funkcionalnosti, ki jih želimo preko sistema vključiti na spletno stran in dodatne prilagoditve sistema.

Določene slabosti pa imajo tudi odprtokodni sistemi CMS. Uporabniki se lahko soočijo s precejšnjimi težavami, če se razvoj sistema preneha ali zaide v slepo ulico, kar pri brezplačnih različicah ni tako redek pojav. Ker se ves razvoj odvija na prostovoljni bazi, ne dajejo razvijalci nobenih garancij za pravilno delovanje sistema ali garancij, da bodo skrbeli za popravke in nadgradnje sistema in da bodo zagotavljali pomoč uporabnikom. Je pa tudi res, da nam zaradi odprte kode lahko nudi pomoč in opravlja nadgradnje skoraj vsak programer, ki pozna programski jezik PHP (angl. Personal Home Page Tool, pomeni orodje za osebno spletno stran).

Nikakor pa se ne moremo izogniti stroškom, ki so povezani z uvajanjem sistema, izobraževanjem uporabnikov in morebitnim prilagajanjem sistema CMS, kot posledice specifičnih potreb in želja [1].

## 2.3. Odprtokodni CMS sistemi

Za večino odprtokodnih sistemov lahko rečemo, da so razviti po »LAMP sistemu« (operacijski sistem Linux, spletni strežnik Apache, podatkovna baza MySQL (angl. My Structured Query Language, pomeni moj strukturalni poizvedben jezik) in skriptni jezik PHP, ki ga lahko nadomestita tudi Perl ali Python). Ker je koda odprtokodnih sistemov dostopna vsakomur, jo lahko seveda kdorkoli preuči, popravlja in nadgrajuje, kar je v nasprotju s komercialnimi rešitvami, ki nastajajo v zaprtem okolju in kjer je programska koda dostopna ozkemu krogu ljudi.

Z razvojem odprtokodnih rešitev, ki nastajajo v širši skupnosti razvijalcev, se lahko hitro razvijajo dodatne funkcionalnosti, ki so na voljo vsem. Zaradi svoje odprtosti in fleksibilnosti imajo odprtokodne rešitve veliko podporo v številnih programerjih, ki lahko nadgrajujejo, popravljajo in vzdržujejo programsko opremo, tudi če jo originalni izvajalec preneha razvijati. Velika izbira programerjev, ki obvladajo programski jezik PHP, pa omogoča, da se z najemom programerjev opravijo prilagoditve, ki bodo bolj ustrezale željam naročnika [1].

Opisal bom tri najbolj pogosto uporabljene CMS sisteme:

- Wordpress,
- Drupal,
- Joomla!.

### 2.3.1. WordPress

WordPress je odprtokodni sistem (angl. Open Source), ki temelji na jeziku **PHP** in z **MySQL** (angl. My Structured Query Language pomeni moj strukturni poizvedbeni jezik) podatkovno bazo v ozadju. Te e na **Apache** strežniku, ki ga podpira ve ina ponudnikov spletnih gostovanj. Sistem se namesti direktno na gostujo i strežnik, kjer se stran ureja preko prijave.

Sprva je bil WordPress strogo **blogerska platforma**, ki je postala priljubljena zaradi hitrega delovanja, enostavne uporabe ter vse bolj priljubljenega bloganja. Z leti se je WordPress razvil v profesionalni CMS z vsemi funkcionalnostmi, ki jih uporabljajo sodobne spletne strani [2].

#### Zgodovina

WordPress je uradni naslednik programa b2/cafelog. Razvijalec je uporabil programski jezik PHP in podatkovno bazo MySQL. Ime WordPress se je prvi pojavilo leta 2003, ki ga je predlagal Christine Selleck, prijatelj glavnega razvijalca Matta Millenwega.

Leta 2004 je mnogo uporabnikov, zaradi licen nih zahtev sistema Movable Type (spletni dnevnik) prešlo na nepla ljni sistem WordPress. Do oktobra 2009 je bil Wordpress najmo nejši odprtokodni sistem za upravljanje z vsebino [2].

Zadnja razli ica programa WordPress je 3.3.1, izdana pa je bila 3. januarja 2012.

#### Glavne lastnosti:

- samostojno, enostavno in poceni upravljanje z vsebino na spletnih straneh,
- z mnogimi dodatnimi razširitvami (angl. plugins) lahko na spletno vstran vklju imo sodobne animirane fotogalerije, interaktivne spletne aplikacije in celo spletno trgovino,
- ena ve jih prednosti WordPress-a pred ostalimi CMS sistemi je vgrajen sistem za optimizacijo spletnih strani, ki sam vzdržuje optimizacije vsebine za potrebe iskalnikov,
- klju ne lastnosti bloga, ki omogo ajo pisanje in komentiranje objav. Omogo imo lahko tudi »zbiranje« uporabnikov oz. obiskovalcev spletne strani ter njihovo registracijo.

### 2.3.2. Drupal

Drupal je ve krat nagrajen odprtokodni sistem upravljanja vsebin, ki uporablja bazo podatkov MySQL ali PostgreSQL, od verzije 4.6 dalje pa je kompatibilen s PHP5. Fleksibilnost, kvalitetna in razumljiva koda, modularna zgradba in prilagodljivost, so pglavitne kvalitete, ki jih navajajo uporabniki omenjenega sistema. Še danes se mu pozna, da je bil na svojem

za etku predvsem orodje za gradnjo spletnih skupnosti, saj ima mnogo funkcionalnosti, ki podpirajo ta tip spletnih strani. Drupal je še posebej dober za vzdrževanje novi arskih spletnih strani in spletnih dnevnikov (blogov), seveda pa se lahko uporablja tudi za vzdrževanje obi ajnih spletnih strani.

Med najbolj uporabne funkcionalnosti sodijo: možnost spreminjanja vsebine v PDF format, izgradnja interaktivnih obrazcev (npr. forma za povpraševanje) prek administrativnega vmesnika, URL(angl. Uniform Resource Locator, pomeni enoli ni naslov vira) filter, ki omogo a samodejno filtriranje vseh URL in e-poštnih naslovov in njihovo spreminjanje v povezave, RSS agregator vsebin, arhiviranje starejših vsebin, možnost komentiranja vsebine s strani obiskovalcev itd [4].

### **Zgodovina**

Za etki Drupala segajo v leto 2001, ko je med svojim študijem Dries Buytaert razvil in postavil na internet »Message Board« (oglasna deska). Zaradi velikega zanimanja za svoj projekt in edalje ve jih zahtevah po izboljšavah, se je odlo il, da bo ponudil izvorno kodo tudi drugim. Tako je za el razvijati Drupal skupaj z drugimi lani skupnosti <http://drupal.org> [5].

### **Glavne prednosti Drupal-a so:**

- je odprtokoden in s tem brezpla en za uporabo,
- ima široko mrežo uporabnikov in razvijalcev, ki stalno nadgrajujejo sistem,
- zagotavlja optimizirano delovanje in je prilagodljiv,
- dovršen sistem sporo anja in statistike omogo a dobro delovanje sistema,
- je izjemno prijazen do spletnih iskalnikov, kar omogo a visoke pozicije na iskalnikovih izpisih,
- je ve jezi en,
- številni dodatni moduli ponujajo skoraj neomejene razširitve obstoje ih spletnih strani,
- kvalitetna koda in modularna zgradba.

Podjetja in organizacije kot so NASA, IBM, Yahoo, FedEx, Nike, Forbes, BBC, Greenpeace, MTV ter mnoga druga, ki uporabljajo odprtokodni Drupal CMS, so spoznala izjemno mo in prilagodljivost enega najzmogljivejših in v svetu vodilnih CMS sistemov [6].

### **2.3.3. Joomla!**

Joomla! je odprtokodni sistem za opravljanje s spletnimi vsebinami, ki temelji na jeziku PHP in podatkovni bazi MySQL. Za spletni strežnik obi ajno uporablja Apache, deluje pa tudi na Microsoft IIS. Uporabniku omogo e izgradnjo enostavnih in zahtevnejših spletnih strani. Je namenjen uporabnikom, brez znanja programiranja [7].

## Zgodovina

prav se je Joomla! pojavila leta 2005, njene korenine segajo v leto 2001, ko so inženirji družbe Miro Corporation ustvarili sistem za upravljanje vsebin imenovan Mambo. Za tisti čas je bil Mambo presenetljivo napreden, pridobil je na popularnosti po vsem svetu, vendar pa se je njegov razvoj v letu 2005 ustavil zaradi nesoglasij med razvijalci in neprofitno fundacijo za vodenje tega razvoja. Na podlagi preteklih izkušenj so se razvijalci posvetili ustvarjanju novega CMS – rodila se je Joomla! in takoj postala priljubljena med uporabniki [7].

## 2.4. Primerjava CMS-sistemov

Tabela 2.1 prikazuje število prenesenih instalacij CMS sistemov na teden in procent obiskanosti strani CMS sistemov v letu 2010. Uporabniki so pri izdelavi bloga ali spletne strani največkrat uporabili CMS sistem WordPress. Glavna razlika med WordPress-om in ostalimi CMS sistemi je v enostavnejšem vmesniku za izdelavo spletnih strani in v primernosti za izdelavo blogov. Posledično imajo spletne strani ali blogi, ki so izdelane s pomočjo WordPress-a, največ obiskov [3].

