

Západočeská univerzita v Plzni

FAKULTA PEDAGOGICKÁ

BAKALÁŘSKÁ PRÁCE

REDAKČNÍ SYSTÉM VĚDECKÉHO ČASOPISU

Adam Janča

Plzeň 2012

Prohlašuji, že jsem práci vypracoval samostatně s použitím uvedené literatury a zdrojů informací.

V Plzni, 28. června 2012

.....

OBSAH

Úvod	1
1 SYSTÉM REDAKCE VĚDECKÉHO ČASOPISU ARNICA	2
1.1 ROLE V REDAKCI	2
1.1.1 Autor.....	2
1.1.2 Oponent.....	2
1.1.3 Šéfredaktor	3
1.2 ŽIVOTNÍ CYKLUS ČLÁNKU	3
1.2.1 Odevzdání článku.....	3
1.2.2 Přiřazení oponentů a napsání posudku	4
1.2.3 Schválení článku	4
1.2.4 Odevzdání nové verze článku	4
1.2.5 Publikace článku	5
2 REDAKČNÍ SYSTÉM JOOMLA! A JEHO POUŽITÍ	6
2.1 POPIS REDAKČNÍHO SYSTÉMU	6
2.1.1 Specifika redakčních systémů.....	6
2.1.2 Systémy CMS	7
2.1.3 Progresivní růst systému Joomla!.....	8
2.2 PŘIZPŮSOBENÍ A NASTAVENÍ	8
2.2.1 Instalace.....	8
2.2.2 Vytvoření článků webové prezentace	10
2.2.3 Nastavení nabídek	11
2.2.4 Jazykové mutace.....	13
2.2.5 Další použité prvky	15
3 NÁVRH VLASTNÍHO ŘEŠENÍ PRO SYSTÉM REDAKCE.....	16
3.1 PRINCIPY TVORBY ROZŠÍŘENÍ.....	16
3.1.1 Rozdělení rozšíření	16
3.1.2 Struktura komponenty	17
3.1.3 Zabudované funkce a třídy Joomla!	18
3.1.4 Předávání požadavků uvnitř komponenty.....	19
3.2 VLASTNÍ TVORBA KOMPONENTY	20
3.2.1 Databázová struktura komponenty.....	20
3.2.2 Ukládání článku	22
3.2.3 Vypisování článků	24
3.2.4 Přiřazení oponenta pro článek	25
3.2.5 Nahrání posudků pro článek.....	27
3.2.6 Schvalování článků.....	28
3.2.7 Publikování článku	29
4 VÝSLEDNÝ PRODUKT A JEHO POPIS	31
4.1 VYUŽITÍ KOMPONENTY Z POHLEDU AUTORA.....	31
4.1.1 Odevzdání nového článku	31
4.1.2 Odevzdání nové verze článku	32
4.1.3 Přehled článků	32
4.2 VYUŽITÍ KOMPONENTY Z POHLEDU OPONENTA.....	33
4.2.1 Odevzdání posudku	33
4.2.2 Hlasování redakční rady	33
4.2.3 Přehled posuzovaných článků a vlastních posudků.....	34

4.3 VYUŽITÍ KOMPONENTY Z POHLEDU ŠÉFREDAKTORA	34
4.3.1 Přiřazení oponentů	34
4.3.2 Schvalování článků.....	35
4.3.3 Publikování článků	36
ZÁVĚR.....	37
SEZNAM OBRÁZKŮ A TABULEK	38
SEZNAM OBRÁZKŮ	38
SEZNAM TABULEK	38
SEZNAM LITERATURY	39
RESUMÉ	40
PŘÍLOHY	I
A. RELACE TABULEK KOMPONENTY COM_CLANKY.....	I
B. KOMPLETNÍ ADRESÁŘOVÁ STRUKTURA KOMPONENTY COM_CLANKY.....	II

Úvod

Za původem každého tištěného či elektronického časopisu stojí redakce časopisu. Ta zodpovídá za kvalitu článků, které se objevují v jejich publikacích. Aby kvalita připravovaných článků dosahovala určité úrovně, využívá redakce vnitřních metod a postupů, během kterých se odstraňují možné nedostatky, jenž články mohou mít. Cílem bakalářské práce bylo analyzovat tyto principy u redakce vědeckého časopisu ARNICA a na jejich základech vytvořit webovou aplikaci usnadňující práci v redakci.

První kapitola bakalářské práce je zaměřena na rozbor systému redakce z pohledu nutného pro vyrvoření webové aplikace. Analýza se proto zabývá fungováním redakce časopisu ARNICA a zmapováním trasy článku, kterou musí absolvovat pro zdárnou publikaci.

Pro webovou prezentaci časopisu ARNICA byl zvolen jako hlavní nástroj redakční systém Joomla!. Tento systém značně ulehčuje a urychluje tvorbu internetových stránek. Další kapitola bakalářské práce představuje možnosti nabízené systémem Joomla!, především takové, které se dají použít v případě tvorby webu časopisu ARNICA.

Na základě rozboru vnitřních principů redakce a získaných znalostí o systému Joomla! se práce zaměřuje hlavní částí na vytvoření webové aplikace v podobě rozšíření systému Joomla!. Postup tvorby rozšíření a napojení na systém Joomla! je popsán ve třetí kapitole. Cílem tohoto rozšíření je usnadnit členům redakce proces odevzdávání, hodnocení a opravování článků.

Poslední kapitola je věnována popisu vytvářeného rozšíření z uživatelského pohledu.

1 SYSTÉM REDAKCE VĚDECKÉHO ČASOPISU ARNICA

K vytvoření funkční aplikace pro správu článků časopisu ARNICA, je nejprve nutno podrobně poznat a pochopit chod redakce tohoto časopisu.

Na procesy v redakci se můžeme dívat z více pohledů. Pro vývoj programu či aplikace jsou podstatné následující dva pohledy. Prvním pohledem je zastoupení rolí v redakci a tím druhým je životní cyklus článku.

1.1 ROLE V REDAKCI

Členy redakce můžeme rozdělit do několika typů, podle toho jaké úkony provádějí. Tyto typy můžeme nazvat jako role v redakci. Jedná se o role autora, oponenta a šéfredaktora. Každá role je jedinečná a pro chod redakce nepostradatelná.

Ve skutečnosti to nemusí vypadat tak, že jedna reálná osoba může vykonávat pouze jednu roli, ale může jich vykonávat i více. V praxi tedy můžeme mít člena redakce, který je autorem, ale zároveň i šéfredaktorem. Na druhou stranu můžeme mít jiného člena redakce, který je pouze oponentem.

1.1.1 AUTOR

Role autora spočívá v psaní nových, dosud nepublikovaných, článků do časopisu. Autor zodpovídá za kvalitu článku a tedy, pokud je potřeba, článek upravuje, tak aby byl korektní a mohl být schválen.

Časopis ARNICA je zaměřen především na tvorbu psanou v českém, slovenském nebo anglickém jazyce. Články, které autor píše, jsou specifikovány i dalšími podmínkami. Tyto podmínky se vztahují k obsahu. Jedná se například o rozsahy klíčových slov, abstraktů či hlavního článku, dále o uvedení seznamu použité literatury v předem stanoveném tvaru a další.

1.1.2 OPONENT

Oponentova činnost spočívá v přečtení a prostudování jemu přidělených článků a posouzení jejich kvalit. V posouzení článků oponent upozorňuje na nedostatky a nepřesnosti článku, tak aby autor mohl snáze opravit vady, které článek obsahuje. Oponent samozřejmě také předkládá i pozitiva posuzovaného článku. Celkové shrnutí posouzení článku se kategorizuje na kladný nebo záporný posudek.

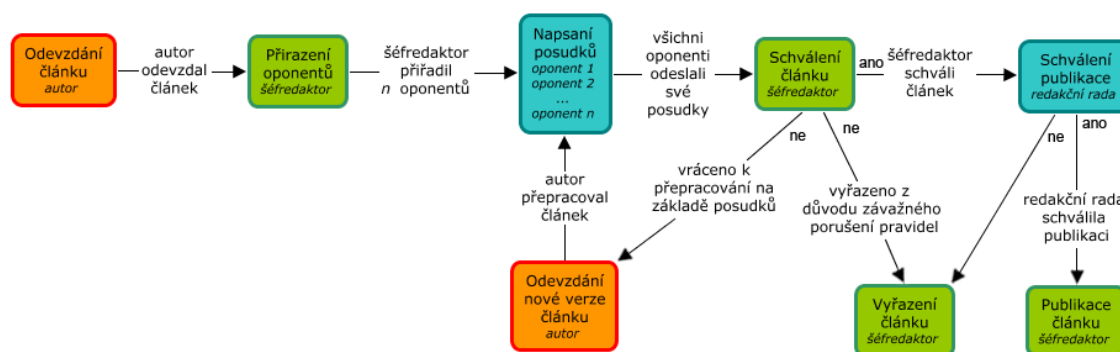
1.1.3 ŠÉFREDAKTOR

Šéfredaktor, jak již název napovídá, zastřešuje celou redakci a zodpovídá za její chod. Dále spojuje kroky autora a oponenta. V první řadě šéfredaktor přiděluje k napsaným článkům oponenty, kteří se budou danými články zabývat. Úkolem šéfredaktora je vybrat takové oponenty, kteří se v problematice napsaného článku dostatečně orientují.

Další část této role se zabývá schvalováním článku. Šéfredaktor si přečte posudky od oponentů k danému článku, a na základě těchto verdiktů rozhodne, zdali je článek napsaný vhodně a správně, či se musí ještě vrátit k přepracování. Článek je přepracováván do té doby, než šéfredaktor na základě posudků rozhodne, že je korektní. Schválené články pak šéfredaktor sestavuje podle potřeby a zaměření jednotlivých čísel časopisu a zasílá je k další publikaci.

1.2 ŽIVOTNÍ CYKLUS ČLÁNKU

Proto, aby byl článek publikován, musí projít cyklem, ve kterém se o něj v konkrétním čase stará aktivně jen jedna z rolí. Schéma tohoto životního cyklu znázorňuje obrázek 1 – Životní cyklus článku. U každého kroku cyklu je uvedena role, která tento krok provádí a mezi jednotlivými kroky jsou popsány akce, které mezi nimi probíhají.



Obrázek 1 – Životní cyklus článku

1.2.1 ODEVZDÁNÍ ČLÁNKU

Prvotním krokem životního cyklu článku je jeho odevzdání. Když tedy autor napíše článek, se kterým je spokojen a o kterém si myslí, že je vhodný k publikaci do toho časopisu, odevzdá ho. Odevzdávání může probíhat například prostřednictvím emailu, kde

se soubor s článkem přiloží do přílohy. Dalším způsobem je odevzdání na elektronickém médii jako je CD a DVD. Disky je možno zaslat poštou. Odevzdávaný článek je přijímán ve formátech „.doc“, „.rtf“ nebo po konzultaci s redakcí ve formátu „.tex“. Vyžaduje se písmo Times New Roman velikosti 12 s řádkováním 2 a odstavce by měli být zarovnány vlevo. Text nesmí obsahovat dělená slova.

1.2.2 PŘIŘAZENÍ OPONENTŮ A NAPSÁNÍ POSUDKU

Jakmile je článek odevzdán autorem, přejímá iniciativu šéfredaktor. Ten má nyní za úkol vybrat vhodné oponenty pro článek. Šéfredaktor vybírá oponenty podle jejich znalostí, tak aby vyhovovali pro daný článek a vyznali se v problematice, kterou se článek zabývá.

V dalším kroku mají oponenti za úkol prostudovat přiřazený článek a napsat na něj posudek. Oponenti v posudcích sledují několik složek práce. Jedná se například o typografii, jazykové zásady a celkovou jazykovou kulturu, autorská práva nebo odbornost článku a další. Oponenti by také měli sledovat, zdali článek již nebyl dříve někde publikován a jestli se tedy opravdu jedná o původní práci. Tyto posudky oponentů jsou dále odevzdávány spolu se článkem opět šéfredaktorovi.

1.2.3 SCHVÁLENÍ ČLÁNKU

Šéfredaktor si projde posudky oponentů, popřípadě sám nahlédne do článku a na základě přiložených verdiktů se rozhodne o dalším vývoji článku. Ten se může ubírat prakticky dvěma cestami. Při kladném rozhodnutí šéfredaktora je článek schválen čímž autorova práce končí. V opačném případě je vrácen autorovi k přepracování.

1.2.4 ODEVZDÁNÍ NOVÉ VERZE ČLÁNKU

Jestliže šéfredaktor rozhodne, že článek není dostatečně kvalitní ke schválení, je vrácen autorovi k přepracování. Jako vodítka pro přepracování článku autorovi slouží posudky oponentů, či popřípadě komentář šéfredaktora.

Poté co autor odevzdá již upravený článek, šéfredaktor už nepřiděluje nové oponenty, ale článek se vrací k oponentům, kteří byli přiděleni při prvním odevzdání. Oponenti napíší další posudky na novou verzi článku a článek je odeslán zpět k šéfredaktorovi. Ten znovu na základě posudků rozhodne, zdali je tentokrát článek

v pořádku, anebo je potřeba jej znovu přepracovat. Článek takto putuje do té doby, než je schválen, nebo je zamítnut z důvodů závažných prohřešků.

