SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE FAKULTA CHEMICKEJ A POTRAVINÁRSKEJ TECHNOLÓGIE

EVIDENČNÉ ČÍSLO: FCHPT-5415-55812

TVORBA WEBSTRÁNOK ŠVK V REDAKČNOM SYSTÉME DRUPAL

BAKALÁRSKA PRÁCA

Martin Šimek

2015

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE FAKULTA CHEMICKEJ A POTRAVINÁRSKEJ TECHNOLÓGIE

TVORBA WEBSTRÁNOK ŠVK V REDAKČNOM SYSTÉME DRUPAL

BAKALÁRSKA PRÁCA

FCHPT-5415-55812

Študijný program: automatizácia, informatizácia v chémii a potravinárstve Číslo študijného odboru: 2621 Názov študijného odboru: 5.2.14 automatizácia, 5.2.52 priemyselné inžinierstvo Školiace pracovisko: Ústav informatizácie, automatizácie a matematiky Vedúci záverečnej práce/školiteľ: Ing. Ľuboš Čirka, PhD.

Bratislava, 2015

Martin Šimek

Slovenská technická univerzita v Bratislave Ústav informatizácie, automatizácie a matematiky Fakulta chemickej a potravinárskej technológie Akademický rok: 2014/2015 Evidenčné číslo: FCHPT-5415-55812



ZADANIE BAKALÁRSKEJ PRÁCE

Študent:	Martin Šimek
ID študenta:	55812
Študijný program:	automatizácia, informatizácia a manažment v chémii a potravinárstve
Kombinácia študijných odborov:	5.2.14. automatizácia, 5.2.52. priemyselné inžinierstvo
Vedúci práce:	Ing. Ľuboš Čirka, PhD.

Tvorba web stránok ŠVK v redakčnom systéme Drupal

Špecifikácia zadania:

Názov práce:

Cieľom práce je vytvorenie web stránok ŠVK v redakěnom systéme Drupal. Drupal je nástroj na tvorbu webových stránok, ktorý umožňuje ľahké vytvorenie, aktualizáciu a spravovanie webových stránok bez nutnosti rozsiahlych technických vedomostí. Jeho architektúra poskytuje ohromnú flexibilitu, a pritom umožňuje aj ľudom bez znalosti programovania postaviť pomerne rozsiahle a komplexné weby.

Úlohy: naštudovať architektúru redakčného systému, nainštalovať a nakonfigurovať redakčný systém, naštudovať tvorbu vlastných modulov, vykonať analýzu súčasných stránok, vytvoriť novú verziu stránok ŠVK v Drupale s využitím štandardných, externých a vlastných modulov.

Rozsah práce:

Zoznam odbornej literatúry:

- 1. Buytaert, D. Drupal. [online]. 2011. URL: http://drupal.org/.
- 2. Polzer, J. 333 tipů a triků pro Drupal. Brno: Computer Press, 2010. 264 s. ISBN 978-80-251-2942-5.
- 3. Polzer, J. Drupal : Podrobný průvodce tvorbou a správou webů. Brno: Computer Press, 2008. 262 s.
- 4. Garfield, L. Dunlap, G. Rickard, K. Drupal: Programování modulů. Praha: Computer Press, 2011. 360 s.

ISBN 978-80-251-3440-5. Riešenie zadania práce od: 16. 02. 2015

30

Dátum odovzdania práce: 24. 05. 2015

L. S.

Martin Šimek študent

prof. Ing. Miroslav Fikar, DrSc. vedúci pracoviska prof. Ing. Miroslav Fikar, DrSc. garant študijného programu

Pod'akovanie

Ďakujem pánovi Ing. Ľubošovi Čirkovi, PhD. za pripomienky, odborné rady, čas a trpezlivosť pri tvorbe záverečnej práce.

Abstrakt

Bakalárska práca sa zaoberá tvorbou modulu do redakčného systému Drupal. Prvá kapitola je zameraná na analýzu súčasného stavu stránok Študentskej vedeckej konferencie. V druhej kapitole je opísaná inštalácia lokálneho serveru, databázy MySQL, skriptovacieho jazyka PHP ako jedného balíčka zvaného WAMP, pretože je to nutné pre fungovanie Drupalu. Hlavnou témou zvyšnej časti kapitoly je nainštalovanie a nastavenie Drupalu a opis typov modulov. Tretia kapitola je zameraná na základnú štruktúru tvorby modulu a objasňuje správanie zdrojového kódu počas inštalácie a fungovaní jadra modulu. Koniec kapitoly zahŕňa inštaláciu a konfigurovanie modulu.

Kľúčové slová: Drupal; Modul; Redakčný systém

Abstract

The thesis deals with the creation of the module to Content Management System Drupal. The first part focuses on the analysis of current state of the Student scientific conference website. The second chapter describes the installation of the so called WAMP, a package consisting of a local server, a MySQL database and the scripting language PHP, as they are all necessary for Drupal to function. The main topic of the remaining part of the chapter is installing and setting Drupal and the description of the module types. The third chapter focuses on the basic structure of creating a module and clarification of the behaviour of the source code during installation and the operation of the module core. The end of chapter covers installation and configuration of the module.