CMS sistemi	WordPress	Joomla!	Drupal
Tedenski prenos v letu 2010	983.625	113.836	33.671
Najbolj obiskani CMS v letu 2010	12,9%	2,5%	1,4%

Tabela 2.1 Primerjava odprtokodnih CMS sistemov.

### 3. UPORABA CMS SISTEMA JOOMLA! V PRAKSI

Aprila 2008 smo se skavti iz Kamnika odločili za izdelavo spletne strani s pomočjo CMS Joomla!. Pred to odločitvijo smo imeli izdelano preprosto stran v HTML-ju. Želeli smo različnim uporabnikom dodeliti pravice pri upravljanju s spletno stranjo in omogočiti enostavno upravljanje z vsebino. To je bilo pri obstoječi spletni strani v HTML-ju nemogoče zagotoviti, zato smo morali preiti na novo rešitev. Lahko bi se lotili prenove obstoječe spletne strani, vendar bi ob tem porabili preveč časa in denarja. Zato smo se odločili za izbor enega od odprtokodnih CMS-jev, Joomla!.

#### 3.1. Razlogi za izbiro Joomla, kot najprimernejši CMS sistem:

- preprosta izdelava kompleksne strani v kratkem času – Joomla! omogoča takojšnje delo po namestitvi (ni potrebno nameštiti dodatnih orodij),
- enostavno dodajanje vsebine na spletno stran (novice, fotogalerije, napovedi dogodkov,...),
- enostavno vodenje evidence uporabnikov in dodelitev njihovih pravic za upravljanje s spletno stranjo,
- administrator ima pregled nad celotno spletno stranjo,
- osnovna namestitvev je podpirala vse naše funkcionalnosti, ki smo jih želeli podpreti,
- na svetovnem spletu je veliko dokumentacije za uporabo (navodila za uporabo) in reševanje določenih problemov,
- programska koda je jasna, kar omogoča razvijalcem lažje razvijanje dodatnih komponent,
- brezplačna orodja, ki so potrebna za namestitvev Joomla.

#### 3.2. Spletna stran kamniških skavtov

V nadaljevanju bom opisal, praktično uporabo CMS Joomla! na spletni strani:

<http://kamnik1.skavt.net/> - spletna stran stega (lokalna enota) Kamnik 1, katerega član sem tudi sam (slika 3.1 prikazuje spletno stran).

The screenshot shows the homepage of the Kamniški skavti - Mamutov steg website. At the top, there is a navigation menu with links: « Kdo smo, Starostne skupine, Fotogalerije/arhiv novic, Kontakt/včlanitev, ZSKSS. Below the navigation is a header area with a large mammoth logo on the left, two small logos in the center, and a group photo of skauts on the right. The main content area is divided into several sections: 'Zadnje novice' (Latest news) with a list of recent news items; 'Zadnje fotogalerije' (Latest photo galleries) with a list of gallery items; 'Dobrodošli na straneh kamniških skavtov' (Welcome to the pages of the Kamniški skavti) with a date and time, a welcome message, and a list of activities; and 'RSS - Prijavi se na prejemanje novic' (RSS - Sign up for news) with a subscription button. The page also shows 'Prisotni 1 gost' (1 guest present).

Slika 3.1 Osrednje spletne strani kamniških skavtov [8].

Spletna stran je razdeljena na osrednje, ki je vidno obiskovalcem in prijavljenim uporabnikom, ter ozadju, ki je namenjeno administratorjem spletne strani.

Za navigacijo po osrednju so na voljo trije meniji:

- osnovni (angl. Main menu),
- zgornji (angl. Top Menu) in
- uporabniški (angl. User Menu).

Ozadje (administrativni del) nam omogoča kontrolo nad vsebino, izgledom in vsem ostalim, kar vpliva na obliko strani.

Vsebinsko predstavljajo članki (angl. Articles). Prikazujejo se na osrednju sistema. Za Joomla! je značilna ilen tri - nivojski sistem kategorizacije vsebine:

- področje (angl. Section) – najvišji nivo,
- kategorija (angl. Category) – pripada enemu področju in
- članek – pripada eni kategoriji.

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

**lanek** je nosilec vsebine, saj vsebuje besedilo, slike in druge datoteke.

Joomlo sestavljajo:

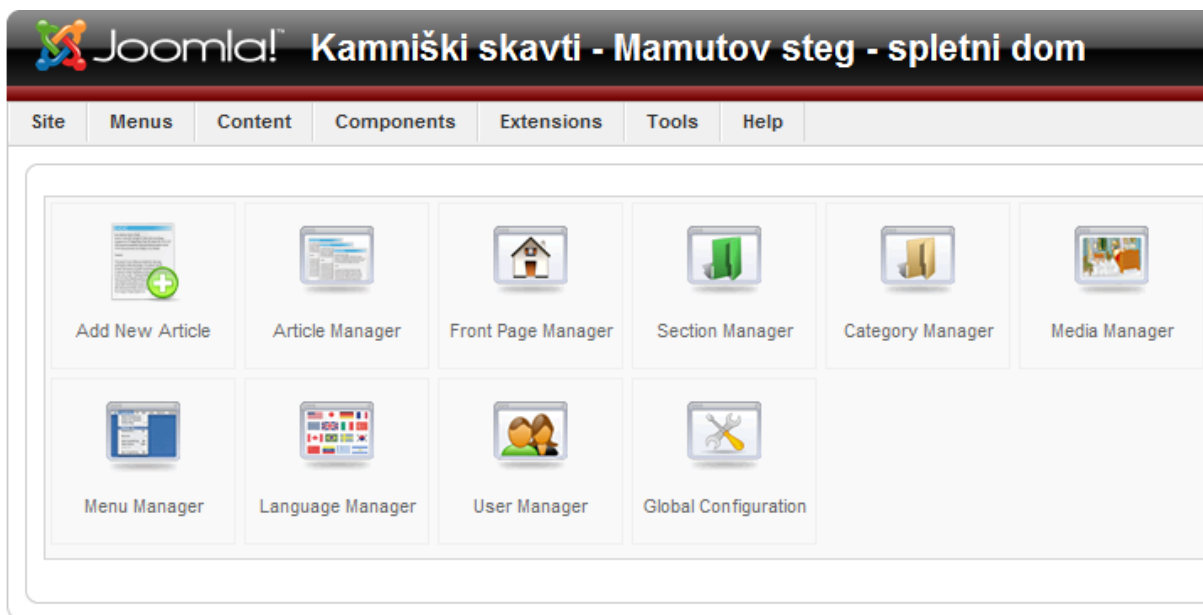
- komponente (angl. Components),
- moduli (angl. Modules) in
- vti niki (angl. Plug-ins).

**Komponente** predstavljajo funkcionalnosti, vsebujejo svojo poslovno logiko in imajo navadno svoje mesto na kontrolni ploš i. Tipi no je to na primer spletna trgovina, forum, vmesnik za upravljanje z uporabniki, vmesnik za vnos novic,...

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

**Vti niki** so del ki programske kode nameš eni na dolo enem mestu v ogrodje sistema, ki dodajo ali pa spreminjajo njegovo funkcionalnost.

Administrator upravlja s sistemom preko nadzorne ploš e v ozadju sistema. Ukaze posreduje preko menijev ali gumbov na nadzorni ploš i, ki jih prikazuje slika 3.2 [22].



Slika 3.2 Nadzorna ploš a sistema Joomla! [9].

Pravice dostopa do spletiš a so urejene preko sistema skupin, ki imajo vnaprej dolo ene pravice.

Skupine so razdeljene na dva tipa :

- Skupine uporabnikov ospredja – ti se lahko prijavijo samo v ospredju:
  - Registrirani uporabniki (angl. Registered)
    - lahko se prijavijo na spletno stran,

- pregledujejo vsebino, ki je označena, kot »Registered« in »Public«.
- Avtor (angl. Author)
  - podedujejo pravice skupine registrirani uporabniki,
  - imajo pravico dodajanja nove vsebine.
- Urednik (angl. Editor)
  - podedujejo pravice skupine avtor,
  - imajo pravico urejenja vsebine.
- Odgovorni urednik (angl. Publisher)
  - podedujejo pravice skupine urednik,
  - imajo pravice dodajanja vsebine,
  - vsebino lahko vzamejo iz objave.
- Skupine uporabnikov ozadja - ti se lahko prijavijo v ozadje sistema (nadzorna plošča):
  - Ravnatelj (angl. Manager)
    - podedujejo pravice skupine odgovorni urednik,
    - lahko dostopa do nadzorne plošče sistema, kjer imajo omejene pravice.
  - Administrator
    - podedujejo pravice skupine ravnatelj,
    - imajo nekaj več pravic pri upravljanju nadzorne plošče.
  - Super Administrator
    - imajo vse pravice nad spletno stranjo [22].