1.2.5 PUBLIKACE ČLÁNKU

Pokud je článek kvalitně napsán, je schválen a čeká tedy na svůj poslední krok – publikaci. K té se přikročí rozhodnutím šéfredaktora, které je založeno na základě hodnocení článku redakční radou složené ze všech oponentů. Rada stručnou formou (ano/ne) vyjadřuje souhlas s publikací článku. Jakmile se vyjádří dostatečný počet členů rady kladně, může šéfredaktor článek publikovat. Zdali se publikovaný článek použije do připraveného čísla, může šéfredaktor rozhodnout podle více aspektů. Těmi hlavními jsou např. téma a rozsah. V procesu publikace článku se může dále měnit grafický design článku, nicméně obsah by měl zůstat nezměněn, tak jak prošel redakcí.

2 REDAKČNÍ SYSTÉM JOOMLA! A JEHO POUŽITÍ

Pro internetové stránky redakce vědeckého časopisu ARNICA byl zvolen redakční systém Joomla!. Systém Joomla! je moduly, komponentami a pluginy rozšiřitelný systém pro správu obsahu internetových stránek. S pomocí nástrojů zabudovaných v redakčním systému a několika dostupných rozšíření jsme schopni vytvořit plnohodnotnou webovou prezentaci, a to bez předchozích znalostí jazyka HTML či programovacích jazyků jako je PHP či JavaScript.

2.1 POPIS REDAKČNÍHO SYSTÉMU

Hlavním charakteristickým rysem redakčního systému Joomla! je zjednodušení procesu tvorby internetových stránek. Joomla! nabízí řadu efektivních nástrojů nejen pro urychlení tvorby, ale také pro administraci webu. Téměř každá část redakčního systému, jak uživatelského, tak administrátorského rozhraní spravující obsah, je vybavena WYSIWYG¹ editorem, díky kterému uživatelé a administrátoři nemusí používat značky jazyka HTML, ale mohou pracovat s obsahem pomocí toho grafického prostředí. Výhodou systému Joomla! tedy je, že jeho prostřednictvím mohou vytvářet webové prezentace jak zkušení kodéři, tak i uživatelé, kteří předchozí zkušenosti nemají. Pokud tvůrce internetových stránek chce rozšířit web o další funkce, které systém Joomla! v základní verzi nenabízí, může si stáhnout některé z dostupných rozšíření, které budou danou funkci vykonávat a jednoduše je nainstalovat. V případě, že rozšíření plně nevyhovuje potřebám tvůrce webu, anebo rozšíření plnicí požadovanou funkci neexistuje, může si jej tvůrce upravit, nebo vytvořit zcela nové rozšíření. Tvorba rozšíření je samozřejmě náročnější a většinou vyžaduje znalost programovacích jazyků. Systém Joomla! ovšem poskytuje řadu vestavěných PHP tříd a funkcí, která nám tuto tvorbu mohou ulehčit.

2.1.1 SPECIFIKA REDAKČNÍCH SYSTÉMŮ

U standardních internetových stránek nevyužívajících redakčních systémů obecně platí, že si veškerý obsah a rozšíření musíme vytvořit sami. Pokud tedy např. chceme na námi tvořený web umístit galerii fotografií, musíme naprogramovat příslušné skripty pro její obsluhu, uživatelské rozhraní a další prvky, např. nahrávání obrázků přes webové

¹ WYSIWYG – Zkratka z anglických slov „What You See Is What You Get“, česky můžeme přeložit jako „Co vidíš, je to co dostaneš.“

rozhraní. Tato práce není vždy snadná a často bývá časově náročná. Při tvorbě webových stránek s pomocí redakčního systému Joomla! je hlavní výhodou používání již vytvořených částí, které mají univerzální využití nehledě na konkrétní zaměření či stylizování stránek. Když bychom tedy chtěli do našich internetových stránek začlenit galerii, stačí si vybrat z několika možných dostupných rozšíření pro Joomla! a toto rozšíření nainstalovat do redakčního systému. Galerii po instalaci nastavíme dle našich potřeb a následně ji již můžeme používat na našem webu. Tím nám odpadla práce s programováním galerie samotné a práce se zúžila pouze na konfiguraci.

Internetové stránky se ale neskládají jen z galerií, ale i z běžnějších částí, kterými návštěvníku stránek sdělujeme převážně textové informace. Na tyto části systém Joomla! nahlíží jako na články. Po přihlášení do administrace systému, anebo i po přihlášení přes uživatelské rozhraní internetových stránek, můžeme tyto články snadno editovat. Pro vytvoření a úpravu článku, a tedy i většinu obsahu webu, můžeme využít WYSIWYG editor. Na první pohled je výhodou redakčního systému, oproti běžné tvorbě webu, rychlejší tvorba, nenáročnost na znalost programování a snadná administrace stránek.

2.1.2 SYSTÉMY CMS

Software typu CMS je software určený pro správu obsahu. Do skupiny CMS software patří i systém Joomla!. CMS systémy jsou charakteristické snadností úpravy a vytváření obsahu na takové úrovni, že tuto úpravu dokáží provádět i uživatelé, kteří nedisponují znalostí jazyka HTML či jiných pokročilejších jazyků určených pro tvorbu webu. Software pro správu obsahu Joomla! je designován na velký počet uživatelů a článků, a tak spolu s uživatelskou nenáročností jsou vhodným nástrojem pro Web 2.0, tedy pro internetové stránky, kde se o obsah nestará jen administrátor či webmaster, ale ve větší či menší míře všichni uživatelé, kteří o to mají zájem. Systémy CMS využívají databáze pro uchovávání dat, a tento obsah dále zobrazují prostřednictvím šablon. Výhodou systému Joomla! je jeho snadná administrace, která se provádí pouze přes webové rozhraní. To není samozřejmostí u všech software CMS. Ve skutečnosti starší systémy CMS byly z hlediska administrace a nastavování prezentace uchovaných dat velmi náročné. [1, s. 10 - 13]

2.1.3 PROGRESIVNÍ RŮST SYSTÉMU JOOMLA!

Původ systému Joomla! je spjat s jeho předchůdcem CMS systémem Mambo. Mambo, původně veřejnosti uzavřený systém, vznikl v Austrálii pod firmou Miro Corporation. V roce 2001 byl tento systém uveřejněn jako zdarma dostupný open source pro komerční a nekomerční použití. Byl tedy dostupný i s jeho zdrojovými kódy. Systém se stal rychle populární a dočkal se spousty rozšíření a šablon. S rokem 2005 ale přišly neshody mezi vývojáři samotného systému Mambo a mezi vývojáři jeho volitelných rozšíření. To vedlo až k ukončení oficiálního vývoje softwaru Mambo. [1, s. 9 - 12]

Tam, kde systém Mambo skončil, navázal nový CMS systém Joomla! 1.0. Tato verze systému Joomla! byla v jádru podobná systému Mambo, ale nabídla nové a pohodlnější uživatelské a administrátorské rozhraní. Systém Joomla! si začal získávat ještě více příznivců a vývojářů, než dřívější Mambo. V roce 2009 bylo, pro tehdy aktuální verzi 1.5 systému Joomla!, zaznamenáno 4000 volně dostupných rozšíření. K letošnímu roku (2012) je podle oficiálních stránek systému Joomla! dostupných již přes 9000 rozšíření pro aktuální verzi Joomla! 2.5. Vývojáři systému Joomla! stále pokračují ve zdokonalování svého produktu a v polovině roku 2013 by měl být zveřejněn systém verze 3.5. [1, s. 9 - 12] [2]

2.2 PŘIZPŮSOBENÍ A NASTAVENÍ

Redakční systém Joomla! je pro správný běh nejprve nutno nainstalovat a posléze nakonfigurovat. V případě vědeckého časopisu ARNICA využijeme základních komponent a modulů starajících se o články, nabídky, jazykové mutace. Do webové prezentace začleníme i předem vytvořenou šablonu vzhledu a rozšíření, jako například galerii fotografií. Pomocí těchto prvků lze sestavit značnou část internetových stránek časopisu.

2.2.1 INSTALACE

Redakční systém, stejně jako jakékoli internetové stránky, musí být umístěn na webovém serveru. Tento server musíme vybrat na základě systémových nároků redakčního systému Joomla!. V případě instalace pro časopis ARNICA jsme použili systém verze 2.5. Ten má tyto konkrétní požadavky: Apache 1.3. + PHP 5.2 a novější + MySQL 5.04. Dalším důležitým požadavkem je kapacita diskového a databázového prostoru serveru. Diskový prostor musíme zvolit přiměřeně velký, aby se do něj vešly články

nahrávané redakcí, publikovaná čísla časopisu a také redakční systém samotný. Velikost databáze je také důležitá, neboť se v ní uchovává obsah webové prezentace.

Joomla! 2.5.1 Instalace

Kroky

- 1: Jazyk
- 2: Předinstalační kontrola**
- 3: Licence
- 4: Databáze
- 5: Nastavení FTP
- 6: Nastavení
- 7: Ukončení

Předinstalační kontrola

Předinstalační kontrola pro Joomla! 2.5.1 Stable [Ember] 2-Feb-2012 14:00 GMT:

Pokud není jedna z následujících položek správně podporována (označena **Ne**) proveďte kroky k nápravě. Pokud to neuděláte riskujete, že vaše Joomla! instalace nebude správně fungovat.

Verze PHP >= 5.2.4	Ano
Podpora Zlib komprese	Ano
Podpora XML	Ano
Podpora databáze: (mysql, mysqli)	Ano
MB Language je Výchází	Ano
MB String Overload vypnuto	Ano
Podpora INI Parseru	Ano
Podpora JSON	Ano
configuration.php Zapisovatelné	Ne

I tak můžete pokračovat v instalaci, protože na konci bude konfigurační soubor vypsán. Ručně jej pak musíte uložit do kódu, klikněte do textové oblasti pro výměru kompletního kódu, který vložíte do nového textového souboru. Přejmenujte jej na 'configuration.php' a nahrajte jej do kořenové složky vaší stránky.

Doporučená nastavení:

Tato nastavení PHP jsou doporučena pro zajištění plné kompatibility s Joomla!.

Nicméně i při drobných odchylkách od doporučených hodnot může Joomla! stále fungovat.

Parametr	Doporučeno	Momentálně
Safe Mode	Vypnuto	Vypnuto
Zobrazit chyby	Vypnuto	Vypnuto
Nahrávání souborů	Zapnuto	Zapnuto
Magic Quotes Runtime	Vypnuto	Vypnuto
INI_MAGIC_QUOTES_GPC	Vypnuto	Vypnuto
Register Globals	Vypnuto	Vypnuto
Ukládání výstupu	Vypnuto	Zapnuto
Auto Start sezení	Vypnuto	Vypnuto
Nativní podpora ZIP komprese	Zapnuto	Zapnuto

Joomla!® je svobodný software šířený pod licenci GNU General Public License.

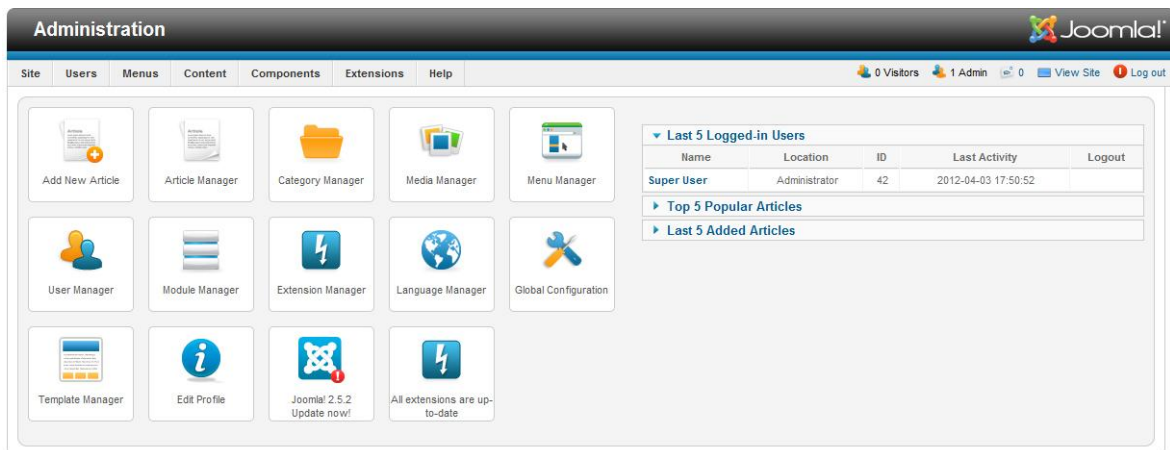
Obrázek 2 – Instalátor systému Joomla!