Key words: Drupal; Module; Content Management System

Obsah

Ú١	vod	
1	Súčasný	stav
2	Redakčn	ý systém 14
	2.1 We	bserver, PHP a MySQL14
	2.1.1	Inštalácia WAMP15
	2.2 Dru	ıpal16
	2.2.1	Inštalácia Drupalu16
	2.2.2	Moduly jadra
	2.2.3	Externé moduly
	2.2.4	Vlastné moduly
3	Tvorba v	vebu ŠVK21
	3.1 Tvo	orba vlastného modulu
	3.1.1	Funkcie inštalácie modulu22
	3.1.2	Funkcie jadra modulu27
	3.1.3	Inštalácia a konfigurácia modulu
4	Diskusia	
5	Záver	
Zo	znam použ	itej literatúry40
Pri	ílohy	

Úvod

Súčasný rozvoj techniky je tak pokročilý, že čo bola pred pár rokmi zdĺhavá či nemysliteľná činnosť, je dnes takpovediac hračka. Väčšinu potrebných informácií už nájdeme na internete. K tomu slúžia webové stránky. Vlastniť webovú stránku už nie je záležitosť nutnej potreby vedieť jazyk HTML. Existuje veľa druhov voľne dostupných redakčných systémov, ktoré majú v sebe zabudované rôzne funkcie na vytváranie a editovanie stránok. Ich charakteristickými črtami sú: rozšíriteľnosť, jednoduchosť, bezpečnosť a hlavne sú zadarmo. Nič však nie je ideálne, preto aj keď redakčné systémy ponúkajú nespočetné množstvo rozšírení, sú situácie kedy sa nezaobídu bez zásahu programovania.

Postup riešenia tejto práce spočíva v analýze súčasného stavu stránok Študentskej vedeckej konferencie, navrhnutí vylepšení v podobe modulu do redakčného systému Drupal. Hlavnou zbraňou modulu je vytvorenie registračných formulárov, špecifických stránok a narábanie s databázou pre ukladanie alebo vypisovanie údajov.

Prínosom práce je prehĺbenie znalostí v oblasti tvorby modulov do Drupalu, spoznanie logiky fungovania systémových súčastí a najmä splnenie požiadaviek kladených na modul.

1 Súčasný stav

Stránky Študentskej vedeckej konferencie (ďalej len ŠVK) sú v súčasnosti určené na poskytovanie informácií pre účastníkov súťaže vrátane ich pred registrácie a ubytovania (obr. 1). Niektoré stránky sú skryté pred bežnými návštevníkmi a slúžia ako zdroj pre spracovávanie údajov. Zber údajov je riešený cez modul Webform, ktorý slúži na vytváranie formulárov a ukladanie ich údajov bez znalosti programovania. Zbieranie údajov prebieha cez stránky *Registrácia príspevkov* a *Ubytovanie*. Na prácu s údajmi sú použité vlastné skripty implementované do každej stránky zvlášť, pretože sa kladú rozličné požiadavky na výstup a zobrazenie dát. Niektoré časti kódu sú však z programátorského hľadiska riešené pracne a neefektívne. Hlavný dôraz sa teda kladie na jednoduchší zber a spracovanie dát v podobe modulu.

Po dôkladnej analýze stavu stránok ŠVK a dostupných schopností programovania bude výsledný modul spracovávať údaje z/do vlastných databázových tabuliek.



Obr. 1 Vzhľad súčasných stránok ŠVK

2 Redakčný systém

[1]

Redakčný systém (ďalej len RS) možno chápať ako sadu nástrojov na obsluhu a spracovávanie obsahu. Označujeme ich ako CMS (Content Management System). Ak sa zameriame na RS určený pre webové stránky, medzi najbežnejšie funkcie patrí: tvorba stránok, správca multimédií, interné nastavenia systému, moduly a vzhľad. Sú navrhnuté tak, aby si webové stránky mohol vytvoriť aj človek bez znalostí programovania. Na svete existuje veľa druhov RS. Môžeme ich však rozdeliť na platené alebo voľne dostupné. Platené sú vytvorené užívateľovi na mieru a presne podľa jeho požiadaviek. Voľne dostupné majú základnú funkcionalitu na správu webu, pričom sa dajú rozšíriť o moduly, ktoré z nich robia silný nástroj. Zoznam najpoužívanejších a voľne dostupných RS, ktoré majú veľkú podporu zo strany vývojárov:

- Drupal jednoduché administračné rozhranie, rozsiahla ponuka funkcií, prepracované zabezpečenie voči útokom. Webstránka: www.drupal.org
- Joomla funkcionalitou podobný Drupalu, administračné rozhranie pomerne komplikovanejšie a neprehľadné.
 Webstránka: www.joomla.org

 Wordress – jednoduchý RS, ktorý je skôr vhodný na osobné, prezentačné weby a jednoduché projekty.

Webstránka: www.wordpress.com

2.1 Webserver, PHP a MySQL

Drupal je vytvorený v skriptovacom jazyku PHP a pre prácu s údajmi využíva databázu (MySQL, PostgreSQL, ...). Preto potrebuje lokálny server s podporou PHP a databázy.

Aby sme v bežnom počítači mohli simulovať funkciu servera a pustiť sa do inštalácie a používania Drupalu, stiahneme si z webovej lokality *http://www.wampserver.com/en/* inštalačný balíček WAMP server (Windows, Apache, Mysql, Php). Je určený pre operačný systém Windows a požiadavkám plne vyhovuje. Výhodou tohto balíka je, že nie sú potreba dodatočné zdĺhavé a komplikované nastavenia programových súčastí medzi sebou ako v prípade jednotlivého inštalovania Apache, PHP a MySQL.

2.1.1 Inštalácia WAMP

- 1. Spustíme si inštalačný súbor, ktorý sme si stiahli z vyššie uvedenej webovej adresy.
- 2. Potvrdíme prípadne bezpečnostné upozornenia na spustenie súboru.
- 3. Pokračujeme podľa inštrukcií programu, čiže ako vo väčšine prípadov klikáme na **Next** a na začiatku odsúhlasíme licenciu.
- 4. Vyberieme si zložku kam sa nainštaluje webserver a podporné programy.
- 5. Program sa opýta, aký webový prehliadač bude preň ako predvolený.
- 6. V ďalšom kroku sa nastavuje SMTP server. Ponecháme bezo zmeny.
- 7. Potvrdíme povolenia na používanie programu ak nás systém vyzve na odblokovanie.
- 8. Dokončíme inštaláciu a pre istotu reštartujeme počítač. Po reštartovaní si spustíme program *start WampSserver*.