## 4. OPIS Skavt AROVNIKA

Skavt AROVNIK (slika 4.1) je sistem za upravljanje s spletno vsebino, ki je brezplačno na voljo vsem članom ZSKSS (Združenje slovenskih katoliških skavtinj in skavtov). Večina programske kode je napisana v spletnih programskih jezikih PHP, JavaScript, HTML. Za shranjevanje podatkov uporablja MySQL podatkovno bazo. Sistem Skavt AROVNIK teče na Apache strežniku, ki ga podpira večina ponudnikov spletnih gostovanj.

Sistem je zasnovan na enostavni uporabi za končnega uporabnika, saj je urejanje v vmesniku logično predstavljeno s hierarhijo strani in podstrani na spletni strani. Končni uporabnik tako vedno ve, na kateri poziciji se nahaja in na katerem nivoju dodaja ali ureja stran oz. podstran.



Slika 4.1 Skavt AROVNIK [10].

Skavt AROVNIK lahko razdelimo na dva dela:

- vidni del (angl. Front-end) in
- administrativni del (angl. Back-end ali Administration).

### 4.1. Vidni del

Prijava SL | EN



**skavtinje in skavti**  
Združenje Slovenskih  
Katoliških Skavtinj  
in Skavtov

POMAGAJ NAM O SKAVTUH ZSKSS NOVICE HITRE POVEZAVE ZA MEDIJE KONTAKT

Postlatni skavt	sl > ZSKSS
Poslanstvo in program	<b>Slovenska skavtska organizacija</b>
Skavtska meloda	 <p>Slovenska skavtska organizacija (Združenje slovenskih katoliških skavtinj in skavtov) je prostovoljna nacionalna <b>mladinska organizacija, ki se ukvarja z vzgojo in izobraževanjem otrok in mladit.</b></p> <p>Temelji na načelih, ki jih je v začetku 20. stoletja napisal Robert Eaden-Powell, ustanovitelj skavtskega gibanja. ZSKSS je bilo ustanovljeno leta 1990 na temeljih predvojnega slovenskega skavtstva.</p> <p>ZSKSS ima vsako leto približno <b>4300 aktivnih članov</b>, večinoma starih od 6–30 let, ki so razdeljeni v tri starostne skupine, približno 750 pa je <b>voditeljev prostovoljcev.</b></p> <p>Osnovni namen vzgoje v ZSKSS je pripovedi k polnemu telesnemu, duševnemu, duhovnemu in družbenemu razvoju otrok in mladit, da bodo lahko postali trdne osebnosti, odgovorni državljani ter člani krajevnih, narodnih in mednarodnih skupnosti.</p> <p>Neformalna vzgojno-izobraževalne dejavnosti izvajamo s pomočjo <b>skavtske metode</b>, ki je v svetu priznana kot ena najučinkovitejših neformalnih vzgojnih metod za vzgojo samostojnih ljudi in obojsko vzgojo.</p> <p>Vsako leto izvedemo več kot 300 večdnevni <b>zimovanj in poletnih taborov</b> ter drugih dejavnosti, ki so dopolnilo šoli predvsem na področju aktivnega državljanstva, osebnostnega razvoja, okoljske vzgoje, medgeneracijskega sodelovanja, pridobivanja veščin za samostojno življenje ter. Letno izvedemo tudi približno 12 daljših <b>usposabljanj</b> za mladinske voditelje in prostovoljce ter več raznovrstnih <b>projektov</b></p> <p>Imamo status društva v javnem interesu na področju vzgoje in izobraževanja, status nacionalne mladinske organizacije in smo člani <b>svetovne skavtske organizacije WAGGGS</b> (World Association of Girl Guides and Girl Scouts).</p>
Skavtska obljuba in zakoni	
Skavti kot del civilne družbe	
Mednarodno delovanje	
Organiziranost ZSKSS	<p><b>Iščemo nove skavte ...</b></p>  <p><b>Najdi nas na Facebooku</b></p> <p><b>Skavtinje in skavti</b> VseČim je</p> <p>776 ljudem je všeč <b>Skavtinje in skavti</b>.</p>  <p>Jernej Urška Andrija Erika Klavdija Matevž Matej Tom</p> <p>Facebookov socialni vodnik</p>
Izvršni odbor	
Regije	
Stegi (lokalne enote)	
Veje (starostne skupine)	
Nadžorni odbor	
Časni o razsodišče	
Strokovna služba	
Zgodovina ZSKSS	
Usposabljanje v ZSKSS	
Taborne šole	
Skavtska šola Življenje v naravi	
Skavtski okoljski center	
Dokumenti in obratci	
Skavtarnica	
Skavtska kazalnica	
Podpirajo nas	
Postanite naši podporniki	
Projekti ESS	
Medgeneracijsko sodelovanje	
Mladinski delavec	
Aktivendržavljan.si	

© Slovenske skavtinje in skavti - Slovenian guides and scouts • 2011-2012 SkaviCAROVNIK • Pagnji uporaba

Slika 4.2 Vidni del Skavt AROVNIKA [11].

Vedno, ko obiskovalec spletne strani obiše stran, se mu prikaže vidni del in na njem prikazana vsebina. Izgled vidnega dela dolo i administrator ob kreiranju spletne strani (primer izgleda vidnega dela predstavlja slika 4.2).

Pri Skavt AROVNIKU vidni del vsebuje:

- strani, podstrani (zgodovina, predstavitev, kontakti,..),
- vsebino v obliki besedila, slik in drugih datotek,
- dostop do socialnih omrežij (npr. facebook) in
- možnost prijave uporabnikov.

## 4.2. Administrativni del

Pozdravljeni, Andrej — Administracijski kuliček — [odjava] SL | EN

ZSKSS skavtinje in skavti  
Združenje Slovenskih Katoliških Skavtinj in Skavtov

PREGLED STRANI NA PRVO STRAN

**Objave**  
Napiši novo  
Ureaj  
Dodaj kategorijo  
Ureaj kategorije

**Strani**  
Ustvari novo  
Ureaj

**Medijska knjižnica**  
Dodaj novo datoteko  
Ureaj

**Nastavitve**  
Navigacija  
Izgled strani  
Naslovna slika

**Objava**

**Pregled strani**

Novice » Kategorije »

**1** **3**

Strani »

**73**

Slike » Datoteke »

**80** **85**

**Obvestila ekipe**

**Težave pri urejanju navigacije?**

Si že dodal/a želene podstrani v navigacijo pa ne veš kako naprej? Zakaj ni nobene vsebine na teh podstranih? Ker nisi edini/a s temi težavami smo se odločili, da t stvar nekoliko olajšamo. [več »](#)

**Novosti v SkavtČAROVNIKU**

V SkavtČAROVNIKU uvajamo nekaj novosti: Obvestila ekipe SkavNET in teme za spletne strani. [več »](#)

Vsa obvestila ekipe SkavNET »

© Slovenske skavtinje in skavti - Slovenian guides and scouts • 2011-2012 SkavtČAROVNIK • Pogoji uporabe

Slika 4.3 Administrativni del Skavt AROVNIKA [12].

Administrativni del, ki ga prikazuje slika 4.3 omogoča vso kontrolo nad vsebino, izgledom in vsem ostalim, kar vpliva na obliko strani. Administrativni del vsebuje:

- dodajanje, urejanje, brisanje novic, strani, kategorij,
- nalaganje, urejanje, brisanje slik in drugih datotek,
- zaklepanje dostopa vsebin,
- urejanje strukture strani,
- preprosto urejanje izgleda strani:
  - možnost izbire različnih tem in
  - spremembo naslovne slike, barve besedila in ozadja strani;
- nalaganje lastnega CSS in
- osnovne nastavitve strani.

### **4.2.1. Dodajanje, urejanje, brisanje novic**

Pri dodajanju novic imamo možnost vpisa naslova, kratkega povzetka, vsebine, kreiranja foto galerije, nalaganja slik in drugih datotek, izbire kategorije in avtorja novice, izbire vidnosti novice glede na vrsto uporabnika. Lahko dovolimo dodajanje komentarjev k novici, določimo datum objave, novico lahko prestavimo na vrh, kar pomeni, da je bolj pomembna in bo vidna prva na strani in pri vsaki novici lahko definiramo kategorijo, kjer se bo pojavila.

Pri dodajanju novic moramo obvezno izpolniti polja: naslov, povzetek, avtor in izbor kategorije.

### **4.2.2. Dodajanje, urejanje, brisanje strani**

Pri dodajanju strani imamo možnost vpisa naslova, kratkega povzetka, vsebine, avtorja, kreiranje foto galerije, nalaganja slik in drugih datotek, izbire vidnosti strani glede na vrsto uporabnika in datum objave strani. Pri dodajanju strani moramo obvezno izpolniti naslov strani.

Glavna razlika med stranjo in novico je, da se skozi čas stran spreminja (vsebina na strani), novica pa ne, saj ostane na strani v arhivu novic.