Po nahrání souborů redakčního systému na hosting je na řadě samotná instalace, jejíž nejnovější verzi nalezneme na oficiálních stránkách www.joomla.org. Proces instalace je řízen přehledným a intuitivním instalačním prostředím, podobně jako u klasické instalace software přímo na PC. Abychom se k instalátoru dostali, stačí zadat do internetového prohlížeče doménu stránek, která odkazuje na hosting. Webový instalátor systému Joomla! můžeme vidět na obrázku 2 – Instalátor systému Joomla!. Instalátor systému tvoří sedm kroků. V prvním kroku si vybíráme jazyk, který nás bude provázet celým instalátorem. Následující krok se jmenuje „Předinstalační kontrola“. V této části nás systém informuje o tom, zdali námi využívaný hosting vyhovuje daným požadavkům. Obrazovka je rozdělena do dvou segmentů, jeden se věnuje podmínkám, které musí být nutně splněny, kdežto druhý se zabývá doporučeným nastavením, které pro chod systému neznamena kritické omezení. Třetí krok je věnován licenci a jejímu odsouhlasení. Systém Joomla! spadá pod licenci GNU. Ve čtvrtém kroku nastavujeme spojení s databází, a tedy vyplňujeme přístupové údaje. Dalším krokem je nepovinné nastavení FTP. Toto nastavení slouží k využití FTP protokolu prostřednictvím systému Joomla!. Je podstatné zejména na hostingách, kde je zakázáno nahrávat soubory na web prostřednictvím jazyka

PHP. Pokud to není nezbytně nutné, z důvodu bezpečnosti není doporučeno tuto službu využívat. Předposlední krok je zaměřen na základní nastavení stránek, jimiž jsou např. název stránek a metadata, ale i nastavení administrátorského účtu a hesla. V poslední části je možné nahrání ukázkových dat, které slouží pro prvotní seznámení s vnitřním systémem a principy redakčního systému Joomla!. Konečný sedmý krok je určen pro odstranění instalační složky systému Joomla! a uložení konfigurace. Odstranit instalační složku je podstatné, neboť pokud by byla zachována, mohl by kdokoli přepsat konfiguraci redakčního systému. Pokud systém Joomla! nemá práva pro mazání a vytváření souboru a složek, můžeme tento krok podle pokynů udělat manuálně například přes FTP. Jakmile je tento krok vykonán, instalace je dokončena. Po zadání webové adresy, na které byl dříve instalátor, se zobrazí webová prezentace zprostředkovaná systémem Joomla!, kterou budeme dále upravovat.



2.2.2 VYTVORENÍ ČLÁNKŮ WEBOVÉ PREZENTACE

V systému Joomla! jsou články základním prvkem prezentace, kterou si návštěvník prohlíží. Napsat články by tedy měla být jedna z prvních věcí, kterou uděláme na novém webu. Proto, abychom mohli napsat články, stejně jako pro většinu administrace, se musíme přihlásit přes administrativní rozhraní systému. K tomu se dostaneme přes internetový prohlížeč. Adresa administrace je www.našedoména.cz/administrator. Pro přihlášení do administrace zadáme údaje, které jsme vyplnili při instalaci v kroku šest, tedy účet a heslo administrátora. Po přihlášení do administrace se objeví rozhraní sloužící pro správu celého systému Joomla!. Toto rozhraní můžeme vidět na obrázku 3 – Administrace systému Joomla!.





Obrázek 3 – Administrace systému Joomla!

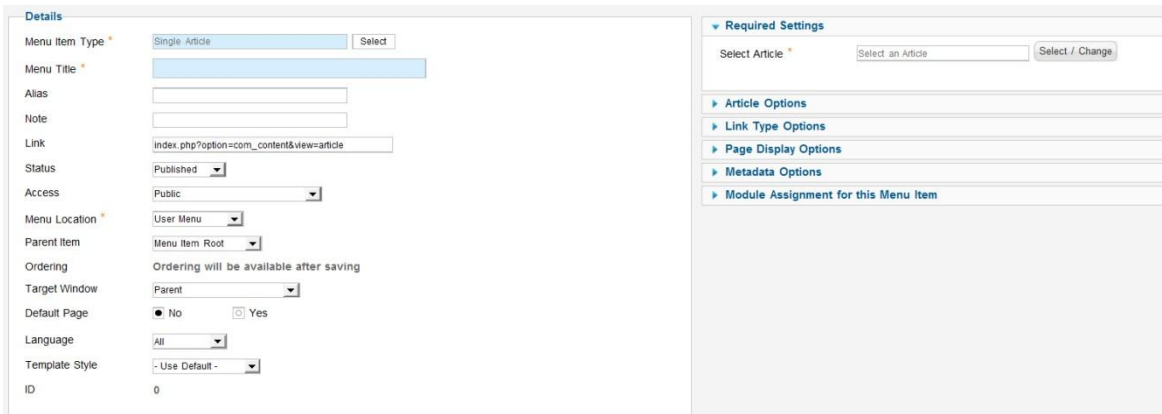
Nový článek můžeme napsat přes ikonu  Přidat nový článek (Add New Article) na úvodním panelu, anebo v horní nabídce pod kartou Obsah (Content) vybere možnost  Správce článků (Article Manager) a dále  Přidat nový článek (Add New Article). Ať už si vybereme jakoukoli cestu, otevře se formulář pro tvorbu článku s WYSIWYG editorem pro psaní těla článku. Formulář obsahuje několik prvků, které se mohou vyplnit, nicméně podmiňující jsou pouze Titulek (Title), Kategorie (Category) a Text článku (Article Text). Titulek je názvem článku a slouží jako identifikátor článku v rámci administrace, dále může sloužit jako nadpis stránky ve webové prezentaci. Kategorie udává příslušnost ke skupině souvisejících článků, s níž lze v redakčním systému dále pracovat. V našem případě bude stačit kořenová kategorie s názvem ARNICA. Pokud bychom v budoucnu potřebovali tvořit nové kategorie, můžeme vytvořit novou kořenovou kategorii, anebo kategorii, která je potomkem již existující kategorie. Tento systém nám umožňuje přehledně vytvořit komplexní hierarchii kategorií bez omezení rozsahu. Vytvoření kategorie se provádí v kartě Obsah (Content) pod nabídkou  Správce kategorií (Category Manager) a dále  Přidat novou kategorii (Add New Category). Pro tvorbu kategorií platí obdobná pravidla jako pro tvorbu článků. Poslední z podmiňujících polí nově tvořeného článku je Text článku (Article Text). Pro práci s textem slouží pohodlný WYSIWYG editor, který dokáže ovládat každý uživatel se základními znalostmi práce s PC. Editor umožňuje používání stylů textu, vkládání odkazů, obrázků a nabízí i další možnosti. Po kliknutí na tlačítko Uložit (Save) se článek uloží do databáze.

Pokud budeme chtít upravit již existující článek, vybereme kartu Obsah a klikneme na nabídku  Správce článků. Také můžeme použít úvodní stranu administrace a zvolit příslušnou ikonu  Správce článků. Následně se zobrazí výpis všech článků, které máme v systému Joomla! napsány. Pro rychlejší orientaci slouží pole pro vyhledávání článků. Když najdeme požadovaný článek, stačí kliknout na jeho titulek a otevře se nám stejný formulář jako při tvorbě článku. Formulář je již předvyplněný obsahem daného článku, který můžeme nyní upravovat. Změny opět uložíme tlačítkem Uložit (Save).


2.2.3 NASTAVENÍ NABÍDEK

Abychom ve webové prezentaci mohli přistupovat ke článkům, musíme vytvořit a správně nastavit nabídky. V administraci k těmto účelům slouží karta Nabídka (Menus). Nejprve musíme vytvořit novou nabídku. To provedeme ve  Správci nabídek (Menu



Manager) pod volbou  Přidat novou nabídku (Add New Menu). Otevře se nám okno, ve kterém musíme vyplnit pouze dvě položky: Titulek (Title) a Typ nabídky (Menu Type). Titulek slouží jako název nabídky a také pro další orientaci v administraci. Typ nabídky plní podobnou funkci jako alias u článku, slouží tedy především k systémovým účelům.





Obrázek 4 – Formulář pro správu položek menu


Každá nabídka se skládá z položek. Tyto položky jsou vlastně navigační odkazy, které se podle kritérií zobrazí na internetových stránkách. Položku nabídky vytvoříme tak, že v kartě Nabídky zvolíme příslušnou nabídku, kde chceme položku vytvořit a pak vybereme možnost  Přidat novou položku nabídky (Add New Menu Item). Tato volba vyvolá formulář pro vytvoření nabídky. Formulář je zobrazen na obrázku 4 – Formulář pro správu položek menu. Pro vytvoření položky nabídky jsou v první řadě podmiňující tři pole formuláře: Typ položky nabídky (Menu Item Type), Titulek nabídky (Menu Title) a Umístění položky (Menu location). Nejzásadnější z nich je Typ položky nabídky. Typ udává, co se zobrazí ve webové prezentaci, pokud se klikne na odkaz. Systém Joomla! v základu nabízí přes 30 typů, které můžeme využít, a s každou doinstalovanou komponentou se tento počet zvyšuje. Pro nás je podstatné využít typ pro zobrazení samotného článku. Takový typ se jmenuje Jeden článek (Single Article). Titulek nabídky je text zobrazený na internetových stránkách jako odkaz v nabídce. Také slouží pro orientaci v administraci nabídky. Pole Umístění nabídky dává na výběr z listu vytvořených menu. Vybereme menu, do kterého chceme vytvářenou položku umístit. V druhé řadě se mohou objevit další podmíněná pole formuláře, ta jsou závislá na volbě typu položky. V případě, že pracujeme s typem Jeden článek, objeví se nové pole Vyberte článek (Select Article), které nabízí seznam vytvořených článků. Vybereme článek, který chceme, aby se zobrazil, po kliknutí

na položku nabídky ve webové prezentaci. Ve formuláři je mnoho dalších nepovinných polí. Za zmínku stojí soubor polí označen jako Možnosti článku (Article Options) sloužící pro úpravu zobrazovaných parametrů daného článku, jako např. datum vytvoření, jméno autora, informaci, zdali titulek slouží jako odkaz, a podobně. Implicitní hodnoty těchto polí se získávají z globálního nastavení článků. Vytvořenou položku nabídky můžeme upravovat podobně jako tomu je u úpravy článku.

Nabídky v systému Joomla! jsou začleňovány do stránek za pomoci modulů. K vytvoření nového modulu přistoupíme v nabídce  Správa modulů (Module Manager) pod kartou Rozšíření (Extensions). Nad výpisem existujících modulů, který se zobrazí, klikneme na tlačítko  Nový (New). Zobrazí se typy modulů, které je možné použít. Z tohoto seznamu vybereme typ Nabídka (Menu). Ve formuláři, který se zobrazí, musíme vyplnit Titulek (Title), sloužící jako nadpis menu (potažmo modulu) ve webové prezentaci. Dalším podmiňujícím prvkem formuláře je Pozice (Position). Pozicí určujeme, kde v šabloně internetové stránky se modul zobrazí. Posledním klíčovým polem je Vyberte nabídku (Select Menu). V tomto poli vybereme příslušnou nabídku, kterou chceme zobrazit ve webové prezentaci.

2.2.4 JAZYKOVÉ MUTACE

Systém Joomla! je v základní instalaci pouze v anglickém jazyce. Abychom na stránkách mohli využívat i českého jazyka, musíme si stáhnout a nainstalovat jazykový balík češtiny. Toto rozšíření se dá stáhnout na českých stránkách věnovaných Joomla! www.joomlaportal.cz a nebo přímo na hlavních stránkách www.joomla.org. Instalace balíku českého jazyka pro systém se provádí skrze  Správce rozšíření (Extension Manager) nacházející se na kartě Rozšíření (Extensions). Balíček češtiny se jednoduše nainstaluje automaticky po nahrání instalačního souboru přes rozhraní formuláře, které Správce rozšíření nabízí. Nainstalovanou češtinu můžeme nastavit jak pro webovou prezentaci, tak i pro administraci pod volbou  Správce jazyků (Language Manager) v kartě Rozšíření (Extensions). Správce jazyků je zobrazen na obrázku 5 – Správce jazyků. Prostředí správce jazyků nabízí čtyři nastavení, se kterými můžeme pracovat. V nastavení Instalované - Stránky (Installed – Site) a Instalované – Administrace (Installed –

Administrator) můžeme nastavit výchozí jazyk pro naše stránky a administraci². Nastavení Obsah (Content) je podstatné pro rozdělení obsahu stránek do příslušných jazyků. Implicitně je v systému Joomla! obsahový jazyk angličtina „English (UK)“. Položku českého jazyka musíme tedy vytvořit. K formuláři sloužícímu pro vytvoření nového obsahového jazyka přistoupíme přes ikonu  Nový (New). Ve formuláři zadáme několik podmiňujících polí. Jedná se o pole Titulek (Title), které plní obdobnou funkci jako u článku. Nativní název (Title Native) je název jazyka v daném jazyce, například tedy „Čeština“. Dalším polem je URL kód jazyka (URL Language Code), jenž slouží pro generování odkazů na danou stránku v tomto jazyce, pokud tedy do pole zadáme „cz“, odkaz na stránky v češtině bude *www.našedoména.cz/cz/*. Prefix obrázku (Image Prefix) je prefixem obrázku vlajky daného jazyka, která se zobrazuje v přepínači jazyků na stránkách. V systému Joomla! je uložena vlajka s prefixem „cz“. Posledním podmiňujícím polem je Značka jazyka (Langue Tag). Ta se musí shodovat se značkou jazyka, který jsme nainstalovali, v tomto případě tedy „cs-CZ“. Posledním nastavením Správce jazyka je Přepsání (Overrides), ve kterém můžeme přepisovat a upravovat konstanty nainstalovaných jazyků.