Ostáva zistiť hlavnú zložku, ktorú WAMP určil ako miesto webových projektov. Všetky súbory alebo zložky, ktoré sa v nej budú nachádzať, vrátane neskôr skopírovaných zdrojových súborov Drupalu, budú určené na spracovanie lokálnym serverom.

Postup ako zistíme zložku dokumentov:

- 1. Klikneme na zelenú ikonku 🔟 (v lište v pravom dolnom rohu) pravým tlačidlom myši.
- 2. Klikneme na www adresár (obr. 2).
- 3. V otvorenom adresári si budeme neskôr vytvárať zložku kde bude umiestnený RS.
- Ak miesto zelenej ikony je oranžová alebo červená, programové časti WAMPu nespolupracujú a je nutné prekontrolovať nastavenia alebo výpisy chýb – Logy. Častou chybou býva využitie portu iným programom (Skype,...)



Obr. 2 Menu WAMP servera

2.2 Drupal

Drupal je nástroj používaný na tvorbu webových stránok a považuje sa za jeden z najlepších RS. Má veľkú podporu zo strany používateľov a vývojárov pričom je úplne zadarmo. Funkcionalita sa dá rozšíriť viac než z desaťtisíc modulov.

Stránky ŠVK sú vytvorené pre RS Drupal, preto je nutné oboznámiť sa so základnou architektúrou a fungovaním systémových častí. Z webovej lokality *https://www.drupal.org/download* si stiahneme stabilnú verziu jadra začínajúcu číslom 7, pretože budúci modul vytvorený pre ŠVK je určený pre verziu jadra Drupal 7.

2.2.1 Inštalácia Drupalu

Po úspešnej inštalácii webservera a jeho súčastí nám ostáva inštalácia samotného redakčného systému. Z webovej lokality sme si stiahli stabilnú verziu jadra Drupalu 7 a obsah extrahovali do dočasnej zložky v počítači. Keď chceme mať rozhranie redakčného systému v inom jazyku ako anglickom, stiahneme si dodatočne zvolený súbor pre náš jazyk zo stránky *https://localize.drupal.org/translate/languages/sk*. Otvoríme si zložku (*www adresár*) na projekty, ktoré bude spracovávať náš lokálny server.

Pre jednoduchosť budeme cestu ku zložke, kde sú všetky webové dokumenty, označovať ako *root* a vo webovom prehliadači ako webovú adresu *localhost*.

V prípade, že chceme vyvíjať aj iné webové projekty, vytvoríme si ľubovoľnú zložku, najlepšie bez diakritiky a medzier. V našom prípade *SVK*. Otvoríme si ju a nakopírujeme obsah zložky, kde sme si dočasne extrahovali archív Drupalu ako je na obr 3.



Obr. 3 Súborový systém Drupalu

Spustíme si ľubovoľný webový prehliadač a zadáme adresu *localhost/phpmyadmin*. PHPMyAdmin je webová aplikácia umožňujúca ľahko vytvárať, upravovať a mazať databázy, ich tabuľky alebo upravovať dáta. Vytvoríme si databázu s ľubovoľným názvom bez diakritiky a medzier. V našom prípade *svk*, s kódovaním *utf8_general_ci*. Do tejto databázy si Drupal nahrá všetky tabuľky potrebné pre svoj chod.

Môžeme pristúpiť k samotnej inštalácií RS. V spustenom prehliadači zadáme adresu *localhost/svk*. Automaticky sa nám zobrazí inštalačné rozhranie. V inštalácií postupujeme podľa nasledujúcich krokov:

1. Choose profile

Ponecháme Standard profile, uložíme a pokračujeme ďalej.

2. Choose language

V základnej verzii drupalu je anglický jazyk. Keď chceme mať rozhranie RS v inej jazykovej mutácií nakopírujeme stiahnutý súbor jazyka do zložky *profiles/standard/translations/* a znovu načítame stránku stlačením klávesy *F5* aby sa obnovil zoznam jazykových prekladov. Zvolíme preferovaný jazyk, v našom prípade ponecháme angličtinu, uložíme zmeny a pokračujeme ďalej.

3. Set up database (obr. 4)

V tomto kroku potrebujeme vedieť základné údaje na pripojenie ku databáze. Predpokladáme, že základné údaje na pripojenie sú: *Používateľ:* root, *Heslo:* nevyplnené, *Server:* localhost, *Databáza:* svk. Database type je MySQL, Database name: svk, Database username: root, Database password: necháme prázdne, otvoríme Advanced options a do Database host napíšeme localhost.

Z hľadiska bezpečnosti sa odporúča nastaviť databázové heslo konfiguračnom súbore MySQL *my.ini*. Pre testovacie účely, kde nehrozí bezpečnostné riziko, nie je potrebné.

Predpokladáme, že databáza *svk* bude len pre naše stránky ŠVK. Ak by sme mali databázu, kde sa nachádzajú aj iné tabuľky pre rôzne projekty, je vhodné vyplniť kolónku *Table prefix*, ktorá tabuľkám dodáva rozpoznateľnú predponu. Uložíme zmeny a pokračujeme ďalej.

4. Install profile

Tu nastavíme základne informácie o stránke a údaje. Do *Sites name* napíše ako príklad: Stránky študentskej vedeckej konferencie, *Site e-mail adress* vyplníme informačnú e-mailovú adresu, nemusí byť pravdivá, ak nám nefunguje na webserveri odosielanie e-mailov, za *Username* zvolíme ako administrátorský účet: *Admin, Email-adress:* e-mailová adresa admina, *Password:* admin, *Confirm password:* potvrdíme to isté heslo: admin, *Default country:* Slovakia, *Default time zone:* Europe/Paris, odškrtneme aktualizácie *Check for updates automatically.* Uložíme a pokračujeme ďalej.