### **4.2.3. Dodajanje, urejanje, brisanje kategorij**

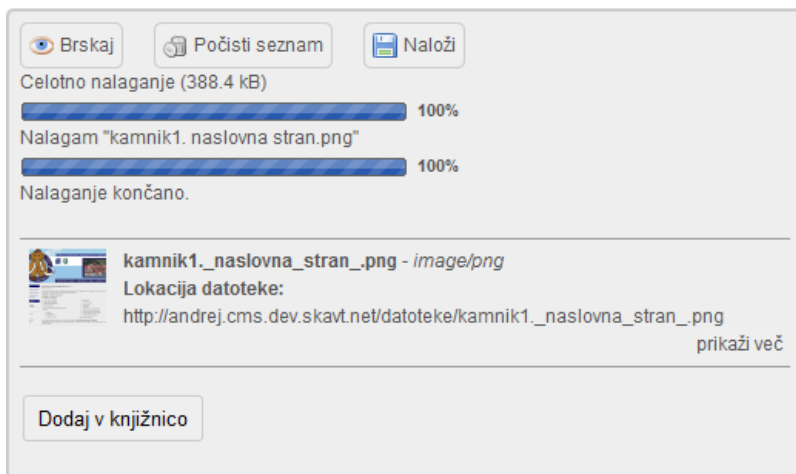
Pri obširnih spletnih straneh imamo vsebino razdeljeno na posamezne kategorije. Tako je vsebino lažje urejati, pri iskanju izgubimo manj časa in posledično je stran za administratorja bolj prijazna.

Pri kategoriji moramo obvezno izpolniti njen naslov. Pri dodajanju kategorij imamo možnost vpisa naslova, opisa in izbire vidnosti kategorije glede na vrsto uporabnika.

### **4.2.4. Nalaganje, urejanje, brisanje slik in drugih datotek**

Nalaganje slik in drugih datotek poteka preko modula (angl. upload plugins) za nalaganje datotek, kar vidimo na sliki 4.4. Preden uspešno naložimo datoteko na server preverimo, da je datoteka večja od 1B. Vsako uspešno naložena datoteko shranimo v medijsko knjižnico, ki jo prikazuje slika 4.5, kjer beležimo datum objave, ime in prikaz datoteke v pomanjšani velikosti. Datoteke lahko enostavno iščemo glede na njihova imena in datume objave. Prav tako imamo tukaj možnost izbrisa datotek.

## Naloži novo datoteko



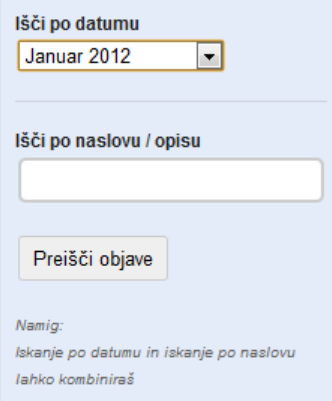
Slika 4.4 Nalaganje datotek preko modula.

## Urejaj datoteke



Slika 4.5 Urejanje in iskanje datotek – medijska knjižnica.

Težave pri nalaganju datotek?  
Poskusi s [preprostimi nalagalnikom...](#)



### 4.2.5. Zaklepanje dostopa vseh vsebin

Skavt AROVNIK uporablja naslednji na in strukturiranja vsebine:

- Novice:
  - vsaka novica mora pripadati eni izmed kategorij.
- Kategorije:
  - novice lahko združujemo po kategorijah, saj je tako sama stran bolj pregledna.
- Strani:
  - stran vsebuje eno ali več novic.
- Slike:
  - seznam vseh slik, ki jih je avtor naložil preko modula na spletišču.

- Datoteke:
  - seznam vseh datoteke brez slik.

Skavt AROVNIK omogoča, da vsebino na spletni strani razdelimo na 4 različne skupine:

- Javno:
  - vsebina na strani je vidna vsem, ki dostopajo do spletne strani.
- Prijavljeni uporabniki:
  - podedujejo pravice skupine Javno,
  - vsebina je vidna vsem prijavljenim uporabnikom.
- Voditelji:
  - podedujejo pravice skupine Prijavljeni uporabniki,
  - vidno vsem prijavljenim uporabnikom, ki imajo definiran status voditelj v bazi članstva,
  - ta skupina je namenjena predvsem zaklepanju nekaterih vsebin, ki so vidne samo voditeljem.
- Uredniki:
  - podedujejo pravice skupine Voditelji,
  - imajo vse pravice nad spletno stranjo.

Dostop je možno omejiti za vsako od zgoraj naštetih oblik strukturiranja vsebin.

#### **4.2.6. Urejanje strukture strani**

Slika 4.6 prikazuje meni Navigacija, kjer urejamo strukturo strani. Tukaj imamo možnost dodajanja strani in podstrani. Vsaka stran in podstran je sestavljena iz obveznega sredinskega dela in neobveznega desnega dela, kjer se prikazuje vsebina. Na oba dela spletne strani lahko priprnemo napisano novico.

V primeru izbrisa strani, ki vsebuje tudi podstrani, se izbrišejo tudi vse podstrani. Na koncu dobimo zemljevid strani.

The screenshot shows a web management interface with a dark blue header containing 'PREGLED STRANI' and 'NA PRVO STRAN'. On the left is a vertical sidebar with orange and grey sections: 'Objave' (with options: 'Napiši novo', 'Ureaj', 'Dodaj kategorijo', 'Ureaj kategorije'), 'Strani' (with options: 'Ustvari novo', 'Ureaj'), 'Medijska knjižnica' (with options: 'Dodaj novo datoteko', 'Ureaj'), 'Nastavitve' (with options: 'Navigacija', 'Izgled strani', 'Naslovna slika'), and 'Odjava'. The 'Navigacija' option is highlighted with a black arrow. At the bottom of the sidebar is a logo for 'ZSKSS' (Združenje Slovenskih Katoliških Skavtj in Skavtov) and the text 'Smo del Združenja Slovenskih Katoliških Skavtj in Skavtov' and '© Andrej DEV • 2010-2012'. The main content area is titled 'Navigacija' and contains a 'Glavni meni' section with a tree structure: 'sl' (with a Slovenian flag icon), 'gamsov steg', 'Kratka predstavitev', 'Zgodovina' (highlighted in blue), and 'Starostne skupine'. Below the menu is a 'Shrani' button and a text box explaining that clicking on any of the existing links will edit the main menu, selecting content on the homepage will edit the language flag, and selecting public content will edit the public content. Below this are four main sections: 'Urejanje/dodajanje elementa v strukturo' (with a green plus icon), 'Kreiranje strani (npr.: zgodovina) in direktno dodajanje v strukturo' (with a blue document icon), 'Urejanje povezave' (with a grey link icon) including 'dodajanje obstoječih vsebin (strani, novice, kategorije)' and 'ročni vnos povezave', and 'Preimenovanje' (with a yellow pencil icon). At the bottom is 'Brisanje' (with a grey trash icon).

Slika 4.6 Navigacija na spletni strani.

#### 4.2.7. Preprosto urejanje izgleda strani

Skavt AROVNIK vsebuje tri različne prednaložene teme strani, zato je priporočeno, da najprej izberemo ustrezno temo, šele nato spreminjamo barve posameznih delov strani. Barvo lahko spremenimo ozadju, besedilu, nazivu strani, ozadju spodnjega dela, zunanjemu in notranjemu ozadju, levemu meniju in obrobam besedila. Narejene spremembe opazimo takoj v predogledu strani. Na sliki 4.7 lahko vidimo, kako enostavno je spreminjati barve na spletni strani.

The screenshot shows the 'PREGLED STRANI' (Preview Page) interface for the SkavtČAROVNIK website. The main content area displays a preview of the website header with the title 'SkavtČAROVNIK' and the author 'Andrej DEV'. Below the preview is a 'PREPROSTA IZBIRA BARV' (Simple Color Selection) tool with fields for 'Ozadje glave' (001066), 'Tekst' (FFFFFF), 'Naziv strani' (FFFFFF), and 'Ozadje noge' (001066). A 'Shranj spremembe' (Save Changes) button is at the bottom. A right sidebar contains a 'Barvna shema' (Color Scheme) section with instructions on how to use the color selection tool.

Slika 4.7 Urejanje izgleda strani Skavt ČAROVNIKA.

#### 4.2.8. Nalaganje lastnega CSS

Administrativni vmesnik v zavihku CSS omogoča nalaganje lastnega CSS. To je namenjeno predvsem bolj zahtevnim uporabnikom.

#### 4.2.9. Osnovne nastavitve strani

Za konec kreiranja spletne strani moramo izpolniti še osnovne podatke o strani. Sem spadajo naslednji podatki: naziv strani, ustreznost postavitev naziva strani, določitev ključnih besed spletišča, kratek opis, vpis administratorjevega e-naslova, vpis sledilne kode za Google Analytics, izbor števila objav na strani in izbor gumba Facebook Like. Ustrezna izbira ključnih besed in opisa strani sta zelo pomembna, saj pripomoreta k boljši uvrstitvi spletne strani na spletnih iskalnikih.