The screenshot shows the Joomla! Language Manager interface. At the top, there is a header with a globe icon and the text "Správce jazyků: Instalované jazyky". To the right of the header are three icons: a star for "Výchozí" (Default), a document with "AB" for "Možnosti" (Options), and a circular arrow for "Nápověda" (Help). Below the header is a navigation bar with tabs: "Instalované - Stránky" (selected), "Instalované - Administrace", "Obsah", and "Přepsání". The main content area contains a table with the following data:

Číslo	Jazyk	Značka jazyka	Umístění	Výchozí	Verze	Datum	Autor	E-mail autora
1	<input type="checkbox"/> Czech (Czech Republic)	cs-CZ	Stránky		2.5.1	2012-02-03	Joomla! Czech Translation Team	admin@joomla.cz
2	<input type="checkbox"/> English (United Kingdom)	en-GB	Stránky		2.5.0	2008-03-15	Joomla! Project	admin@joomla.org

Obrázek 5 – Správce jazyků

Jazykový balík češtiny počestil samotný systém Joomla!, a nový obsahový jazyk vytvořil kategorii jazyka, pomocí které budeme obsah přiřazovat k požadovanému jazyku. Abychom ale mohli udělat vícejazyčný web, musíme dále zveřejnit modul a plugin sloužící pro zobrazování přepínače jazyků na stránkách a dále nastavit vícejazyčný obsah. Modul pro přepínání jazyka „Language Switcher“ povolíme ve Správci modulů, na kartě Rozšíření a v případě potřeby můžeme změnit jeho umístění na stránkách. Plugin „System – Jazykový filtr“, se stará o to, aby systém zobrazoval požadovaný obsah pro zvolený jazyk.

² Z důvodů přehlednosti a orientace je dobré nastavit administrátorskou část v českém jazyce, proto budou klíčové prvky (titulky, nabídky, volby a podob.) následně popisovány pouze v češtině.

Oba tyto prvky zveřejníme tím, že je označíme a klikneme na tlačítko  Zveřejnit. Nyní zbývá dotvořit obsah, který přiřadíme k daným jazykům. At už tvoříme článek, položku nabídky, nebo třeba modul, můžeme nyní v poli Jazyk vybrat jazyk, který jsme výše definovali, tím tedy určíme, že tento obsah je dán jen pro námi vybraný jazyk. Proto, aby přepínač jazyků správně fungoval, musíme ještě vytvořit úvodní stránky pro každý použitý jazyk. Úvodní stránka je stránka, která se zobrazí poté, co vybereme v přepínači jazyků požadovaný jazyk. Úvodní stránku nastavíme v kartě Nabídka, kde vybereme nabídku obsahující položku, jenž odkazuje na obsah, který je vhodný na to, aby byl úvodní stránkou některého z jazyků. Po vybrání nabídky označíme zvolenou položku a klikneme na ikonu  Úvod. Tím nastavíme úvodní stránku pro jazyk, který je nastaven pro tuto položku.

2.2.5 DALŠÍ POUŽITÉ PRVKY

Pro potřeby webu časopisu ARNICA využijeme i dalších částí a prvků, které základní instalace systému nenabízí. Jednou touto částí je galerie fotografií. V našem případě jsme použili rozšíření Phoca Gallery, které je volně dostupné na stránkách vývojářské skupiny <http://www.phoca.cz/phocagallery>. Instalace tohoto rozšíření, a i větší ostatních, probíhá stejně jako instalace jazyka, jak bylo uvedeno v kapitole 2.2.4.

2.2.4 Jazykové mutace. Pro správu galerie slouží robustní nástroj, ke kterému se dostaneme přes kartu Komponenty a následně volbu Phoca Gallery. Nástroj nabízí spousty možností a nastavení, pomocí kterých upravujeme funkce galerie a její zobrazení. Dalším prvkem je šablona webu, která zobrazuje (formuje, ilustruje?) obsah systému Joomla!. Šablona by měla vypadat originálně a měla by zaujmout návštěvníka stránek. V našem případě byla použita již předložená šablona.

3 NÁVRH VLASTNÍHO ŘEŠENÍ PRO SYSTÉM REDAKCE

Redakční systém Joomla! spolu s nabízeným rozšířením, tvoří stabilní základ webu, který slouží jako webová prezentace pro návštěvníky. Další částí internetových stránek vědeckého časopisu ARNICA je systém pro odevzdávání a hodnocení článků, který není veřejný pro běžné návštěvníky, ale využívají jej pouze členové redakce časopisu. Redakce časopisu ARNICA má specifický postup pro publikaci článků, tento postup je podrobně popsán v kapitole 1 spolu s rolemi, které jsou pro chod redakce nepostradatelné. Na základě tohoto postupu jsme schopni vytvořit vlastní rozšíření pro systém Joomla!, které bude sloužit jako nástroj pro členy redakce a usnadní tak proces publikace článků v elektronické podobě.

3.1 PRINCIPY TVORBY ROZŠÍŘENÍ

Tvorba vlastních rozšíření v systému Joomla! je velmi zásadní součástí redakčního systému. Díky této možnosti můžeme teoreticky neomezeně rozšiřovat vytvářené stránky podle našich představ a požadavků. Navíc máme podporu vestavěných Joomla! tříd a funkcí, které nám tuto práci mohou z části usnadnit. Tato silná možnost s sebou ovšem přináší i jistá pravidla a omezení, která je nutno respektovat a nově vznikající rozšíření podle nich tvořit. Stinnou stránkou tvorby vlastních rozšíření je časová náročnost tvorby oproti pouhému nainstalování a následného nakonfigurování již existujícího rozšíření. Na druhou stranu je, díky vestavěným pomocným prvkům, tvorba rozšíření snazší, než tvorba bez redakčního systému.

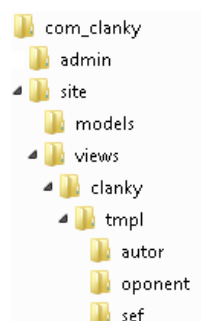
3.1.1 ROZDĚLENÍ ROZŠÍŘENÍ

V redakčním systému Joomla! se můžeme setkat se třemi hlavními druhy rozšíření. Jedná se o moduly, komponenty a plug-iny. I když každý z těchto typů je do základu specifický a používá se pro konkrétní problematiku, robustnější rozšíření může pokrýt i více těchto aspektů a tak se řadit do více tříd zároveň. Nejkomplikovanější z hlediska zásahu do jádra systému jsou, z výše uvedených tři typů, plug-iny. Obecně je funkcí plug-inu úprava vstupních nebo výstupních dat. U výstupních dat plug-in může pozastavit výstup z jádra systému Joomla!, následně ho upravit podle svého účelu a dále ho předat uživateli. U vstupních dat je tomu právě naopak. Příklad základního plug-inu systému Joomla! je WYSIWYG editor, který formuje HTML kód zapsaný v databázi systému na

grafický obsah a opačně. Dalším složitým rozšířením v kontextu manipulace s daty jsou komponenty. Systém Joomla! zobrazuje hlavní obsah pomocí jedné komponenty, která umožňuje obsah prohlížet a mnohdy i upravovat. Například zobrazování článků systému Joomla! je realizováno pomocí jedné komponenty, která tento obsah načítá z databáze systému a zobrazuje v pozici určené pro komponentu. Komponenty se dále vyznačují možností detailního nastavení a upravení funkčnosti daného rozšíření. Posledním uvedeným typem jsou moduly, které nejsou tolik robustní jako komponenty a nutně nezasahují do jádra systému jako plug-iny. Moduly slouží pro zobrazování menších prvků většinou dodatečného charakteru. Systém Joomla! umožňuje zobrazování několika modulů na jedné stránce. Mezi moduly patří například navigace, vyhledávání nebo upoutávky na dílčí části webu a podobně. V praxi spolu moduly a komponenty velmi často spolupracují. [1, s. 217 - 220]

3.1.2 STRUKTURA KOMPONENTY

Na základě analýzy požadavků budeme pro potřebu redakce vyvíjet rozšíření typu komponenta. Aby komponenta šla do systému Joomla! řádně implementovat a následně využívat, musí mít vytvořené systémové soubory v dané adresářové struktuře specifikované frameworkem systému Joomla!. Adresářová struktura komponenty se dělí na dvě hlavní části. První je složka *admin*, která je určena pro administraci komponenty a druhou složkou je *site*, jenž slouží pro zobrazování obsahu webové prezentace. Podrobnější rozčlenění adresářů je zřejmé z obrázku 6 – Adresářová struktura komponenty.



Obrázek 6 – Adresářová struktura komponenty

Částí spojenou s administrací se dále zabývat nebudeme, namísto toho se zaměříme na zobrazení pro uživatele komponenty. Adresář *site* musí obsahovat několik systémových souborů. První z těchto souborů je tzv. *controller*. Název tohoto souboru je

„controller.php“ a jeho úkolem je propojit komponentu s jádrem systému. Tato složka musí obsahovat také PHP soubor, který volá controller. Tento soubor se jmenuje podle názvu komponenty, v našem případě tedy „clanky.php“. Další podmíněnou částí je tzv. *view*, jejíž úkolem je zobrazovat obsah komponenty. Na adrese „site/views/clanky/view.html.php“ je PHP třída obsahující funkce, které jsou určeny pro zobrazování obsahu. Hlavní tělo programu je pak uloženo v souboru „default.php“ na adrese „site/views/clanky/tmpl/“. V tomto souboru můžeme volat funkce a proměnné z třídy „view.html.php“, importovat do něj další soubory a skripty týkající se už přímo funkce komponenty a celkově v něm tvořit komponentu tak, jak ji uvidí koncový uživatel. Posledním podmiňujícím souborem je XML soubor „clanky.xml“ umístěný v kořenovém adresáři komponenty. V tomto souboru jsou uloženy základní informace o komponentě, jako je název komponenty, jméno autora, datum vytvoření a hlavně také adresářová struktura komponenty. Toto jsou základní nutné náležitosti pro správný chod komponenty. V případě, že budeme chtít využít některých dalších možností systému Joomla!, je nutno přidat další systémové soubory, které budou tyto funkce obstarávat. [3]

3.1.3 ZABUDOVANÉ FUNKCE A TŘÍDY JOOMLA!

Redakční systém Joomla! disponuje zabudovanými třídami a funkcemi, které mohou usnadňovat vytváření komponenty. Mezi třídy, které byly využity v komponentě pro vědecký časopis ARNICA, patří třída *JFactory*. Třída *JFactory*³ dokáže pracovat mimo jiné s uživatelskými údaji uloženými v databázi a dále jsme jejím prostřednictvím schopni spojit se s databází a provádět vlastní dotazy. Další třídou je *JRequest*⁴. Tato třída slouží pro práci s globálními proměnnými a uplatníme ji především pro předávání požadavků uvnitř komponenty. Využitím metod třídy *JHtml*⁵ dokážeme pracovat s různými HTML prvky, kalendářem a získávat formátované datum. Poslední třídou využívanou v komponentě je *JFile*⁶, která obstarává práci se soubory, jako je nahrávání na server, kopírování, přesouvání a podobně. V komponentě je řešeno taktéž vytváření několika

³ Více informací o třídě *JFactory* na <http://docs.joomla.org/JFactory/1.6>

⁴ Více informací o třídě *JRequest* na <http://docs.joomla.org/JRequest/1.6>

⁵ Více informací o třídě *JHtml* na <http://docs.joomla.org/JHTML/1.5>

⁶ Více informací o třídě *JFile* na <http://docs.joomla.org/JFile/1.6>

složek pro tyto účely slouží třída *JFolder*⁷. Pro potřeby komponenty nám ale v tomto případě vystačí třída *JFile*.

Na prvním řádku ve skriptech komponenty se setkáme s otestováním existence konstanty „JEXEC“. Pro toto ověření slouží následující kód: *defined('_JEXEC') or die('Restricted access');* [4]. Ve skutečnosti se nejedná přímo o funkci systému Joomla!, nicméně o veřejnou PHP funkci, kterou testujeme existenci, respektive definování konstanty deklarované systémem Joomla!. Pokud funkce zjistí, že konstanta *JEXEC* nebyla definována, zastaví se vykonávání příslušného skriptu. V praxi tento kód slouží jako ochrana proti spuštění vykonávacích skriptů mimo systém Joomla!.