5. Finished

Drupal je nainštalovaný a pripravený na prvé použitie

Database configuration

Database type *
MySQL, MariaDB, or equivalent
SOLite
The type of database your Drupal data will be stored in
The type of database your brapar data will be stored in.
Database name *
svk
The name of the database your Drupal data will be stored in. It must exist on your server before Drupal can be installed
before brupai can be instancu.
Database username *
root
Database password
••••
→ ADVANCED OPTIONS
These options are only necessary for some sites. If you're not sure what you
should enter here, leave the default settings or check with your hosting provider.
Database host *
localhost
If your database is located on a different server, change this.
Detailers and
Database port
If your database server is listening to a non-standard port, enter its number.
Table prefix
If more than one application will be sharing this database, enter a table prefix such as
drupal_ for your Drupal site here.

Obr. 4 Ukážka inštalácie Drupalu pri nastavení databázových údajov

2.2.2 Moduly jadra

Drupal ako súborový celok pozostáva z rôznych zložiek a skriptov, ktoré z neho robia komplexný RS. Treba však vedieť, ktoré zložky, či súbory, by mali byť pre bežného používateľa tabu a nespôsobiť si tak škodu v podobe nefunkčnosti systému alebo stratených dát. Funkčnosť celého systému zabezpečujú do veľkej miery moduly jadra. Nachádzajú sa v zložke *modules* a všetko čo je tam, musí ostať neporušené bez zmeny. Každý modul ma inú úlohu. Kým jeden sa stará o užívateľov, druhý treba zabezpečuje správny chod a komunikáciu s databázou. Ich detaily

nájdeme *localhost/admin*, sekcia *Modules* v kategórii *Core*. Niektoré sa dajú vypnúť, iné nie. Odstrániť sa však nedajú. Ak by sme chceli rozšíriť náš redakčný systém o iné moduly, treba si naprogramovať vlastný alebo stiahnuť z internetu externé moduly.

2.2.3 Externé moduly

Drupal ponúka aj iné možnosti rozšírení skrz iné ako systémové moduly. Na webovej lokalite *https://www.drupal.org/project/project_module* je viac než desaťtisíc overených modulov na stiahnutie. Výsledky hľadania sa dajú obmedziť na určité požiadavky, ako napríklad verzia jadra, kategorické zaradenie, názov modulu...

Inštalácia modulu prebieha priamym kopírovaním a nastavením v adminovi. Otvoríme zložku sites/all/modules/ a do nej nakopírujeme zložku s jedinečným názvom modulu. V prehliadači otvoríme *localhost/admin*, klikneme na *Modules* a hľadáme názov modulu, ktorý sme si nahrali do zložky. Označíme *checkbox* a uložíme zmeny. Modul by sa mal aktivovať, ak všetko prebehne v poriadku.

2.2.4 Vlastné moduly

Ak však potrebujeme naprogramovať modul podľa vlastných požiadaviek, je potreba sa naučiť skriptovací jazyk PHP na objektovej úrovni, vedieť zákonitosti práce s databázami a tabuľkami, napr. jazyk MySQL a základ tvorby webových stránok v HTML. Tým to však nekončí. Musíme sa naučiť logiku fungovania celého RS. Postup ako sa vytvára modul je cieľom tejto práce a je vysvetlený v ďalších kapitolách.

Vlastné moduly je možné aj publikovať pri splnení určitých podmienok a podporiť tak rozvoj Drupal komunity. V tom prípade sa z nášho vlastného projektu stane Externý modul dostupný pre všetkých.

3 Tvorba webu ŠVK

Z hľadiska súčasného stavu webových stránok ŠVK a požiadaviek nie je nutné vytvárať novú grafickú verziu stránok. Použijeme aktuálnu tému – vzhľad a zvyšok sa bude aplikovať prostredníctvom systémových funkcií RS Drupalu a vlastného modulu.

3.1 Tvorba vlastného modulu

[2][3][4][5]

Predtým než začneme niečo programovať potrebujeme vhodné vývojové prostredie. Z voľne dostupných nástrojov môžeme použiť *PSPad*. Pri tvorbe tejto práce bol použitý *Adobe Dreamweaver*.

Každý modul, ktorý sa rozhodneme vytvoriť, alebo použiť, musí pre správne fungovanie spĺňať určité kritériá. Vytvoríme si novú zložku v **root/sites/all/modules**/ s jednoduchým názvom budúceho modulu s použitím anglickej abecedy. Volíme **svk**. Podmienkou je vytvoriť si tri súbory s rovnakým názvom ako je zložka modulu a ako koncovky budú mať **.module**, **.install** a **.info**. Čiže **svk.module**, **svk.install** a **svk.info**. Zdrojové súbory celého modulu vidno na obr. 5.

svk.diplomy.a.potvrdenia.o.ucasti	🔤 svk.info	🔤 svk.install	svk.module
svk.program.inc	svk.registracia.prispevkov.inc	svk.sekcie.inc	svk.ubytovanie.inc
svk.uprava.programu.inc	svk.vysledky.inc	svk.zaradenie.do.sekcie.inc	svk.zoznam.prispevkov.inc
svk.zoznam.prispevkov.v.sekcii.inc	svk.zoznam.ubytovanych.inc		

Obr. 5 Zdrojové súbory modulu svk

svk.info – cieľom tohto zdrojového súboru je povedať redakčnému systému základné informácie o module. Podľa referenčnej príručky existuje veľa možností čo všetko použiť. Minimom je však názov modulu, verzia jadra a popis.

svk.install – vykonávajú sa rôzne definované operácie, keď sa modul inštaluje, odinštaluje, aktivuje či deaktivuje. Príkladom môže byť vytvorenie alebo odstránenie databázových tabuliek.

svk.modul - samotné srdce funkcionality modulu obsahujúce naše zdrojové kódy.

Tvorba modulov pre Drupal je založená na takzvaných **háčikoch** (anglicky *hook*). Nie je to nič iné ako bežná PHP funkcia, ktorá plní úlohu ako keby záchytného bodu a vykonáva sa pri určitej operácii. Skladá sa z názvu modulu a názvu špecifickej operácie. Napríklad podľa referenčnej príručky Drupalu **hook_install**() je volaná iba pri inštalácii modulu. Namiesto názvu

hook dáme názov nášho modulu, ktorý sa volá rovnako ako zložka kde je uložený. Výsledný názov funkcie je **svk_install**().