### 4.3. Zakaj smo se odlo ili razviti Skavt AROVNIKA?

V današnjem okolju je lastna spletna stran osebna izkaznica za ekonomske subjekte, razne organizacije, društva in tudi posameznike.

Organizacija ZSKSS deluje na treh ravneh, lokalni, regijski in državni. Na lokalni ravni je sestavljena iz treh regij (ljubljanska, mariborska in primorska), regije pa sestavlja preko 60 stegov. Na nacionalni ravni je najvišji organ Svet Združenja. Prav vsi želijo svoje spletne strani, zato smo se odlo ili, da je pravi trenutek, da razvijemo Skavt AROVNIK za skavte. Trenutno je registriranih 82 domen za Skavt AROVNIKA, od tega je aktivnih 25 strani.

### 4.4. Prednosti in slabosti Skavt AROVNIKA glede na druge CMS sisteme

#### Prednosti:

- splet nam danes omogo a ogromno izbire med razli nimi CMS sistemi, zato moramo imeti veliko znanja, da izberemo pravi CMS, ki najbolj ustreza našim potrebam,
- administracija pri Joomla oz. drugih CMS sistemih je za uporabnika težavna, ker omogo ajo uporabo veliko funkcionalnosti, ki jih obi ajni uporabnik ne potrebuje, zato povzro ajo zmedo,
- ko naletimo na težavo, napako lahko hitro odpravimo, saj poznamo sistem do potankosti,
- zaprta koda - varnost pred hekerskimi vdori.

#### Slabosti:

- porabljen as in denar za razvoj aplikacije,
- premalo razvijalcev ima vpogled v kodo, kar pomeni, da sistema ni možno redno posodabljati in izboljševati.

### 4.5. Predstavitev uporabljene tehnologije pri Skavt AROVNIKU

Opisal bom uporabljene tehnologije za razvoj aplikacij, ki smo jih v ekipi uporabljali za razvoj Skavt AROVNIKA.

#### 4.5.1. Jezik HTML

**HTML** (angl. HyperText Markup Language) je programski jezik, ki z zaporedjem ukazov pove, kako naj se stran prikaže na zaslonu. Popularen je postal kmalu po svojem nastanku leta 1990, od takrat pa je doživel že precej sprememb. Ukaze imenujemo zna ke oziroma po angleško

»tags«. Za etek zna ke vedno pišemo med znake < in >, konec pa med </ in >, datoteke imajo kon nico .html ali .htm. HTML ozna evalni jezik se uporablja za izdelovanje spletnih strani na svetovnem spletu. Je podmnožica mednarodnega standarda imenovanega SGML – Standard Generalized Markup Language. Uporabnik za pregledovanje spletne strani potrebuje posebno programsko opremo, ki jo imenujemo brskalniki. Jezik je dokaj preprost in se lahko kombinira z ostalimi spletnimi jeziki, kot so JavaScript, PHP, XML [13, 14].

HTML ponuja:

- elektronsko objavo dokumentov z besedili, tabelami, fotografijami...,
- preusmeritev in napotitev na druge informacije s pomo jo hiperpovezav,
- uporabo obrazcev, ki omogo ajo iskanje po spletu, naro anje izdelkov in
- dodajanje video izrezkov, zvoka in drugih aplikacij v dokumente.

#### 4.5.2. Jezik JavaScript

**JavaScript** je objektni skriptni programski jezik za ustvarjanje interaktivnih spletnih strani. Namenjen je pisanju programov, ki jih lahko nespremenjene izvajamo na vsakem računalniku, ne glede na vrsto procesorja ali operacijskega sistema. Zaradi teh lastnosti so programi idealni za vklju itev na spletne strani, saj tam ne vemo v naprej, na kakšnih vrstah računalnikov bodo uporabniki prebirali naše strani.

Program vgradimo ali vklju imo v HTML, da opravlja naloge, ki niso mogo e samo s statično strukturo. Na primer: odpiranje novih oken, preverjanje pravilnost vnesenih podatkov, enostavni izra uni, itd [15].

##### Zgodovina

Da bi izboljšali podporo programom, napisanim v jeziku Java so pri podjetju Netscape še istega leta razvili programski jezik, ki so ga sprva poimenovali LiveScript, iz tržnih razlogov pa so ga kmalu preimenovali v JavaScript (zaradi popularnosti Java). To ni bila dobra poteza, saj je mnogo uporabnikov mislilo, da sta Java in JavaScript isto. Kljub za etnim pomankljivostim (slaba varnost, nestandardiziranost, pomanjkanje razvojnih orodij) je JavaScript s asoma postal eno od najbolj priljubljenih orodij za izdelavo dinami nih spletnih strani [15].

#### 4.5.3. Jezik PHP

**PHP** (angl. Personal Home Page Tool) je razširjen odprtokodni programski jezik, ki se uporablja za razvoj dinami nih spletnih vsebin, ki rpajo podatke iz podatkovne baze.

PHP deluje na spletnem strežniku, kjer kot izhod generira HTML kodo s pomojo PHP jezika. PHP jezik je zelo enostaven za novince v programiranju, kot tudi za programerje, ki so vajeni drugih programskih jezikov, ker omogoča hitro privajanje in enostavno uporabo. Izdelavo spletne strani kreiramo s HTML-jem in PHP-jem. V statino HTML stran vključimo PHP kodo, ki je omejena s posebnimi PHP znaki. Ko obiskovalec odpre stran se za ne na strežniku izvajati PHP koda, ki obiskovalcu vrne izpis v HTML obliki, tako da dejansko obiskovalec ne vidi PHP kode v ozadju, za razliko od JavaScripta.

PHP omogoča odlično kompatibilnost z različnimi podatkovnimi zbirkami kot so: MySQL, Oracle, Sybase, PostgreSQL, Mungo in Generic ODBC. Najbolj popularna kombinacija je PHP-MySQL. Kot PHP je tudi MySQL odprtokodni sistem. PHP vsebuje ogromno spisanih funkcij na medmrežju: povezava na oddaljen strežnik, preverjanje e-pošte preko POP3 (angl. Post Office Protocol version 3) oz. IMAP (angl. Internet Message Access Protocol) protokola, nastavljanje piškotkov (angl. Cookies), itd [16].

### **Zgodovina**

PHP je bil napisan kot skupina CGI-programov v programskem jeziku C. Napisal ga je dansko kanadski programer Rasmus Lerdorf leta 1994, da bi zamenjal nekaj skript napisanih v Perlu, ki jih je uporabljal za upravljanje svoje spletne strani.

Lerdorf je sprva napisal PHP, da bi prikazal svoj življenjepis in hkrati zajemal podatke obiskovalcev strani. Tako je 8. julija 1995 izdal »Personal Home Page Tools«, potem ko ga je združil z svojim interpreterjem za spletne obrazce (PHP/FI).

### **4.5.4. Podatkovna baza MySQL**

**MySQL** (angl. My Structured Query Language) je sistem za upravljanje s podatkovnimi bazami. MySQL je odprtokodna implementacija relacijske podatkovne baze, ki za delo s podatki uporablja jezik SQL.

MySQL deluje na principu odjemalec - strežnik, pri čemer lahko strežnik namestimo kot sistem, porazdeljen na več strežnikov. Obstaja veliko število odjemalcev, zbirk ukazov in programskih vmesnikov za dostop do podatkovne baze MySQL.

Omogoča nam shranjevanje podatkov na podatkovnem strežniku v poljubni obliki in izvedbo specifičnih nalog v podatkovni bazi ter izvajanje poizvedb nad podatki [17].

#### 4.5.5. CSS

CSS (angl. Cascading Style Sheets) so predloge, ki določajo izgled spletnih strani. Z njimi določamo pisavo, velikost, barvo ter vizualno predstavitev spletne strani. HTML naj bi predstavljal hierarhijo dokumenta, CSS pa predstavitevno vlogo.

CSS podpira danes večina spletnih brskalnikov. Spletni brskalniki, ki standarda CSS ne podpirajo, bodo prikazali običajen HTML-dokument brez oblikovanja.

CSS je bil izdelan z namenom, ločiti vsebino dokumenta (napisanega v HTML-ju ali katerem izmed drugih označevalnih jezikov) od izgleda dokumenta (za katerega skrbi CSS). Z uporabo CSS-a lažje spreminjamo izgled strani, stran je bolj fleksibilna in prilagodljiva. Zmanjša se tudi zahtevnost strani, odpravijo se ponavljanja (npr. neprestano oblikovaje naslovov, saj lahko izgled naslovov določimo v predlogi CSS). Za spremembo izgleda strani ni potrebno spreminjati HTML-ja, ampak samo dopolnimo ali popravimo CSS dokument. S CSS-jem si pomagamo tudi pri predstavitvi iste strani v različnih oblikah (za branje, tiskanje, itd). Ko uporabnik prebira vsebino strani, je stran prilagojena uporabniku, da jo ta lažje razume. Uporabnik pri branju strani uporablja določene povezave (gumbe), ki ga usmerjajo po dokumentu na strani. Določeni deli teksta in gumbi so posebej označeni, da jih uporabnik lažje najde in uporablja. Posebej označenih gumbov in teksta pa ne potrebujemo, ko želimo stran natisniti. Za odstranitev vseh barvnih gumbov, ki na papirju ne bi bili v pomoč, skrbi CSS, ki narekuje, kako brskalnik prikaže stran uporabniku in kako tiskalniku.