3.1.4 PŘEDÁVÁNÍ POŽADAVKŮ UVNITŘ KOMPONENTY

Předáváním požadavků uvnitř komponenty rozumíme přechod z jedné stránky, její části, či ze skriptu komponenty na jinou stránku, jinou část stránky či jiný skript uvnitř komponenty spolu s předáním určitých parametrů, na kterých jsou závislé následující prováděné akce. Tento aspekt je klíčovým pro správné provádění skriptů a pro celkový chod komponenty. Prvním použitým mechanismem je předávání požadavků za pomoci volby položky nabídky obsluhující komponentu. Proto, abychom tohoto dosáhli, musíme komponentě přidat následující dva systémové sobory. Soubor „default.xml“ umístíme na adresu „site/views/clanky/tmpl/“. Jeho funkcí je zobrazení typu položky nabídky (viz. kapitola 2.2.3) při vybírání typů položky nabídky v administrátorské části systému Joomla!. Dalším úkolem toho souboru je následné zobrazení volby, ve které můžeme vybrat podtyp nabídky komponenty např. Oponent – Moje posudky. Další soubor je tzv. *model* nesoucí stejný název jako komponenta a musí být umístěn na adrese „site/models/clanky.php“. V tomto souboru je metoda typu *get*, která získává hodnotu proměnné na základě volby podtypu nabídky v administraci. Aby metoda *get* v komponentě správně fungovala, musíme přidat do souboru typu *view* funkci, která propojí tuto metodu s tělem komponenty. [5] [6]

Následující mechanismus využívá formulářů a třídy *JRequest* přímo v hlavním kódu komponenty. Když odešleme formulář s vyplněnými položkami, jejich obsah se uloží do proměnných v paměti systému Joomla!. K těmto proměnným můžeme následně

⁷ Více informací o třídě *JFolder* na <http://docs.joomla.org/JFolder/1.6>

přistupovat za pomoci funkce `getVar('proměnná')`, ze třídy `JRequest`. Principem posledního využitého mechanismu je ukládání proměnných do pole `REQUEST`, které PHP server v základu nabízí. Jednou z možností jak proměnou do pole přidáme je její přiřazení do kódu odkazu, sloužícímu k přesměrování. Držíme se přitom následující syntaxe ``. Z pole se hodnoty následně načítají prostřednictvím klasického PHP příkazu: `$pom = $_REQUEST["identifikátorProměnné"]`.

3.2 VLASTNÍ TVORBA KOMPONENTY

Úskalí tvorby komponenty již bylo nastíněno. Dále budou rozepsány dílčí části komponenty, které jsou nepostradatelné pro celkovou funkčnost komponenty sloužící pro proces publikace článků časopisu ARNICA. Kapitola je zaměřena na hlavní principy a myšlenky, které jsou implementované do zdrojových kódů, a podle kterých komponenta funguje. Pro dobré pochopení problematiky jsou uváděny části kódů. Kompletní zdrojové kódy komponenty jsou na přiloženém CD. Organizace jednotlivých souborů zdrojového kódu je patná z přílohy B. Kompletní adresářová struktura komponenty `com_clank`.

3.2.1 DATABÁZOVÁ STRUKTURA KOMPONENTY

Komponenta využívá databáze pro ukládání informací o člancích, které jsou v procesu publikace. Hlavní tabulkou databáze je tabulka s názvem „`clanky_clanky`“. Do této tabulky se zapisují všechny články, které autoři odevzdali. Mimo identifikátoru článku (`id`) obsahuje tabulka sloupce: název článku (`nazev`), identifikátor autora (`autor_id`) a stav článku (`stav`). Účel prvních tří sloupců je zřejmý z jejich názvu, komplikovanější je ale sloupec `stav`. Tento sloupec je určen pro data typu *integer*, tedy celá čísla a přímo s ním pracují jen skripty komponenty. Stav článku mohou nabývat hodnot 1 až 8. Konkrétní soupis stavů článku je uveden v tabulce 1 – Přehled stavů článku. Stav článku hraje podstatnou roli v chodu komponenty.

Další databázová tabulka uchovává záznamy s informacemi o verzích článku. Její název je „`clanky_verze`“, a obsahuje sloupce identifikátor verze (`id`), identifikátor článku (`id_clanek`), název souboru verze článku (`nazev_souboru`) a datum (`datum`). Do sloupce název souboru se ukládají řetězce, které jsou složeny z unikátního prefixu a názvu souboru spolu s koncovkou formátu. Tento prefix zaručuje jedinečnost názvu souboru,

takže nemůže dojít k náhodnému přepsání při nahrávání souboru na server. Je zřejmé, že tabulka „clanky_clanky“ a „clanky_verze“ je propojena pomocí příslušných identifikátorů. Pro jeden záznam v tabulce „clanky_clanky“ může existovat více záznamů v tabulce „clanky_verze“ – jedná se tedy o relaci 1:N.

Stav článku	Popis stavu
1	Článek čeká na přiřazení alespoň jednoho oponenta.
2	Článek čeká na odevzdání posudků od všech přiřazených oponentů.
3	Článek čeká na rozhodnutí šéfredaktora, zdali článek schválí (5), nebo ho zašle autorovi k přepracování, (4) anebo zdali článek zcela zamítne (6).
4	Článek byl zaslán autorovi zpět a čeká na odevzdání nové verze.
5	Článek byl schválen.
6	Článek byl zcela zamítnut (konečný stav).
7	Článek byl zaslán redakční radě pro vyjádření k publikaci.
8	Článek byl publikován (konečný stav).

Tabulka 1 – Přehled stavů článku

Tabulka „clanky_posudek“ v sobě uchovává informace o posudcích, které nahráli oponenti k určitým verzím článkům jako své hodnocení. Tabulka využívá následujících sloupců: identifikátor posudku (id), identifikátor verze článku (id_clanku), identifikátor oponenta (oponent), stručné hodnocení (hodnoceni) a název souboru daného posudku (nazev_souboru). Název souboru je podobně jako v případě tabulky „clanky_verze“ obohacen o unikátní prefix. Sloupec hodnoceni slouží pro uchování systémové informace o stručném hodnocení verze článku. Popis hodnot, které může sloupec nabývat, je zaznamenán v tabulce 2 – Hodnoty sloupce *hodnoceni* z tabulky *clanky_posudek*. Spojujícím prvkem mezi oponenty, potažmo posudky, a články je tabulka „clanky_opONENTI“. Tabulka obsahuje pouze sloupce identifikátor článku (id_clanek) a identifikátor oponenta (id_oponent), pomocí kterých tvoří relace článků a oponentů.

Poslední tabulkou databáze je tabulka s názvem „clanky_redakce“ a je sestavena ze sloupců identifikátor hodnocení (id), identifikátor oponenta (id_oponent), identifikátor článku (id_clanek) a hodnocení redakční rady (hodnoceni). Úkolem tabulky je uchovávat data o hodnocení redakční rady pro články, které byly zaslány redakční radě k vyjádření o publikaci. Přehled těchto hodnot, které může nabývat sloupec *hodnoceni* je uveden v tabulce 3 – Hodnoty sloupce *hodnoceni* z tabulky *clanky_redakce*. Znárodnění tabulek

databáze komponenty spolu s vyznačenými relacemi ilustruje příloha A. Relace tabulek komponenty `com_clanky`.

Hodnota pole	Popis
1	Doporučuji
2	Doporučuji s výhradami
3	Nedoporučuji

Tabulka 2 – Hodnoty sloupce *hodnoceni* z tabulky `clanky_posudek`

Hodnota pole	Popis
1	Zdržuji se hlasování
2	Ano (Publikovat)
3	Ne (Nepublikovat)

Tabulka 3 – Hodnoty sloupce *hodnoceni* z tabulky `clanky_redakce`

Pro komunikaci mezi komponentou a tabulkami databáze slouží dotazovací jazyk SQL. Systém Joomla! nabízí třídu *JFactory*, která je určena pro komunikaci s databází. Příkazem `$db = JFactory::getDBO();` získáme objekt, se kterým můžeme dále pracovat jako s databází. Například tedy příkazem `$db->setQuery($query);` provedeme SQL dotaz, který je uložen jako řetězec v proměnné `$query`. Uložení výsledku dotazu do databáze provedeme funkcí `loadRowList()`. Tato funkce zároveň může fungovat i jako načtení hodnot, které vrací provedení dotazu. Pomocí třídy *JFactory* ale můžeme získávat i jiné objekty než celou databázi, například funkce `getUser()` navrací objekt uživatele.

3.2.2 UKLÁDÁNÍ ČLÁNKU

Jak je patrné z obrázku 1 – Životní cyklus článku, prvním krokem v životním cyklu článku v rámci redakce je jeho odevzdání. Ze systémového pohledu se jedná o uložení souboru na server. V komponentě slouží k odevzdání nového článku soubor `odevzdat.php`, který je na adrese „`site/views/clanky/tmpl/autor/`“. Tento soubor je rozdělen do dvou větších funkčních bloků. První z nich je výchozí a zobrazuje pomocí HTML tagů `<form>` a `<input>` formulář, který po správném vyplnění slouží pro odeslání souboru. Formulář neodesílá data do žádného externího skriptu, ale volá sám sebe.

Ve druhém bloku zpracovávající odeslaný formulář se stará PHP skript o odevzdání samotného souboru. Tento blok je spuštěn pouze v případě, že byl odeslán formulář z bloku prvního. Hodnoty odeslaných polí získáme pomocí funkce `getVar('proměnná')`, třídy *JRequest*. Pokud ale chceme pracovat s informacemi o zaslaném souboru, nedostaneme jen jednu hodnotu, ale pole hodnot. K jeho získání slouží příkaz `$file = JRequest::getVar('up_file', 'chyba', 'files', 'array');`. Parametry funkce `getVar()` jsou v tomto případě při popisu zleva: název pole formuláře, implicitní návratová hodnota, původ

proměnné a typ vrácené proměnné⁸. Funkce navrací do proměnné *\$file* pole různých informací, jako je velikost souboru, jeho dočasné umístění, nebo např. jeho název. Pokud byl odevzdán neprázdný soubor, PHP skript může uložit článek na server.

K odevzdávanému souboru článku se ale vážou i další informace, jako je název článku a datum kdy byl odevzdán. Tyto informace se ukládají spolu s názvem souboru do databáze komponenty. Spojení s databází zrealizujeme prostřednictvím příkazu *\$db =& JFactory::getDBO()*; První operací s databází je vložení záznamu o článku do tabulky „clanky_clanky“. Mimo názvu článku ukládáme i identifikátor autora článku a výchozí stav článku s hodnotou rovné 1. Identifikátor je dostupný z objektu uživatele, který získáme příkazem *\$user =& JFactory::getUser()*; Identifikátor autora je pak uložen v proměnné *\$user->id*.

Další operací s databází je vložení informací o první, tedy odevzdávané, verzi článku do tabulky „clanky_verze“. V tomto případě se vkládaný záznam skládá ze tří hodnot. První je identifikátor odevzdávaného článku, tedy identifikátor předchozího provedeného záznamu. Dalším je název souboru a posledním je datum vložení. Protože může nastat situace, kdy se na server nahrají dva soubory s totožným názvem, ale rozdílným obsahem, musí se tyto dva soubory rozlišit, aby nedošlo k jejich přepsání a následné ztrátě dat. Toto rozlišení provedeme přidáním unikátního prefixu. Tento prefix se skládá z identifikátoru článku a čísla verze článku. Celkové složení názvu souboru je patrné na příkazu: *\$navez_souboru = \$id_clanku."_1_".\$file['name']*; Namísto proměnné čísla verze článku je zde napevno zadána hodnota 1, to proto, že tento skript slouží k odevzdávání pouze nových článků, tedy prvních verzí. Tyto operace s databází provedeme příkazem *\$zapis = \$db->setQuery(\$query)*; V případě první operace s databází se v proměnné *\$query* bude nacházet tento SQL dotaz: *INSERT INTO `clanky_clanky` (`navez`,`autor_id`,`stav`)VALUES (\'".\$navez.\', \\'\$user->id.\',\'1\')*.

Samotné uložení souboru na disk se provádí pomocí funkce: *JFile::upload(\$src,"images/com_clanky/clanky/".\$navez_souboru)*; a jde prakticky o poslední část procesu uložení článku. Prvním parametrem funkce je zdroj nahrávaného

⁸ Více informací o funkci *getVar()* třídy *JRequest* na <http://joomla.org/gf/project/joomla/scmsvn/?action=browse&path=/development/tags/1.6.x/1.6.3/libraries/joomla/environment/request.php&view=markup>

souboru a tedy dočasné místo, kam webový server nahrál soubor, který byl zadán do formuláře⁹. V našem případě tuto adresu máme uloženou v proměnné `$file['tmp_name']`. Druhým parametrem funkce je cílová adresa, tedy místo na serveru, kam se soubor uloží. Úložný prostor byl zvolen do adresáře `images` z důvodu, jelikož v případě systému Joomla! je právě tento adresář velmi často nastaven s vhodnými právy pro zápis prováděný systémem.

V komponentě se setkáváme i s odevzdáváním novějších verzí článků. Tento úkol řeší soubor `odevzdat_novou.php`, který se nachází na stejném místě jako předchozí soubor. V principu se jedná o stejný průběh, rozebraný výše. Hlavní rozdíl je ve formě názvu ukládaného souboru, který se liší tvorbou prefixu. Identifikátor článku zůstává stejný jako u předchozí verze článku, ale číslo verze se zde liší. Řetězec s názvem souboru vypadá tedy takto: `$navez_souboru = $idc."_"$nova_verze."_"$file['name']`. Proměnná `$idc` je identifikátor článku, pro který nahráváme novou verzi. V řetězci `$nova_verze` je uloženo číslo nahrávané verze a význam `$file['name']` zůstává stejný jako v případě odevzdání první verze.