3.1.1 Funkcie inštalácie modulu

Aby vedel Drupal rozpoznať náš modul, je potreba definovať základné informácie do súboru **svk.info** pomocou unikátnych konštánt. Zoznam použiteľných konštánt a ich použitie je možné nájsť v referenčnej príručke. Každá konštanta musí byť na novom riadku.

Obsah nášho súboru svk.info tvoria nasledujúce riadky:

```
    name = ŠVK
    description = Modul pre študentskú vedecké konferenciu
    core = 7.x
    package = Študentská vedecká konferencia
    stylesheets[all][] = styl.css
    configure = admin/config/content/svk
```

- 1. name celé meno modulu
- 2. description obsahuje stručný popis funkcie a práce modulu
- 3. core pre ktorú verziu jadra je modul navrhnutý
- 4. package zaradenie modulu do určitej skupiny v administračnej časti, ktorá zoskupuje iné podobné moduly alebo iba modul samotný. My však máme vlastnú skupinu, preto bude náš modul v tejto skupine sám.
- **5.** stylesheets[all][] použitý súbor kaskádových štýlov pre všetky typy zobrazovacích zariadení, každý nový súbor kaskádového štýlu dávame na nový riadok
- 6. configure pridanie budúceho odkazu pre nastavenia modulu

Modul využíva pri svojej činnosti vlastné databázové tabuľky a vlastnú zložku **svk** určenú na nahrávanie abstraktov. Preto potrebujeme pri inštalácii modulu tieto tabuľky nahrať do databázy a vytvoriť zložku v root/sites/default/files/**svk**. Riešime to programovo. V prípade odinštalácie sa nám tieto tabuľky vymažú z databázy a tak isto zložka s abstraktmi. Využijeme hook_install(), _uninstall() a _schema(). Pri inštalácii sa najprv zavolá svk_schema(), ktorá vytvorí požadované tabuľky začínajúce prefixom svk_ – nastavenia, prispevky, registracie, sekcie, skoly, tituly, ubytovanie.

Ukážka zdrojového kódu vytvorenia štruktúry tabuľky svk_nastavenia:

```
1. function svk_schema() {
2. $schema = array();
3. $schema['svk_nastavenia'] = array(
     'description' => 'Zoznam zakladnych nastaveni',
4.
5.
     'fields' => array(
        'Id' => array(
6.
          'description' => 'Id premennej',
7.
          'type' => 'serial',
8.
          'not null' => TRUE,),
9.
        'Premenna' => array(
10.
          'description' => 'Nazov premennej',
11.
          'type' => 'varchar',
          'length' => '255',
12.
          'not null' => TRUE,),
13.
        'Hodnota' => array(
          'description' => 'Hodnota premennej',
          'type' => 'varchar',
          'length' => '255',
          'not null' => TRUE,),),
14.
    'primary key' => array('Id'),);
15. //.... Kód štruktúry ďalších tabuliek .....
16. return $schema;}
```

- 1. Háčik, ktorý združuje štruktúru DB tabuliek na ich inštaláciu alebo vymazanie
- 2. Inicializácia premennej \$schema typu pole
- 3. Definícia názvu DB tabuľky svk_nastavenia
- 4. Stručný popis tabuľky
- 5. Pole obsahujúce štruktúru stĺpcov tabuľky
- 6. Inicializuje sa stĺpec Id a následne jeho ďalšie parametre
- 7. Popis stĺpca Id
- 8. Typ stĺpca Id je celé číslo s automatickým číslovaním
- 9. Hodnota ukladajúca sa do stĺpca Id nemôže byť prázdna
- 10. Inicializuje sa stĺpec Premenna a následne jeho ďalšie parametre
- 11. Stĺpec Premenna je typu reťazec
- 12. Maximálna dĺžka hodnoty vkladajúca sa do stĺpca Premenna je 255
- 13. Inicializuje sa stĺpec Hodnota a následne jeho ďalšie parametre

- 14. Nastaví sa stĺpec Id ako primárny prvok
- 15. Zvyšný kód pre definície ostatných databázových tabuliek
- 16. Vrátenie pola so štruktúrami tabuliek

Schematické znázornenie štruktúry tabuliek modulu vidno na obr. 6. Šípkami sú zobrazené relácie medzi jednotlivými tabuľkami. Legenda popisu stĺpcov:

[#] – celé číslo, Primárny, automatické číslovanie, [**I**] – celé číslo, [**T**] – Text, kódovanie utf8_general_ci, [**B**] – *Boolean*, [**V**] – reťazec, kódovanie utf8_general_ci. V prípade, že sú za názvom stĺpcu zátvorky s hodnotnou ide o preddefinovanú hodnotu, null – stĺpec môže byť prázdny. Na maximálnu dĺžku hodnoty v stĺpcoch sa neberie ohľad.



Obr. 6 Entitno – relačný diagram (ERD)

Ďalej požadujeme vytvoriť zložku **svk**, ktorú nám zabezpečí kód vo vnútri funkcie **svk_install**(). V prípade neúspešného vytvorenia je potreba ju vytvoriť manuálne. Problémom bývajú práva na zapisovanie.

[8][9][10]

Novo vytvorené databázové tabuľky **svk_nastavenia**, **svk_tituly** a **svk_skoly** naplníme začiatočnými hodnotami. V tabuľke **svk_nastavenia** sú názvom unikátne premenné a ich hodnoty pre nastavenia modulu, **svk_tituly** sú slovenské akademické tituly a **svk_skoly** sú štátne a verejné vysoké školy v Slovenskej a Českej republike, vrátane ich fakúlt.