Problemi, ki so jih povzročili brskalniki z lastnimi implementacijami HTML-ja, se pojavljajo tudi pri njihovem razumevanju CSS standarda. Tako imajo razvijalci spletnih strani dodatno delo, da njihova stran izgleda enako v različnih brskalnikih. V preteklosti je bilo največje problemov z brskalnikom Mozilla Firefox in Internet Explorer, vendar se z vsako novo verzijo brskalnikov stanje izboljšuje [18].

## 5. IMPLEMENTACIJA KOMENTARJEV PRI SKAVT AROVNIKU

Implementacija komentarjev in prikaz na spletni strani je bila moja naloga pri tem projektu. Projekta implementacije komentarjev sem se lotil decembra 2010 in ga dokončal do maja 2011. Moje delo je opisano v podpoglavju 5.

Za implementacijo Skavt AROVNIKA smo izbrali programsko orodje CodeIgniter.

### 5.1. CodeIgniter



Slika 5.1 Logotip programskega orodja CodeIgniter [19].

CodeIgniter je odprtokodno spletno programsko orodje (angl. Framework), ki omogoča pisanje programov v PHP jeziku. Razvilo ga je podjetje ElisLab. Prva različica CodeIgniterja je izšla 28. februarja 2006, danes je na voljo različica 2.1.

#### Zakaj smo izbrali Codeigniter:

- uporablja MVC arhitekturo,
- preprost za uporabo,
- dokumentacija je predstavljena na preprost način z dobrimi primeri,
- zelo prilagodljiv in razširljiv, ne vsiljuje lastnih konceptov,
- »lahak«, ampak zelo zmogljiv in hiter (hitrejši od ostalih spletnih programskih orodij),
- velika in aktivna skupnost programerjev, ki z veseljem prisloji na pomoč (forum, wiki),
- obstaja veliko uporabnih knjižnic, ki so jih napisali uporabniki,
- veliko programske kode je že razvite,
- ne zaseda veliko pomnilniškega prostora.

Za delo s podatkovno bazo Codeigniter uporablja razred ActiveRecord.

Podprte platforme za delo s podatkovno bazo:

- MySQL, MySQLLi,
- PostgreSQL,

- Oracle,
- SQLite in
- ODBC (angl. Open DataBase Connectivity pomeni odprto povezovanje podatkovne baze).

### 5.1.1. MVC arhitektura

Codeigniter uporablja **MVC** (M - Model, V – pogled (angl. View), C – krmilnik (angl. Controller)) arhitekturo. To je na rtovalski vzorec, ki lo i poslovno logiko od pogleda oz. prikaza (slika 5.2).

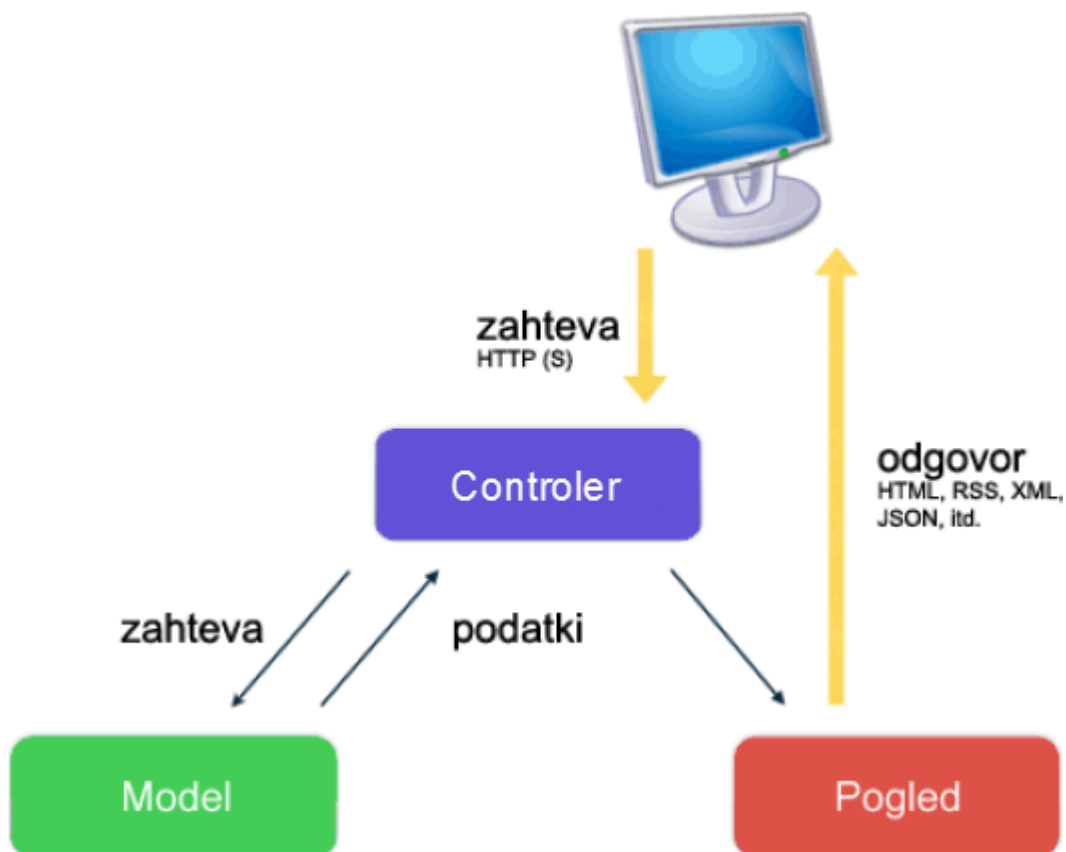
**Model** (glavni del aplikacije) vsebuje poslovno logiko. Vsebuje kodo, ki deluje na podatkih. Pri spletnih aplikacijah bi lahko rekli, da se v modelu odvije ve ina operacij s podatkovno bazo.

**Pogled** je vizualna predstavitev modela, primerna za interakcijo z uporabniki.

**Krmilnik** je vmesnik, ki povezuje model in pogled. Pri spletnih programskih orodjih obravnava HTTP (angl. HyperText Transfer Protocol pomeni protokol za prenašanje hiperteksta) zahteve, glede na te zahteve se kli ejo ustrezne metode modela in nalagajo pogledi.

Prednost MVC arhitekture:

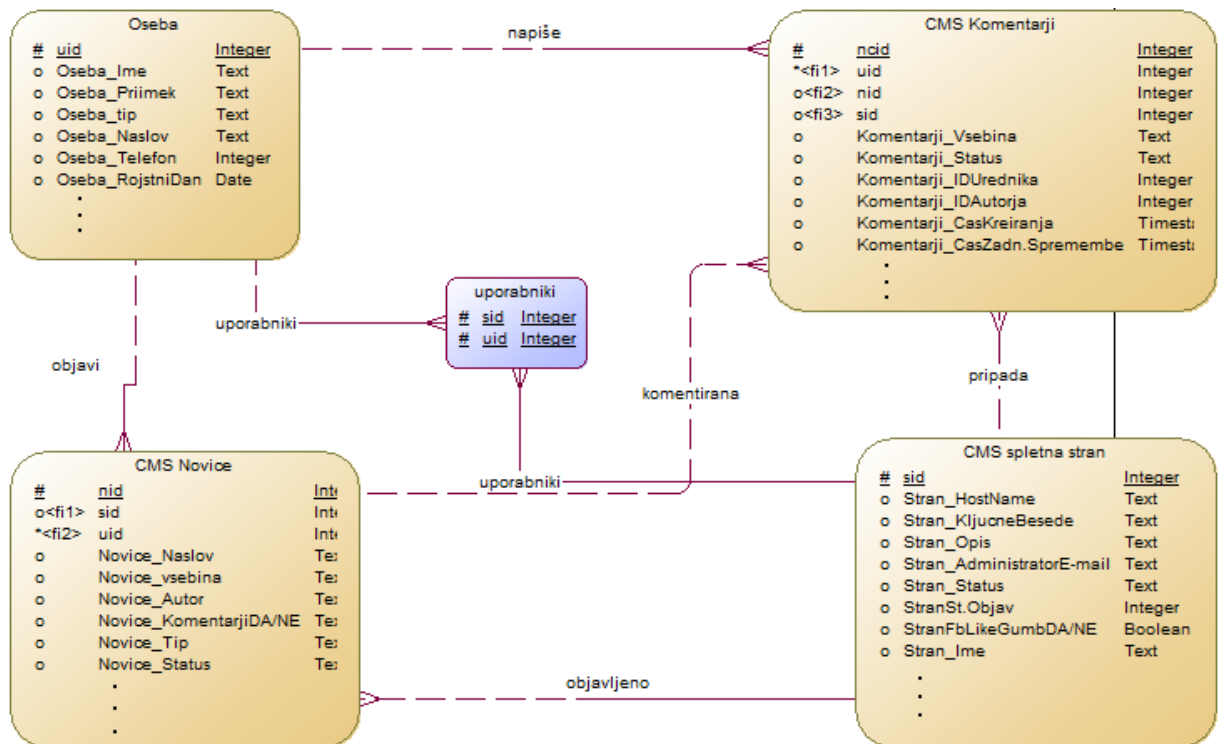
Strogo lo evanje modela in pogleda je vitalnega pomena, saj smo pri spletnih aplikacijah pri a pogostim vizualnim spremembam uporabniškega vmesnika. e je le-ta strogo lo en od modela, je z njegovim spreminjanjem nemogo e pokvariti model, kar izboljšpa robustnost aplikacije [20].