3.2.3 VYPISOVÁNÍ ČLÁNKŮ

Články, které byly nahrány na server a k nim přiřazená data uložena do databáze, musíme nejprve prezentovat, abychom s nimi mohli dále pracovat. S výpisem článků se setkáváme ve většině částí komponenty. Princip výpisu je až na drobnosti všude stejný. V každém případě se jedná o výpis záznamů z databáze, které jsou formovány HTML tabulkou a jsou zasazeny do cyklu.

V případě skriptu uloženého v souboru `publikovane_clanky.php`, který je na adrese „`site/views/clanky/tmpl/sef/`“, bude výpis řešen následovně. Skript začíná dotazem, který z databáze vybere všechny články, jež byly publikovány a seřadí je sestupně podle identifikátoru nejnovější verze článku. Tím docílíme vypsaní článků od nejnovějšího po nejstarší i s ohledem na novější verze článků. Složený databázový dotaz bude vypadat takto: `SELECT C. * FROM (SELECT A. *, B.datum, B.navez_souboru, B.id AS id_verze FROM `clanky_clanky` AS A, clanky_verze AS B WHERE A.`stav` =8 AND A.id = B.id_clanek ORDER`

⁹ Více informací o funkci `upload()` třídy `JFile` na <http://joomla.org/gf/project/joomla/scmsvn/?action=browse&path=/development/tags/1.6.x/1.6.3/libraries/joomla/filesystem/file.php&view=markup>

BY id_verze DESC) AS C GROUP BY C.id ORDER BY id_verze DESC. V dotazu je patrné, že vybíráme pouze články se stavem, který je roven hodnotě 8. Tato hodnota je určená pro publikované články. Pokud tedy chceme vypsat v jiném souboru např. články schválené, změnímme zadanou hodnotu podle tabulky 1 – Přehled stavů článku.

S načtenými výsledky dotazu dále pracujeme v cyklu, který se bude opakovat tolikrát, kolik jsme dostali záznamů z databáze. V cyklu se vypisují jednotlivé údaje o článcích, které jsou formovány do tabulky pomocí HTML tagů. Z předchozího dotazu ale nezjistíme všechny údaje potřebné pro kompletní výpis článku. Záznam jednotlivého článku s sebou nese identifikátor jeho autora. Pro uživatele je ale příjemnější orientovat se podle autora jména nežli podle identifikátoru. Proto na základě tohoto identifikátoru vybereme z tabulky systému Joomla! „users“ příslušného autora, a z tohoto výsledku získáme jeho jméno. Dalším dodatečným údajem je seznam oponentů, kteří jsou aktuálně přiděleni pro daný článek. Tento seznam získáme SQL dotazem, ve kterém se zeptáme na všechny záznamy v tabulce „clanky_opONENTI“, ve kterých je identifikátor článku roven identifikátoru článku, na něhož se dotazujeme. Pomocí vnořeného cyklu pak jména oponentů přidáme do řetězce. Tento řetězec můžeme nakonec vypsat v tabulce spolu s odkazem na soubor samotného článku a se všemi ostatními zbývajícími údaji o článku.

Výpisy článků v jiných částech komponenty jsou až na drobné změny podobné tomuto příkladu. Jeden z rozdílů se nachází například ve skriptu *moje_clanky.php* umístěném v adresářové struktuře na místě „site/views/clanky/tmpl/autor/“. Jedná se o dodatečný výpis stavu článku, který informuje autora článku o tom, v jaké fázi se jeho článek právě nachází. Články se pak samozřejmě nevypisují na základě hodnoty stavu, ale podle hodnoty identifikátoru příslušného autora tak, aby měl autor vypsané pouze ty články, které napsal.

3.2.4 PŘÍRAZENÍ OPONENTA PRO ČLÁNEK

Aby oponenti mohli posuzovat články, musí je nejprve šéfredaktor k příslušným článkům přiřadit. O proces přiřazování oponentů ke článkům se stará soubor *add_oponent.php*, který je umístěn na adrese „site/views/clanky/tmpl/sef/“. Kód tohoto souboru pracuje s konkrétním článkem, který byl vybrán v některém z výpisu článků.

Soubor můžeme rozdělit na dva funkční bloky, a to blok pro přidávání oponentů a blok pro jejich odebrání.

Blok zajišťující přidávání oponentů ke článku začíná výpisem oponentů nepřijezzených pro tento článek. Jak získáme seznam těchto oponentů je popsáno v dalším odstavci. Oponenti jsou vypisováni do formuláře jako jednotlivé *<option>* pro tag *<select>*. Díky tomuto mechanismu můžeme vybrat jednotlivého oponenta a odesláním formuláře předáme jeho identifikátor dále ke zpracování. Formulář volá, stejně jako v případě ukládání článků, sám sebe. Po odeslání formuláře se v tomto bloku splní podmínka a začne se provádět další část kódu. V první řadě se vloží záznam do tabulky „clanky_oponenti“, který je složen z identifikátoru oponenta, jež jsme získali z formuláře a z identifikátoru článku, pro který oponenta přidáváme. Po přidání oponenta k článku následuje podmínka, která se vykoná v případě, že stav článku je roven hodnotě 1, tedy že článek čeká na přiřazení oponenta. V těle podmínky se pak provede SQL příkaz: *UPDATE `clanky_clanky` SET `stav` = '2' WHERE `id` = '\$idc.'*, který změní stav vybraného článku na hodnotu 2. Tato hodnota je určena pro stav, kdy článek čeká na odevzdání posudků všech přiřazených oponentů. Tento blok pak končí přesměrováním stránky na sebe na sama, což slouží jako ochrana proti přidání redundantního záznamu, aktualizováním webové stránky.

Další funkční blok je zaměřen na odebrání oponentů. Z databázové tabulky „clanky_oponenti“ vybereme všechny záznamy, které mají identifikátor vybraného článku. Takto získáme identifikátory oponentů, přiřazených k tomuto článku. Na základě porovnávání se seznamem všech oponentů jsme schopni získat množinu oponentů, kteří nebyli přiřazeni. S touto množinou pak pracujeme v úvodu funkčního bloku, popsaného jako první. Dále pak v cyklu procházející záznamy z předchozího dotazu zjistíme na základě identifikátoru jednotlivého oponenta jeho jméno. Jméno a příjmení oponenta vypíšeme jako položku seznamu ** a zároveň s ním, jako další součást položky seznamu, vypíšeme i odkaz: *odebrat*. Tento odkaz přesměruje stránku na skript *del_oponent.php*, který provede samotné smazání relace oponenta a článku. V odkazu je patrné, že proměnná *ido* se bude rovnat identifikátoru zvoleného oponenta a proměnná *idc* se bude rovnat identifikátoru článku.

Ve skriptu *del_oponent.php* dochází v první řadě k mazání záznamů z tabulky „clanky_oponenti“ na základě identifikátorů oponenta a článku. Další částí, po smazání relace, je ověření, zdali není počet odevzdaných posudků roven počtu přiřazených oponentů. V případě že tomu tak je, změní se stav článku, se kterým pracujeme, na hodnotu 3. Tato hodnota signalizuje, že byly odevzdány posudky od všech přiřazených oponentů. Tato část kódu slouží jako opatření proti zaseknutí cyklu článku v případě, že téměř všichni oponenti odevzdali posudky a následně je neaktivní část oponentů odebrána z článku. Skript je zakončen přesměrováním zpátky na stránku *add_oponent.php*.

3.2.5 NAHRÁNÍ POSUDKŮ PRO ČLÁNEK

Pokud šéfredaktor přiřadil ke článku oponenta, může nyní daný oponent nahrát posudek pro tento článek. Přiřazené články, ke kterým má oponent napsat posudky, vypisuje soubor *napsat_posudek.php*. Tento soubor je uložen na adrese „site/views/clanky/tmpl/oponent/“. Po vybrání článku je oponent přesměrován na soubor *add_posudek.php* nacházející se ve stejné složce jako předchozí soubor. Soubor *add_posudek.php* se pak stará o nahrání posudku pro vybraný článek a můžeme jej opět rozdělit na dva funkční bloky, kde jeden slouží pro zobrazení formuláře a druhý pro práci s daty, které jsme získali z formuláře.

Formulářový blok zobrazí formulář v případě, že oponent ještě nezaslal posudek. Pokud tato podmínka není splněna, vypíše blok varovnou hlášku. Tato situace by díky systému stavů článku neměla nastat. Podmínka zde slouží hlavně jako ochrana proti duplicitnímu odevzdání jednoho posudku při aktualizování webového prohlížeče. Samotný formulář je složen především z pole, do kterého zadáváme soubor, a z dodatečného pole typu *<select>*, ve kterém vybíráme stručné textové hodnocení, jež je uvedeno v tabulce 2 – Hodnoty sloupce *hodnoceni* z tabulky *clanky_posudek*. Následující funkční blok skriptu zpracovává již odeslaná data.

První část druhého bloku toho souboru se velmi podobá druhému bloku u ukládání článků, který byl popsán v kapitole 3.2.2. Namísto vkládání dvou záznamů do databáze nám stačí vložit pouze jeden záznam a to prostřednictvím tohoto dotazu: *INSERT INTO `clanky_posudek` (`id_verze` , `oponent` , `hodnoceni` , `nazev_souboru`)*

`VALUES (\$.šveze["0"]["0"].\', \$.šuser->id.\', \$.šhodnoceni.\', \$.šnavez_souboru.\')`. Tento dotaz tedy vloží záznam do tabulky „clanky_posudek“, kde do sloupce *id_verze* vkládáme identifikátor nejnovější = aktuální verze článku. Do sloupce *oponent* vkládáme identifikátor oponenta. Stručné hodnocení vybrané ve formuláři se v podobě systémové proměnné ukládá do sloupce *hodnoceni*. Poslední sloupec *navez_souboru* je určen pro název souboru posudku, který jsme také zadali do formuláře. Tento název je obohacen podobně jako u problematiky ukládání článků o unikátní prefix. V samotné funkci sloužící pro uložení souboru na server je rozdíl oproti skriptu ukládání článku pouze v cílové adrese, která je v tomto případě takováto: „images/com_clanky/posudky/\".šnavez_souboru“.

Následující část tohoto funkčního bloku zjišťuje, jestli byly nahrány všechny posudky pro daný článek. Jedná se o testování, zdali se počet posudků rovná počtu oponentů. Tento princip byl již popsán u souboru *del_oponent.php*, který byl uveden v kapitole 3.2.4 – Přiřazení oponenta pro článek, a pro chod komponenty je nepostradatelný. Po provedení funkčních bloků se na stránce můžeme pomocí odkazu zpět vrátit na výpis článků určených pro posouzení.

3.2.6 SCHVALOVÁNÍ ČLÁNKŮ

Schvalovat se může takový článek, který obdržel posudky od všech svých oponentů. Výpis těchto článků umístěných v sekci „Posouzené články“ je v souboru *ne_clanky.php*, který nalezneme na adrese „site/views/clanky/tmpl/sef/“. Ve výpisu článků jsou zde navíc tři odkazy sloužící pro přesměrování na stránky, které změní stav na základě jejich funkce. Soubory se nacházejí ve stejné složce jako soubor *ne_clanky.php*. Tyto tři odkazy jsou: schválit článek, vrátit k přepracování, zcela zamítnou článek. Každý z odkazů s sebou nese identifikátor článku, na jehož základě provádí své následující akce.

Odkaz „schválit článek“ vede k souboru *schvalit_clanek.php*. V tomto souboru se o schválení článku stará z počátku formulář, který se pouze dotazuje na schválení operace. Tento mechanismus zde slouží jako obrana proti nechtěnému spuštění skriptu, neboť změna stavu článku, jako je schválení, je konečná a nejde vrátit zpět. Po odeslání formuláře, který volá sám sebe, se provede další část kódu. V té je pouze provedení SQL dotazu měnící stav článku na hodnotu 5 a dále vypsání hlášky o provedení operace.

Soubory, ke kterým se dostaneme skrze odkazy „vrátit k přepracování“ a „zcela zamítnout článek“, fungují na stejném principu jako soubor *schvalit_clanek.php*. Jejich rozdíly jsou především v hodnotě, na jakou se mění stavy článků a v textových výpisech.

3.2.7 PUBLIKOVÁNÍ ČLÁNKU

Poté, co je článek schválen, je dále zaslán redakční radě pro vyjádření o publikaci. O tento krok se stará skript *zaslat_rade.php* umístěn na adrese „site/views/clanky/tmpl/sef/“. V tomto skriptu se v podstatě vykoná pouze změna hodnoty stavu příslušného článku na hodnotu 7. Skript můžeme, pro konkrétní článek spustit z výpisu článků provedeným souborem *schvalene_clanky.php*, který se nachází ve stejné složce.