Príklad zdrojového kódu naplnenia tabuľky **svk_nastavenia** dvomi riadkami premenných a vytvorenie zložky **svk**:

```
1. function svk_install() {
2.
     $hodnoty = array();
     $hodnoty[] = array(
з.
                        'Premenna' => "prezentacia dlzka",
                        'Hodnota' => "15");
4.
     $hodnoty[] = array(
5.
                        'Premenna' => "ubytovanie_datum",
                        'Hodnota' => "Definovat datum");
6.
     // Zvyšné dáta tabuľky svk nastavenia na zápis
7.
     foreach($hodnoty as $hodnota){
8.
       drupal_write_record('svk_nastavenia', $hodnota);
9.
10.
    // Kód pre naplnenie tabuliek svk_skoly,svk_tituly dátami
11. $Zlozka = "public://svk";
12. if(!is_dir($Zlozka)){
13.
    if(!file_prepare_directory($Zlozka,
                                   FILE_CREATE_DIRECTORY)) {
14.
          drupal_set_message("Varovná správa.....","error");
       }
     }
    }
```

- 1. Háčik, ktorý sa zavolá iba pri inštalácii modulu.
- 2. Inicializácia premennej **\$hodnoty** typu pole, kde sa budú ukladať ďalšie polia, ktorých indexy sú názvy stĺpcov danej tabuľky a hodnoty poľa ich hodnoty.
- 3. Stĺpec Premenna má hodnotu prezentacia_dlzka
- 4. Stĺpec Hodnota má hodnotu 15
- 5. Stĺpec Premenna má hodnotu ubytovanie_datum
- 6. Stĺpec Hodnota má hodnotu Definovat datum
- 7. Zvyšný kód pre definovanie polí s hodnotami pre tabuľku svk_nastavenia

- Cyklus, ktorý prechádza pole **\$hodnoty** tak, že premenná **\$hodnota** obsahuje pole s indexom názvu stĺpca tabuľky **svk_nastavenia** s jeho hodnotou na zápis
- 9. Drupal funkcia, ktorá zapíše do tabuľky svk_nastavenia aktuálny záznam poľa \$hodnota
- 10. Zdrojový kód, ktorý obsahuje dáta pre naplnenie ďalších tabuliek
- 11. Premenná \$Zlozka obsahuje cestu ku zložke svk
- 12. Zistenie či zložka svk neexistuje
- 13. Vytvorí sa zložka svk
- 14. Ak sa zložka svk nevytvorila, Drupal zobrazí správu

Veci potrebné pri inštalácii modulu sme vyriešili. Je však i dobrým gestom zahladiť za sebou stopy, ak chceme modul odinštalovať. Preto je treba vymazať obsah zložky **svk**, ako aj zložku samotnú. Kód na elimináciu súborov a zložky sa nachádza v háčiku **svk_uninstall**(). V prípade neúspešného odstránenia je nutné zložku odstrániť manuálne. Za príčinu chyby sa znovu považujú práva na zápis.

Drupal automaticky odstráni naše databázové tabuľky definované v háčiku **svk_schema**, preto nie je potrebné zaoberať sa nad programovým riešením.

Ukážka zdrojového kódu na odstránenie zložky svk:

```
1. function svk_uninstall(){
2. $Zlozka = "public://svk";
3. if(file_unmanaged_delete_recursive($Zlozka)){
4. if(!drupal_rmdir($Zlozka)){
5. drupal_set_message(" Varovná správa....","error");
    }
6. }else{drupal_set_message(" Varovná správa....","error");
}
```

- 1. Háčik sa zavolá iba pri odinštalácií modulu
- 2. Premenná \$Zlozka obsahuje cestu ku zložke svk
- 3. Obsah zložky svk sa vymaže rekurzívne
- 4. Vymaže sa zložka svk
- 5. Ak sa zložka svk neodstránila, Drupal zobrazí varovanie
- 6. Ak sa obsah zložky svk nevymazal, Drupal zobrazí varovanie

3.1.2 Funkcie jadra modulu

Podstatou tejto kapitoly bude ozrejmenie hlavnej práce modulu aj jeho programové riešenie s vysvetlením častí kódu.

Hlavnou funkciou modulu je generovanie určitého obsahu pre konkrétne stránky. Logika programového spracovania sa nachádza v súbore **svk.module**. Základom je využitie blokov, ktoré sú priradené do regiónu. Pod pojmom región chápeme vyhradené miesto na stránke definované v šablóne témy, v ktorom sa podľa nastavenia v administračnom rozhraní generuje určitý obsah. Obsahom môže byť myslené bočné menu, obsah stránok, prihlasovací formulár atď. Dostupné regióny si môžeme zobraziť v sekcii *Structure – Blocks - Demonstrate block regions*. Na obr. 7 sú znázornené všetky využiteľné regióny.

Banner 1			
Top 1	Top 2	Top 3	
First vertical menu	Banner 2		
First sidebar	User 1	User 2	
	Banner 3		
	Home		
	verzia5		
	Content		
	Banner 4		
	User 3	User 4	
	Banner 5		
Bottom 1	Bottom 2	Bottom 3	
Banner 6			
		Footer	
		Copyright	

Obr. 7 Zoznam dostupných regiónov na stránke ŠVK

Keďže každý generovaný obsah našim modulom bude predstavovať časť hlavného obsahu stránky, bloky nastavíme neskôr tak, aby sa zobrazovali v regióne **Content**.

Súbor svk.module obsahuje 5 háčikov: svk_menu, svk_block_info, svk_block_view, svk_nastavenie_modulu_form, svk_nastavenie_modulu_form_submit, ktorých funkcionalita je rozpísaná nižšie.