Slika 5.2 MVC arhitektura [21].

## 5.2. Na rtovanje komentarjev

Preden sem začel z implementacijo komentarjev, sem s programom za na rtovanje informacijskih sistemov PowerDesigner naredil na rt. Najprej sem izdelal konceptualni podatkovni model (določil entitete, njihove attribute in razmerja (povezave/relacije) med njimi, kandidiral za ključne in primarne ključne, števnost (kardinalnost) razmerja) in ga pretvoril v logični model (relacijski podatkovni model). Dobil sem Model podatkov, ki upošteva ciljno podatkovno bazo. Logični podatkovni model je predstavljen na sliki 5.3.


































































Slika 5.3 Relacijski podatkovni model pretvorjen iz konceptualnega modela.

### 5.2.1. Na rtovanje podatkovne baze

V podatkovni bazi sem kreiral ime tabele `cms_news_comments`. Znotraj te tabele sem določil attribute, ki so prikazani na sliki 5.4:

- **ncid** (primarni ključ komentarja),
- **nid** (tuji ključ novice),
- **sid** (tuji ključ spletne strani),
- **content** (vsebina komentarja),
- **status** (status komentarja – komentar ima lahko status aktiviran (angl. active) ali pa izbrisan (angl. deleted)),
- **last\_edited\_by** (šifra urednika spletne strani),
- **created\_by** (šifra uporabnika, ki je avtor komentarja),
- **created** (čas kreiranja komentarja),
- **timestamp** (čas zadnje spremembe komentarja).

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> <b>ncid</b>	int(11)			No	None	auto_increment	      
<input type="checkbox"/> <b>nid</b>	int(11)			No	None		      
<input type="checkbox"/> <b>sid</b>	int(11)			No	None		      
<input type="checkbox"/> <b>content</b>	text	utf8_general_ci		No	None		      
<input type="checkbox"/> <b>status</b>	enum('deleted','published','temp')	utf8_general_ci		No	temp		      
<input type="checkbox"/> <b>last_edited_by</b>	int(11)			No	0		      
<input type="checkbox"/> <b>created_by</b>	varchar(255)	utf8_slovenian_ci		No	None		      
<input type="checkbox"/> <b>created</b>	int(11)			No	0		      
<input type="checkbox"/> <b>timestamp</b>	int(11)			No	0		      

Slika 5.4 Tabela cms\_news\_comments v podatkovni bazi.

## 5.3. Implementacija in opis kode za komentarje

Implementacija komentarjev obsega razred cms\_comments\_model (model), news\_comments (pogled), frontend (kontroler) in razred Elements. Datoteko routes krmilimo s pomočjo kontrolerja, kjer določimo URL naslove.

### 5.3.1. Preverjanje, če je uporabnik urednik spletne strani

#### Programska koda:

```
// knjižnica uporabnik (angl. User library)
function is_site_editor($cookie = FALSE)
{
    return $this->CI->cms_users_model->is_site_editor($this->get_uid($cookie));
}
//Model – cms_users_model
function is_site_editor($uid)
{
    $this->db->where(array('uid' => $uid, 'sid' => $this->config->item('sid')))->limit(1);
    $q = $this->db->get('cms_sites_users');

    if ($q->num_rows()) return TRUE;
    else return FALSE;
}
```

#### Razlaga programske kode:

Ob klicu funkcije (is\_site\_editor) najprej preko piškotka (angl. cookie) dobimo ključ uporabnika. Ključ se funkcija is\_site\_editor(\$uid) v Modelu – cms\_users\_model, kjer preverimo ali ključ spletne strani (sid) pripada uporabniku, ki je definiran kot urednik (uid) spletne strani. Kot rezultat dobimo TRUE (urednik) ali FALSE (ni urednik).

### 5.3.2. Uporabnik je prijavljen/ni prijavljen

#### Programska koda:

```

if ( ! is_loggedin()
{
    //uporabnik ni prijavljen
    $this->CI->template->row_activate('IF_NOT_LOGEDIN', $template);
    $comments_count = $this->CI->cms_comments_model->get_comments_count($nid);
    $this->CI->template->row('COMMENTS_COUNT', $comments_count, $template);
}
else
{
    //uporabnik je prijavljen
    $this->CI->template->row_activate('IF_LOGEDIN', $template);
}

```

#### Razlaga programske kode:

e je uporabnik prijavljen na spletno stran, se pod novico izpišejo komentarji. Vsak komentar vsebuje sliko uporabnika, ime uporabnika, datum in uro objave in vsebino komentarja. V primeru, da uporabnik ni prijavljen, dobimo št. komentarjev, ki jih novica vsebuje in poziv, da se naj uporabnik prijavi, e želi videti komentarje. (Primer izpisa: Ta objava ima 3 komentarje. Komentarji so vidni le prijavljenim uporabnikom.)

### 5.3.3. Preverjanje možnosti komentiranja novice

#### Programska koda:

```

if (isset($news["comments"]) && $news["comments"] == "users")
{
    $this->CI->template->row_activate('IF_COMMENTS_ENABLED', $template);
}
else
{
    $this->CI->template->row_activate('IF_NO_COMMENTS_ENABLED', $template);
}

```

#### Razlaga programske kode:

Pri pisanju novice moramo ozna it »Dovoli komentarje?«, e želimo, da je možno novico komentirati (v podatkovno bazo se zapiše "users"). Implementacija najprej preveri, e ima

`$news["comments"]` nastavljeno vrednost in je vrednost v podatkovni bazi enaka "users". Če sta pogoja izpolnjena, pomeni, da so komentarji omogočeni, v nasprotnem so onemogočeni.

### 5.3.4. Vstavljanje komentarjev v podatkovno bazo

#### Programska koda:

```
//klic funkcije insert, ki vsebuje argument polje (ključ in vsebino komentarja)
$data = array(
    'nid' => $this->input->post('nid'),
    'content' => $this->input->post('content')
);
$this->load->model('cms_comments_model');
$this->cms_comments_model->insert($data);

//funkcije insert v Modelu - cms_users_model
function insert($data = array())
{
    $timestamp = time();
    $data['sid'] = $this->config->item('sid');
    $data['status'] = 'published';
    $data['created_by'] = $this->users->uid;
    $data['created'] = $timestamp;
    $data['timestamp'] = $timestamp;
    $this->db->insert('cms_news_comments', $data);
}
```

#### Razlaga programske kode:

Funkcija za vpis komentarjev `insert($data = array())` vpiše v podatkovno bazo ključ (nid), vsebino (content), status (status), avtorja ('created\_by'), čas kreiranja (created), čas zadnje spremembe komentarja (timestamp) in ključ spletne strani (sid).

### 5.3.5. Izpis vseh komentarjev pod določeno novico

#### Programska koda:

```
function get_comments($nid = FALSE)
{
    if ( ! $nid)
        return array();
    $this->db->where('nid', $nid);
```

```

$this->db->where('status', 'published');
$this->db->where('sid', $this->config->item('sid'));
$this->db->order_by('created', 'DESC');
$q = $this->db->get('cms_news_comments');

if ($q->num_rows())
    return $q->result_array();
else
    return array();
}

```

### Razlaga programske kode:

Funkcija za izpis vseh komentarjev `get_comments($nid = FALSE)` pod novico potrebuje kot argument šifro novice (`nid`). Če šifra novice obstaja, potem pod novico izpišemo vse padajoče razvršene komentarje (glede na čas kreiranja). Rezultat dobimo v obliki seznama komentarjev.

### 5.3.6. Brisanje komentarjev

#### Programska koda:

```

function delete_comment($ncid)
{
    if (is_site_editor() && $ncid)
    {
        $this->load->model('cms_comments_model');
        $this->cms_comments_model->update($ncid, array('status' => 'deleted'));
    }

    if (isset($_SERVER["HTTP_REFERER"]))
        redirect($_SERVER["HTTP_REFERER"]);
    else
        redirect("");
}

```

#### Razlaga programske kode:

Možnost brisanja komentarja ima samo urednik (administrator) spletne strani, vsi ostali uporabniki te možnosti nimajo. V primeru, da uporabnik doda komentar pod določeno novico ali pa prijavi neprimeren komentar, dobi urednik e-pošto s povezavo do novice, kjer se nahaja komentar. Če želi urednik ta komentar izbrisati, najprej program preveri, če je uporabnik res

urednik (is\_site\_editor) spletne strani in e komentar (ncid) obstaja. V primeru, da sta pogoja izpolnjena, se kli e funkcija »update« v Modelu, kjer se v podatkovni bazi spremeni status komentarja na »deleted«. Posledi no se komentar ne prikaže pod novico. Po kon ani operaciji nas program vrne (angl. redirect) eno stran nazaj (angl. one page back).