Po změně stavu na hodnotu 7 má redakční rada možnost vyjádřit svůj souhlas či nesouhlas s publikací článku. Redakční rada se skládá ze všech oponentů a k hlasování má přístup ze stránky *napsat_posudek.php*, kde se nachází výpis jak přiřazených článků k oponentuře, tak i výpis článků, které byly zaslány celé redakční radě. Z této stránky se pomocí odkazu nesoucí s sebou identifikátor zvoleného článku, přesměrujeme na soubor *redakce_schvalovani.php*, jenž je stejně jako předchozí soubor uložen na místě „site/views/clanky/tmpl/oponent/“. První částí v tomto souboru je formulář, ve kterém se vyplňuje stručné hlasování. Možnosti hlasování jsou zřejmé z tabulky 3 – Hodnoty sloupce *hodnoceni* z tabulky *clanky_redakce*. Po zvolení požadované hodnoty v poli `<select>`, můžeme odeslat formulář. Formulář volá sám sebe a po jeho odeslání se začne plnit další část kódu. V této části se ze získaných hodnot z formuláře provede SQL dotaz, který vytvoří nový záznam hodnocení daného oponenta do tabulky „clanky_redakce“.

Toto hlasování není nevratné jako například odesílání posudků pro článek a dá se podle uvážení měnit. Při vypisování formuláře se v kódu prověřuje, zdali daný uživatel již hlasoval. Pokud tomu tak je, tak se do příslušného tagu `<option>`, který slouží pro výpis a výběr hodnocení, připiše `selected="selected"`. Tímto docílíme toho, že uživatel bude mít předem vybranou položku, kterou zvolil v předchozím hlasování. Pokud si uživatel bude přát změnit hodnocení, změní tak vybranou položku hlasování na jinou a formulář odešle. Po odeslání formuláře se nyní nebude přidávat do tabulky nový záznam, ale změní se ten záznam, ze kterého jsme čerpali při vypisování formuláře. Tuto operaci provedeme SQL

dotazem `UPDATE `clanky_redakce` SET `hodnoceni` = '$hodnoceni_redakce' WHERE `id` = '$id_hodnoceni'`, kde proměnná `$hodnoceni_redakce` je hodnocení získané z formuláře a proměnná `$id_hodnoceni` je identifikátor předchozího hodnocení.

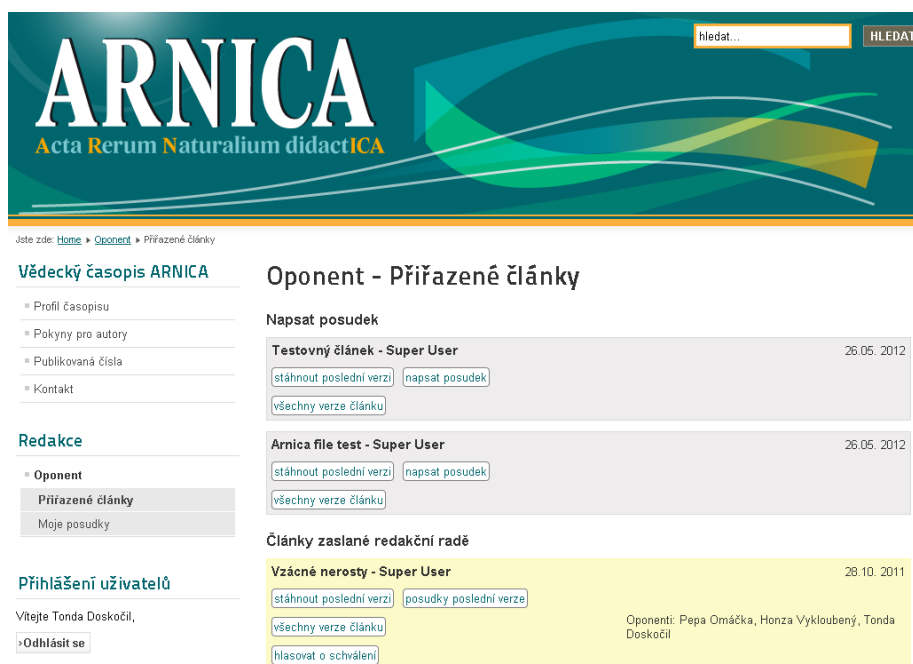
Na základě hodnocení redakce se šéfredaktor rozhoduje, zdali článek publikuje, nebo zcela zamítne. K tomuto konečnému rozhodnutí se šéfredaktor dostane přes soubor `hodnoceni_rady.php`, ve kterém je přehled hlasování redakční rady. Přehled je rozdělen do dvou částí. V první části se vypisují všichni členové redakční rady spolu s jejich hodnocením, které je uloženo v tabulce „clanky_redakce“. V druhé části výpisu je zobrazen číselný počet jednotlivých hodnocení. Když šéfredaktor uzná za vhodné, že už se vyjádřil dostatečný počet oponentů, může článek publikovat, anebo zcela zamítnout. K přesměrování na provedení těchto operací slouží opět odkazy. Průběh akce „zamítnutí článku“ byl již popsán v kapitole 3.2.6. Princip operace „publikovat článek“ je velmi podobný předchozí akci zamítání článku. V kódu je v tomto případě ubrán formulář pro potvrzení akce, neboť šance náhodného kliknutí na odkaz sloužící k přesměrování na skript je řádově menší. Další změnou je samozřejmě hodnota, na kterou se stav mění.

4 VÝSLEDNÝ PRODUKT A JEHO POPIS

V předchozí kapitole byla komponenta představena především z hlediska její tvorby, principů a systému jejího fungování. Tato kapitola je zaměřena zejména na popis uživatelského prostředí a způsob ovládání z pohledu různých členů redakce. Vytvořenou komponentu pro vědecký časopis ARNICA můžeme užívat celkem třemi pohledy, které jsou charakterizovány jednotlivými rolemi v redakci časopisu. Pro zpřístupnění nabídky obsluhující komponentu slouží přihlášení do webových stránek. To umožňuje vestavěné řešení od systému Joomla!.

4.1 VYUŽITÍ KOMPONENTY Z POHLEDU AUTORA

Po přihlášení na webových stránkách se autorovi zobrazí nabídka „Redakce“ s podnabídkou „Autor“. Tato nabídka obsahuje jednotlivé položky: „Odevzdat nový článek“, „Vrácené články“ a „Moje články“. Na obrázku 7 – Rozložení webové stránky po přihlášení je znázorněna webová stránka časopisu po přihlášení oponenta. V levém sloupci je patrná nabídka pro obsluhu komponenty z role oponenta.



Obrázek 7 – Rozložení webové stránky po přihlášení

4.1.1 ODEVZDÁNÍ NOVÉHO ČLÁNKU

Pro odevzdání nového článku slouží autorovi formulář, jeho ukázkou můžeme vidět na obrázku 8 – Formulář pro odevzdávání nových článků. Tento formulář je přístupný pod

odkazem nabídky „Odevzdat nový článek“. Pro odevzdání článku je nutné vyplnit název odevzdávaného článku, a v dalším poli vybrat soubor se samotným článkem. Formulář je vytvořen tak aby nebylo možné odevzdat článek bez názvu.

Obrázek 8 – Formulář pro odevzdávání nových článků

4.1.2 ODEVZDÁNÍ NOVÉ VERZE ČLÁNKU

Pod odkazem „Vracené články“ v nabídce „Redakce“ se vypisuje seznam článků, které mají být přepracovány na základě posudků, které ke článku oponenti zaslali. Tlačítko „odevzdat novou verzi článku“, které náleží každému vypsánému článku, přesměruje prohlížeč na novou podstránku. Na této podstránce je formulář, podobný formuláři pro odevzdání nového článku, avšak pouze s jediným polem. Toto pole slouží pro odevzdání souboru nové verze článku. Název článku již tedy není třeba vyplňovat.

Obrázek 9 – Část kompletního výpisu článků autora

4.1.3 PŘEHLED ČLÁNKŮ

Autor si může nechat vypsát všechny své odevzdané články. K tomuto výpisu je přístup prostřednictvím odkazu „Moje články“ v nabídce „Redakce“. Zlomek takového výpisu ilustruje obrázek 9 – Část kompletního výpisu článků autora. Do výpisu jsou zahrnuty všechny články a jsou barevně rozlišeny podle toho, v jakém stavu publikace se nacházejí. Pro dodatečnou orientaci je každý článek doprovázen textovou informací o jaký stav se jedná. Tento přehled zobrazuje i položky, které jsou shodné s většinou výpisů článku v komponentě. Jedná se především o název článku, datum odevzdání, dále pak o

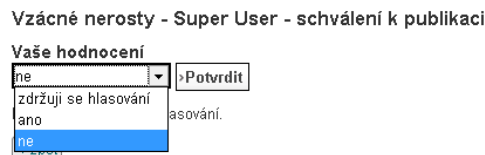
možnost stáhnout si poslední, ale i předchozích verze článku nebo zobrazit existující posudky oponentů pro daný článek.

4.2 VYUŽITÍ KOMPONENTY Z POHLEDU OPONENTA

Jakmile se přihlásí uživatel s právy oponenta, zobrazí se nabídka „Redakce“ s podnabídkou „Oponent“. V této podnabídce jsou dále dvě volby, a to „Přiřazené články“ a „Moje posudky“.

4.2.1 ODEVZDÁNÍ POSUDKU

Zhodnocení autora článku, v podobě odevzdání posudku probíhá na stránce „Přiřazené články“. Pod nadpisem *Napsat posudek* se nachází výpis článků, které byly přiřazeny přihlášenému autorovi pro posouzení. Jak takový výpis může vypadat, je znázorněno na obrázku 7 – Rozložení webové stránky po přihlášení. U každého z vypsaných článků nalezneme odkaz „napsat posudek“, který nás zavede k formuláři. V tomto formuláři jsou dvě pole. Jedno slouží pro výběr souboru posudku pro nahrání na server. Druhé pole pak slouží jako stručné hodnocení vyplývající z posudku a může nabývat hodnot „doporučuji“, „doporučuji s výhradami“ anebo „nedoporučuji“.



Obrázek 10 – Formulář pro hlasování redakční rady

4.2.2 HLASOVÁNÍ REDAKČNÍ RADY

Hlasování redakční rady o publikaci článku probíhá pod stejnou položkou nabídky jako odevzdávání posudků, tedy pod položkou „Přiřazené články“. V tomto případě se ale zaměříme na výpis článku, který je pod nadpisem *Články zaslané redakční radě*. V tomto výpisu je přidáno ke každému článku tlačítko „hlasovat o schválení“. Tlačítko přesměrovává prohlížeč na podstránku s formulářem, který soužijí k vyjádření souhlasu či nesouhlasu, zdali se má daný článek publikovat. Ve formuláři je pouze jedno pole, ve kterém uživatel vybere, jestli s publikací souhlasí. Možnosti výběru pole formuláře jsou „ano“, „ne“ anebo „zdržuji se hlasování“. Formulář s rozvinutou nabídkou hlasování je znázorněn na obrázku 10 – Formulář pro hlasování redakční rady. Po odhlasování

souhlasu či nesouhlasu s publikací článku může oponent své hlasování měnit, to ovšem jen do té doby než je článek publikán.

4.2.3 PŘEHLED POSUZOVANÝCH ČLÁNKŮ A VLASTNÍCH POSUDKŮ

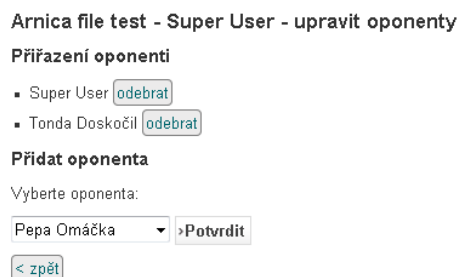
Druhá položka nabídky sloužící pro oponenty je „Moje posudky“. Tento odkaz zobrazí stránku obsahující kompletní výpis všech článků, ke kterým oponent psal posudky. Tento výpis je řazen chronologicky od nejnovějšího článku a nabízí základní zobrazení článku, kde si oponent může projít posudky nebo jednotlivé verze článku.

4.3 VYUŽITÍ KOMPONENTY Z POHLEDU ŠÉFREDAKTORA

Protože role šéfredaktora je velmi komplexní, má i nabídka pro tuto roli nejvíce položek. Jedná se o položky: „Články ke schválení“, „Schválené články“, „Zamítnuté články“ a „Publikované články“.

4.3.1 PŘIŘAZENÍ OPONENTŮ

Pokud je článek odevzdán, pak první operací v rámci systému, kterou s ním šéfredaktor provede je, že jej přidělí oponentům, kteří článek následně posoudí. K tomuto úkolu se šéfredaktor dostane přes položku nabídky „Články ke schválení“. Na zobrazené stránce je několik výpisů článků, těm kterým nebyli přiřazeni oponenti, jsou v části s nadpisem *Články bez oponentů*. K přidávání oponentů se šéfredaktor dostane přes odkaz „upravit oponenty“. Tento odkaz není jen v prvním výpisu zaměřeném na články bez oponentů, ale můžeme se k němu vracet i v rámci celé stránky „Články ke schválení“.