Aby sme mohli používať vlastnú stránku na nastavenie ŠVK modulu, bude potreba najprv pridať odkaz do databázy. O to sa postará háčik **svk_menu**, ktorý sa zavolá iba keď sa vyčistí vyrovnávacia pamäť Drupalu. Na ukážku zdrojový kód, ktorý zaregistruje odkaz:

function svk_menu(){

- 1. Index premennej \$item predstavuje URL odkaz na nastavenie modulu
- 2. Jedná sa o klasický viditeľný odkaz
- 3. Názov odkazu
- 4. Popis linku pod názvom odkazu
- 5. Spätné volanie funkcie vracajúcej v tomto prípade vygenerovaný formulár
- 6. Parametre pre page callback funkciu háčik, ktorý obsahuje štruktúra formulára
- 7. Určuje aby sa odkaz zaradil na administračnú stránku
- 8. Vrátenie zoznamu štruktúry odkazov

[6][7]

Vytvorili sme si v administračnom rozhraní odkaz na stránku **Nastavenia modulu ŠVK**, ale chýba nám obsah samotný. Ten zabezpečíme cez háčik **svk_nastavenie_modulu_form**. Definuje potrebné formulárové políčka s ich pred vyplnenými hodnotami načítanými z databázovej tabuľky **svk_nastavenia**. Funkcia má dva vstupné parametre. Prvý je premenná **\$form** obsahujúca štruktúru formulárových prvkov a druhý je adresa premennej **\$form_state** obsahujúca spracované dáta celého formulára. Avšak nato, aby sa nám vygeneroval formulár do HTML podoby, pretože funkčnou návratovou hodnotou **\$form** je pole polí, použijeme drupalovskú funkciu **drupal_get_form**, ktorá bola použitá pri tvorbe odkazu nastavenia modulu. Viac podrobností k jednotlivým prvkom nastavenia modulu bude v ďalšej kapitole.

Pre názornú ukážku uvádzame skrátený zdrojový kód pre vytvorenie 2 formulárových prvkov:

```
function svk_nastavenie_modulu_form($form, &$form_state){
1. //Kód pre načítanie nastavení z databázy
```

2. \$form['prezentacia_dlzka']=array(

```
3. '#type'=>'select',
```

- 4. '#title'=>t('Dĺžka prezentácie'),
- 5. '#description'=>'Popis.....',
- 6. '#options'=>array(1=>1, ..., 59=>59),
- 7. '#default_value'=>@\$NastaveniaQ['prezentacia_dlzka'],
- 8. '#required'=> true,
- 9. //Kód pre definovanie ostatných prvkov formulára

```
10. $form['submit']=array('#type'=>'submit',
```

'#value'=>t('Uložit'));

11.return \$form; }

- Načítanie všetkých premenných a ich hodnôt z databázovej tabuľky svk_nastavenia pre budúce pred vyplnenie. Hodnoty sa ukladajú do pola \$NastaveniaQ indexované názvom prvkov resp. premenných z DB tabuľky
- 2. Definovanie štruktúry prvku s identifikátorom zhodujúcim sa s premennou v DB tabuľke
- 3. Typ elementu vo forme výberu selectboxu
- 4. Názov prvku Dĺžka prezentácie
- 5. Bližšia špecifikácia prvku
- 6. Pole hodnôt pre *selectbox* ako čas v minútach od 1 do 59
- 7. Získanie hodnoty pre daný prvok z premennej **\$NastaveniaQ** indexom premennej.
- 8. Povinný prvok, ktorý musí byť vyplnený vybraný
- 9. Zvyšný kód obsahujúci definovanie štruktúry ďalších prvkov
- 10. Definovanie prvku ako odosielacieho tlačidla
- 11. Vrátenie štruktúry všetkých prvkov

Doteraz sme si vytvorili v admine odkaz **Nastavenie modulu ŠVK**. Pri kliknutí na odkaz nám Drupal zavolá funkciu **drupal_get_form** vracajúcu vygenerovaný HTML formulár pretože jej parametrom je háčik **svk_nastavenia_modulu_form** vracajúci štruktúru formulára na nastavenie modulu.

Ostáva nám spracovať údaje formulára odoslaní. Háčik ро jeho svk nastavenia modulu form submit sa zavolá až po úspešnej validácii odoslaného formulára. Pričom štruktúra názvu funkcie musí byť vo formáte NázovModulu NazovFormulara form submit. Ako však nastáva validácia? Je definovaná programátorom s využitím háčika validate, alebo ak sa nám nachádza čo i len jeden prvok s atribútom **required(true)**, Drupal vopred skontroluje či je daný prvok vyplnený, alebo kombinácia oboch prípadov. My si však vystačíme s bežnou kontrolou nenulového vyplnenia položiek. Teda všetkým prvkom formulára nastavíme **required(true)**.

Na ukážku zdrojový kód spracovania dát odoslaných formulárom:

```
function svk_nastavenie_modulu_form_submit($form, &$form_state){
1. $Hodnota=$form state['values'];
2. $ZoznamPrvkov=array("prezentacia_dlzka",
                        "ubytovanie_datum",
                       "diplom_datum",
                       "admin kontakt",
                        "registracia_email",
                        "ubytovanie email");
3. foreach($ZoznamPrvkov as $prvok) {
     db_update("svk_nastavenia")
4.
             ->fields(array(
               'Hodnota' => $Hodnota[$prvok],))
5.
             ->condition('Premenna', $prvok)
6.
             ->execute();}
7. drupal set message("Nastavenia boli uložené", "status");}
```

- \$form_state s indexom values obsahuje pole ošetrených hodnôt formulára, preto aby sme ju
 v ďalších krokoch nemuseli pracne vypisovať, definujeme si ju ako \$Hodnota, kde keď
 pridáme index s názvom formulárového prvku získame jeho hodnotu.
- 2. Pole identifikátorov všetkých formulárových prvkov
- 3. Cyklus, ktorý prechádza všetky prvky pola \$ZoznamPrvkov
- 4. Databázová tabuľka svk_nastavenia sa bude aktualizovať
- 5. Prepíše sa stará hodnota novou alebo starou hodnotou daného prvku
- 6. V tabuľke, kde sa nachádza premenná s názvom práve zisteného prvku poľa
 \$ZoznamPrvkov, sa zapíše daná hodnota
- 7. Oznámime správcovi, že údaje boli uložené

Pozrime sa však na srdce modulu. Podľa požiadaviek sa vyžaduje od modulu, aby obsluhoval následné stránky: *Registrácia príspevkov*, *Ubytovanie*, *Zoznam ubytovaných*, *Zoznam príspevkov* v sekcii, *Zoznam príspevkov*, *Výsledky*, *Diplomy a potvrdenia o účasti*, *Zaradenie príspevkov do sekcie*, Úprava sekcií, *Program*, Úprava programu.