### 5.3.7. Prijava neprimerne komentarja

#### Programska koda:

```
if ($this->uri->segment(3) == 'confirmYes')
{
    $this->load->model('cms_comments_model');
    $ncid = $this->uri->segment(2);
    $comment = $this->cms_comments_model->get_single_comment($ncid);
    $user = $this->users->get_info();
    // sestavljeno sporo ilo
    $gender_add = $user['sex'] == 'm'? '': 'a';
    $message = $user['name'] . " " . $user['surname'] . " je ozna il$gender_add spodnji
komentar kot neprimeren!\n\n"
. "Vsebina komentarja:\n"
. $comment['content'] . "\n"
. "-----\n\n"
. "Povezava do novice: $referer\n"
. "Povezava do komentarja: $referer#comment$ncid\n\n"
. " e se ti zdi komentar primeren in ga ni potrebno brisati preprosto ignoriraj to
sporo ilo.\n\n"
. "-----\n\n"
. "Na ta email ne odgovarjaj, ker odgovora ne bo dobil nih e!";

    // pošiljanje sporo ila uredniku spletne strani
    $this->load->library('email');
    $this->email->from('ne-odgovarjaj@skavt.net');
    $this->email->to($this->config->item('admin_email'));
    $this->email->subject('Prijava neprimerne komentarja');
    $this->email->message($message);

    $this->email->send();
}
```

#### Razlaga programske kode:

V primeru, da uporabnik meni, da je komentar neprimeren, klikne na povezavo ob komentarju in prijavi neprimeren komentar. Pojavi se okno, ki želi potrditev. Prvi if stavek preveri tretji segment URL-ja (Primer URL-ja: `andrej.cms.dev.skavt.net/flag_comment/27/'confirmYes'`). V primeru, da nam if stavek vrne »TRUE«, pošljemo uredniku e-pošto, v katerem se nahajajo naslednji podatki: ime in priimek osebe, ki je prijavila neprimeren komentar, vsebina komentarja, povezava do novice in neprimernega komentarja. Urednik ima možnost izbrisati komentar ali pa prezreti sporo ilo. Primer poslane e-pošte uredniki vidimo na sliki 5.5.

### Prijava neprimernega komentarja

📧 | X Prejeto | X

☆ **ne-odgovarjaj@skavt.net** Za andrej.strajhar

Andrej Štrajhar je označil spodnji komentar kot neprimeren!

Vsebina komentarja:  
test

-----

Povezava do novice:  
[http://andrej.cms.dev.skavt.net/test\\_komentarji\\_nova\\_novica](http://andrej.cms.dev.skavt.net/test_komentarji_nova_novica)  
Povezava do komentarja:  
[http://andrej.cms.dev.skavt.net/test\\_komentarji\\_nova\\_novica#comment47](http://andrej.cms.dev.skavt.net/test_komentarji_nova_novica#comment47)

Če se ti zdi, da komentarja ni potrebno brisati, preprosto ignoriraj to sporočilo.

-----

Na ta email ne odgovarjaj, ker odgovora ne bo dobil nihče!

Slika 5.5 Primer poslane e-pošte uredniku za neprimeren komentar.

## 6. ZAKLJUČEK

Namen razvoja Skavt AROVNIKA je bil ponuditi možnost izdelave spletne strani uporabnikom, ki nimajo programerskega znanja.

V ekipi, s katero smo se lotili problema, je bila moja naloga implementacija komentiranja novic na spletni strani. Pri samem delu sem imel na začetku težave s spoznavanjem novega spletnega programskega orodja CodeIgniter. Zelo pomembno je bilo, da smo se na začetku kodiranja uspešno uskladili glede pravil pisanja programske kode, tako je lažje iskanje po programski kodi in seveda na dolgi rok bo lažje kodo vzdrževati.

Implementacija komentarjev vsebuje 484 vrstic programske kode.

Možnost razširitve sistema je še veliko, saj sistem podpirale osnovne funkcionalnosti. Glavni ideji za izboljšanje sta bili RSS novice in dodajanje lastnih predlog. Pri RSS novicah imam v mislih prenašanje novic iz sistema na lokalni disk, prenašanje novic med spletnimi stranmi in obveščanje o novostih na spletnem mestu. Trenutno ima urednik možnost izbire med tremi prednaloženimi izgledi spletne strani. Radi bi omogočili, da bi urednik lahko sam nalagal nove predloge.

Za konec prilagam nekaj komentarjev naših uporabnikov Skavt AROVNIKA.

Uporabnik 1:

BRAVO!! ESTITAM!

arovník ja prav dobra FORA!!! :)

Uporabnik 2:

...moram povedat, da ste razvili res dober CMS (BRAVO BRAVO)...

Uporabnik 3:

Hvala za vaš trud in podporo, ki jo nudite! :)

Uporabnik 4:

Možnost, da se v galerijo doda več slik istočasno (ne vsako posebej), bi bila super izboljšava!

## 7. VIRI

- [1] mag. Radoš Skrt: Odprtokodni sistemi za samostojno upravljanje spletnih vsebin. Moj mikro, februar 2006
- [2] (2012) WordPress. Dostopno na:  
<http://en.wikipedia.org/wiki/WordPress>
- [3] (2011) Comparing the Numbers: WordPress, Joomla, Drupal. Dostopno na:  
<http://www.resourcenation.com/blog/comparing-the-numbers-wordpress-joomla-drupal/31988/>
- [4] (2012) Drupal. Dostopno na:  
<http://en.wikipedia.org/wiki/Drupal>
- [5] (2011) About Drupal – History. Dostopno na:  
<http://drupal.org/about/history>
- [6] (2011) Drupal CMS sistem urejanja vsebin. Dostopno na:  
<http://hauki.si/drupal-cms-sistem-urejanja-vsebin>
- [7] (2012) Joomla. Dostopna na:  
<http://en.wikipedia.org/wiki/Joomla>
- [8] (2011) Kamniški skavti - Mamutov steg - Kamnik 1 - doma a stran. Dostopno na:  
<http://kamnik1.skavt.net/index.php>
- [9] (2011) Kamniški skavti - Mamutov steg - spletni dom - Administration. Dostopno na:  
<http://kamnik1.skavt.net/administrator/index.php>
- [10] (2011) Skavt AROVNIK – SkavtNET. Dostopno na:  
<http://www.skavt.net/carovnik>
- [11] (2012) Slovenska skavtska organizacija | Slovenske skavtinje in skavti - Slovenian guides and scouts. Dostopno na:  
<http://skavti.si/>
- [12] (2012) Pregled strani | Slovenske skavtinje in skavti - Slovenian guides and scouts. Dostopno na:  
<http://skavti.si/admin>
- [13] (2012) Programski jezik HTML | Pomagalnik. Dostopno na:  
<http://www.pomagalnik.com/slovar/programski-jezik-html/>
- [14] (2012) HTML - Wikipedia, the free encyclopedia. Dostopno na:  
<http://en.wikipedia.org/wiki/HTML>
- [15] (2011) JavaScript. Dostopno na:  
<http://zaversnik.fmf.uni-lj.si/gradiva/javascript/>
- [16] (2012) PHP - Wikipedia, the free encyclopedia. Dostopno na:  
<http://en.wikipedia.org/wiki/PHP>
- [17] (2011) MySQL. Dostopno na:

- [http://gradiva.txt.si/racunalnistvo/racunalniska-omrezja/70\\_strezniki/mysql/](http://gradiva.txt.si/racunalnistvo/racunalniska-omrezja/70_strezniki/mysql/)
- [18] (2012) CSS - Wikipedija, prosta enciklopedija. Dostopno na:  
<http://sl.wikipedia.org/wiki/CSS>
- [19] (2011) Welcome to CodeIgniter : CodeIgniter User Guide. Dostopno na:  
[http://codeigniter.com/user\\_guide/](http://codeigniter.com/user_guide/)
- [20] (2011) DKUM - OGRODJE ZA IZDELAVO SPLETNIH STRANI V PHP. Dostopno na:  
<http://dkum.uni-mb.si/IzpisGradiva.php?id=20631>
- [21] (2011) DKUM - DOBRE PRAKSE RAZVOJA SODOBNIH ASP.NET SPLETNIH APLIKACIJ. Dostopno na:  
<http://dkum.uni-mb.si/IzpisGradiva.php?id=20605>
- [22] (2009) SISTEM ZA UPRAVLJANJE SPLETNIH VSEBIN. Dostopna na:  
[http://eprints.fri.uni-lj.si/846/1/Pucelj\\_M\\_UN.pdf](http://eprints.fri.uni-lj.si/846/1/Pucelj_M_UN.pdf)