Obrázek 11 – Úprava oponentů

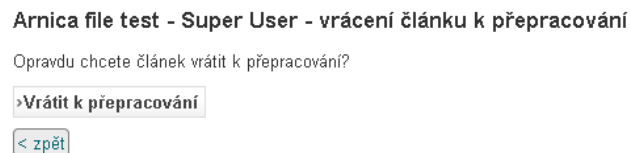
Podstránka sloužící pro úpravu oponentů je rozdělena na dva segmenty, těmi jsou *Přiřazení oponenti* a *Přidat oponenta*. Příklad podstránky úpravy oponentů je zobrazen na obrázku 11 – Úprava oponentů. V segmentu *Přiřazení oponenti* se nachází výpis všech přiřazených oponentů s možností jednotlivého odebrání. Pokud šéfredaktor přidává oponenty pro nové články, bude tento výpis samozřejmě prázdný. Do následujícího

segmentu s nadpisem *Přidat oponenta* je zakomponován formulář s jedním výběrovým políčkem, ve kterém jsou na výběr všichni oponenti, kteří nebyli přiřazeni pro tento článek. Můžeme tak snadno jednotlivé oponenty přidávat, nebo odebírat.

Pokud bude ke článku přiřazen alespoň jeden oponent, přesune se na stránce „Články ke schválení“ z prvního výpisu článku *Články bez oponentů* do dalšího výpisu s nadpisem *Posuzované články*. V tomto výpisu článek zůstane do té doby, než všichni oponenti, kteří byli ke článku přiřazeni, neodevzdají své posudky.

4.3.2 SCHVALOVÁNÍ ČLÁNKŮ

Schvalování článků probíhá pod volbou „Články ke schválení“ v nabídce redakce. Pokud ke článku napíší všichni oponenti posudky, článek se přesune ze sekce *Posuzované články* do oblasti s nadpisem *Posouzené články*. V této oblasti jsou u článků tři speciální volby, jedná se o „schválit článek“, „vrátit k přepracování“ a „zcela zamítnou článek“. Při zvolení jakékoli z těchto voleb je šéfredaktor přesunut na formulář, který po uživateli požaduje potvrzení operace. Potvrzující formulář můžete vidět na obrázku 12 – Formulář potvrzení *vracení k přepracování*.



Obrázek 12 – Formulář potvrzení *vracení k přepracování*

Pokud zvolíme „schválit článek“, článek se přesune do výpisu, který se nachází pod volbou „Schválené články“ v nabídce redakce. V případě volby možnosti „vrátit k přepracování“ bude článek navrácen autorovi a šéfredaktor tak čeká na novou verzi tohoto článku. Z pohledu šéfredaktora jsou tyto články vypisovány v poslední sekci s názvem *Články čekající na novou verzi* na stránce přístupné z nabídky redakce pod odkazem „Články ke schválení“. Poslední možnost „zcela zamítnou článek“ přesune článek do výpisu, ke kterému se šéfredaktor může dostat pomocí volby „Zamítnuté články“ v redakční nabídce. Článek, který se dostane do toho výpisu, je již v konečném stavu a nelze jím již dále nijak přesouvat.

4.3.3 PUBLIKOVÁNÍ ČLÁNKŮ

Aby se mohl schválený článek publikovat, musí být zaslán redakční radě, která hlasuje o jeho publikaci. Toho docílíme pomocí odkazu „zaslat redakční radě ke schválení pro publikaci“, který se nachází ve výpisu článků v sekci *Schválené články* na stránce přístupné z volby „Schválené články“ v nabídce redakce. Jakmile je článek zaslán redakční radě, přesune se z tohoto výpisu do dalšího výpisu, který je označen nadpisem *Články zasláné redakční radě* a je na téže stránce. Pomocí odkazu „výsledky hlasování rady“ v tomto výpisu článků se šéfredaktor dostane na podstránku s výpisy hodnocení celé redakční rady. Jak takový výpis vypadá, je zřejmé z obrázku 13 – Přehled hodnocení redakční rady. Jakmile šéfredaktor uváží, že se vyjádřil dostatečný počet členů redakční rady, může článek publikovat anebo zcela zamítnout. Článek, který je publikovaný, je stejně jako zamítnutý článek v koncovém stavu a již se nedá dále přesouvat. Výpis všech publikovaných článků je přístupný přes odkaz „Publikované články“ z redakční nabídky.

Vzácné nerosty - Super User - výsledky hlasování redakční rady

- Super User - ne
- Tonda Doskočil - zatím nehodnotil
- Pepa Omáčka - ano
- Honza Vyklobený - zdrzuje se hlasování

Přehled hodnocení redakční rady

ano: 1
 ne: 1
 nehodnotilo: 1
 zdržují se: 1

Pozor: Nevyjádřila se celá redakční rada.

nebo

Obrázek 13 – Přehled hodnocení redakční rady

ZÁVĚR

Bakalářská práce analyzovala systém redakce vědeckého časopisu ARNICA z pohledů potřebných pro vytvoření rozšíření, které bude efektivně napomáhat členům redakce při procesu publikace článků. Rozbor vnitřního systému redakce se týkal dvou pohledů. Tím prvním bylo rozdělení rolí v reakci, které jednotliví zaměstnanci zastupují. Další pohled byl pak zaměřen na životní cyklus článku, kterým jednotlivý článek musí projít proto, aby byl publikován.

Webové stránky časopisu ARNICA byly postaveny na robustním redakčním systému Joomla!. V bakalářské práci byl uveden obecný popis redakčních systémů spolu s CMS, pod které systém Joomla! spadá. Následně byly popsány hlavní části, jako je instalace, vytvoření nabídek nebo jazykové mutace, které byly použity u webové prezentace vědeckého časopisu ARNICA.

V hlavní části bakalářské práce byly uvedeny pravidla a omezení, které s sebou tvorba rozšíření – komponenty nese. Jednalo se zejména o adresářovou strukturu a podmíněné systémové soubory. Představili jsme si mechanismy frameworku systému Joomla!, které jsou pro tvorbu vlastních rozšíření velkým přínosem. Dále jsme si uvedli klíčové principy a postupy, které byly použity při vlastní tvorbě. Následně byla popsána i databázová struktura, která figuruje na pozadí celé komponenty.

Výstupem bakalářské práce byla komponenta pro systém Joomla!, která ulehčuje publikaci článků v rámci vědeckého časopisu ARNICA. Práce byla zastřešena popisem vytvořené komponenty. Popis byl zaměřen na jednotlivé způsoby, jak může být komponenta používána a obsluhována. Ty jsou rozděleny na základě rolí v redakci. Jedná se o využívání z pohledu autora, oponenta a šéfredaktora. Jednotlivé využití bylo popsáno tak, že i uživatel, jenž nezná principy ani vnitřní mechanismy, na kterých je komponenta postavena, by byl schopen po jeho přečtení s rozšířením bez problému pracovat.

SEZNAM OBRÁZKŮ A TABULEK

SEZNAM OBRÁZKŮ

Obrázek 1 – Životní cyklus článku.....	3
Obrázek 2 – Instalátor systému Joomla!.....	9
Obrázek 3 – Administrace systému Joomla!	10
Obrázek 4 – Formulář pro správu položek menu	12
Obrázek 5 – Správce jazyků	14
Obrázek 6 – Adresářová struktura komponenty	17
Obrázek 7 – Rozložení webové stránky po přihlášení	31
Obrázek 8 – Formulář pro odevzdávání nových článků.....	32
Obrázek 9 – Část kompletního výpisu článků autora	32
Obrázek 10 – Formulář pro hlasování redakční rady	33
Obrázek 11 – Úprava oponentů	34
Obrázek 12 – Formulář potvrzení <i>vracení k přepracování</i>	35
Obrázek 13 – Přehled hodnocení redakční rady	36

SEZNAM TABULEK

Tabulka 1 – Přehled stavů článku.....	21
Tabulka 2 – Hodnoty sloupce <i>hodnoceni</i> z tabulky <i>clanky_posudek</i>	22
Tabulka 3 – Hodnoty sloupce <i>hodnoceni</i> z tabulky <i>clanky_redakce</i>	22

SEZNAM LITERATURY

1. RAHMEL, D. *Joomla!: Podrobný průvodce tvorbou a správou webů*. Brno: Computer Press, 2010. ISBN 978-80-251-2714-8.
2. The Joomla! Extensions Directory. *Joomla!* [online]. 2005 - 2012 [cit. 2012-03-04]. Dostupné z: <http://extensions.joomla.org/>
3. DEMKO, C. a O. AKSU. Joomla! Documentation. In: *Developing a Model-View-Controller (MVC) Component for Joomla!2.5 - Part 02* [online]. 19. 4. 2012 [cit. 2012-06-05]. Dostupné z: [http://docs.joomla.org/Developing_a_Model-View-Controller_\(MVC\)_Component_for_Joomla!2.5_-_Part_02](http://docs.joomla.org/Developing_a_Model-View-Controller_(MVC)_Component_for_Joomla!2.5_-_Part_02)
4. OPEN SOURCE MATTERS, I. Why do most of the Joomla! PHP files start with "defined('JEXEC')...? In: *Joomla! Documentation* [online]. 22. 5. 2010 [cit. 2012-06-05]. Dostupné z: [http://docs.joomla.org/Why_do_most_of_the_Joomla!_PHP_files_start_with_%22defined\('JEXEC'\)...%3F](http://docs.joomla.org/Why_do_most_of_the_Joomla!_PHP_files_start_with_%22defined('JEXEC')...%3F)
5. DEMKO, C. a O. AKSU. Joomla! Documentation. In: *Developing a Model-View-Controller (MVC) Component for Joomla!2.5 - Part 04* [online]. 13. 2. 2012 [cit. 2012-06-13]. Dostupné z: [http://docs.joomla.org/Developing_a_Model-View-Controller_\(MVC\)_Component_for_Joomla!2.5_-_Part_04](http://docs.joomla.org/Developing_a_Model-View-Controller_(MVC)_Component_for_Joomla!2.5_-_Part_04)
6. DEMKO, C. a O. AKSU. Joomla! Documentation. In: *Developing a Model-View-Controller (MVC) Component for Joomla!2.5 - Part 05* [online]. 16. 5. 2012 [cit. 2012-06-13]. Dostupné z: [http://docs.joomla.org/Developing_a_Model-View-Controller_\(MVC\)_Component_for_Joomla!2.5_-_Part_05](http://docs.joomla.org/Developing_a_Model-View-Controller_(MVC)_Component_for_Joomla!2.5_-_Part_05)
7. ŽMIJOVSKÝ, M. *Joomla!: Sbíрка nejlepších řešení a postupů pro váš web*. Brno: Computer Press, a.s. 2011. ISBN 978-80-251-3230-2.
8. LACKO, L. *PHP 5 a MySQL 5: Hotová řešení*. Brno: Computer Press, a. s. 2007. ISBN 978-80-251-1695-1.

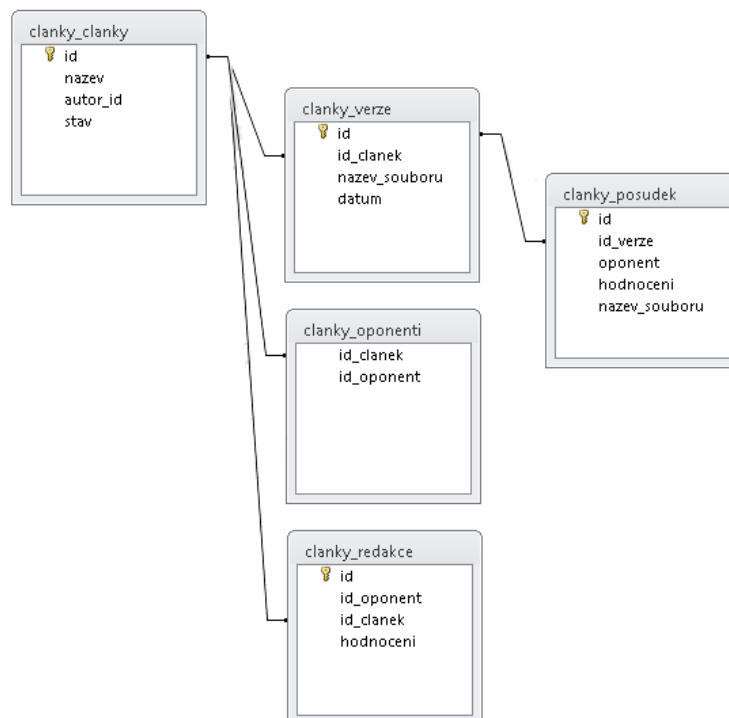
RESUMÉ

The bachelor thesis describes the functioning system in the editorial office of the science magazine ARNICA. The analysis of this system is primarily focused on positions in the editorial office, that are held by the staff and also on the life cycle of an article. The next part of thesis introduces the Joomla! system and the possibility of its application within the website of ARNICA magazine. This part of thesis also includes restrictions and rules, that have to be followed during the process of web extension.

Considering the analysis of the functioning system in the editorial office of the science magazine ARNICA and stated information about Joomla! system, the main part of thesis is focused on the creation of extension itself, that serves as a useful help when articles in the editorial office are being submitted, evaluated and published. The bachelor thesis also deals with some integrated functions and framework classes of the Joomla! system, that are together with principles and methods used in the process of extension creation. The last part of thesis is dedicated to the description of created extension from the viewpoint of users. These viewpoints are influenced by the positions in the editorial office of ARNICA magazine.

PŘÍLOHY

A. RELACE TABULEK KOMPONENTY COM_CLANKY



B. KOMPLETNÍ ADRESÁŘOVÁ STRUKTURA KOMPONENTY COM_CLANKY