Každá stránka ponúka návštevníkovi základné informácie a popis, preto bude potreba spolupráce blokov a stránok. Stránka bude zobrazovať informačný obsah a bloky budú generovať podľa typu stránky svoj obsah. Nastavenie prepojenia spolupráce medzi jednotlivými súčasťami RS bude ozrejmené v ďalšej kapitole.

Keďže podstata generovania obsahu je v blokoch, použijeme na to háčik **svk_block_info**, ktorý oznámi drupalu, aby zaregistroval do systému určité bloky s preddefinovanými hodnotami. Na ukážku skrátený kód vytvorenia jedného bloku.

```
function svk_block_info() {
1.
    $blocks = array();
2.
    $blocks['registracia'] = array(
      'info' => t('Blok SVK - Registrácia príspevkov'),
3.
4.
      'cache' => DRUPAL NO CACHE,
     'status' => TRUE,
5.
6.
      'region' => 'content',
      'visibility' => BLOCK_VISIBILITY_LISTED,
7.
8.
      'pages'=>'registracia-prispevkov');
    //Kód pre štruktúru zvyšných blokov
9.
10. return Sblocks;
}
```

- 1. Inicializácia premennej typu pole, ktorý bude obsahovať štruktúru blokov
- 2. Definícia bloku s jedinečným identifikátorom registracia
- 3. Názov bloku v administračnom rozhraní
- 4. Aby sa obsah neukladal do vyrovnávacej pamäte a vždy sa generoval nový
- 5. Automaticky sa aktivuje
- 6. Obsah bloku sa bude generovať v regióne Content kde sa generuje aj obsah stránok
- 7. Blok sa zobrazí len pre určitú stránku, ktorej URL spresníme nižšie
- 8. Predpokladané URL stránky kde sa bude obsah generovať. Každý blok musí mať uvedenú URL stránky, inak by sa mohol generovať obsah aj na iných stránkach
- 9. Zvyšný kód pre ďalšie štruktúry blokov
- 10. Vrátenie štruktúry blokov

Posledným krokom je určenie správnosti generovania obsahu. Využijeme háčik svk_block_view. Parametrom funkcie je premenná **\$delta**, ktorá obsahuje unikátny názov

identifikátoru daného bloku ŠVK modulu definovanom v háčiku **_block_info**. Potrebujeme teda priradiť k danému bloku správny obsah. Riešime to pomocou *switchu*.

Ukážka skráteného kódu generovania obsahu formulára **Registrácia príspevkov** pre blok s identifikátorom **registracia**:

```
function svk_block_view($delta = ''){
1. $Blok_Obsah=&$block['content'];
    switch($delta){
2. case'registracia':
3. module_load_include('inc','svk',
                         'svk.registracia.prispevkov');
4. $Blok_Obsah = drupal_get_form(
                         "svk_registracia_prispevkov_form");
        break;
5. // Zvyšný kód pre určenie daného obsahu bloku
    }
6. return $block;
}
```

- Uloženie adresy premennej **\$block** s indexom **content** určený na obsah do premennej
 \$Blok_obsah pre l'ahšiu manipuláciu
- 2. Identifikátor názvu bloku sa zhoduje s identifikátorom spracovávaného bloku
- **3.** Vloží pomocný súbor kde je kód pre generovanie obsahu daného bloku. Prvý parameter je koncovka súboru, druhý jedinečný názov modulu a tretí názov súboru bez koncovky
- **4.** Uloží do premennej vygenerovaný obsah registračného formulára, ktorého háčik je definovaný v súbore nášho modulu **svk.registracia.prispevkov.inc**
- 5. Definícia zvyšného kódu pre iné bloky
- 6. Vráti vygenerovaný obsah

Zoznam stránok, ktoré modul obsluhuje, ich jedinečné URL – ALIAS, ktoré sa pre uľahčenie štandardne nastavujú blokom pri inštalácii modulu, jedinečný identifikátor daného bloku v zdrojovom kóde a zdrojový súbor so skriptami vidno v tabuľke Tab. 1. Názvy URL a stránok možno meniť.

Názov stránky	URL stránky ALIAS	Identifikátor bloku	Skriptovací súbor svk. názov .inc
Registrácia príspevkov	registracia-prispevkov	registracia	registracia.prispevkov
Ubytovanie	ubytovanie	ubytovanie	ubytovanie
Zoznam ubytovaných	zoznam-ubytovanych	ubytovany	zoznam.ubytovanych
Zoznam príspevkov v sekcii	zoznam-prispevkov-v -sekcii	prispevky_v_sekcii	zoznam.prispevkov.v.sekcii
Zoznam príspevkov	zoznam-prispevkov	prispevky	zoznam.prispevkov
Výsledky	vysledky	vysledky	vysledky
Diplomy a potvrdenia o účasti	diplomy-a-potvrdenie-o -ucasti	diplomy	diplomy.a.potvrdenia.o. ucasti
Zaradenie príspevkov do sekcie	zaradenie-prispevkov-d o-sekcie	sekcie_zaradenie	zaradenie.do.sekcie
Úprava sekcií	uprava-sekcii	sekcie	sekcie
Program	program	program	program
Úprava programu	uprava-programu	program_uprava	uprava.programu

Tab. 1 Stránky a ich parametre

Uvedieme príklad. Máme stránku *Ubytovanie*, ktorej URL sme vyplnili ako *ubytovanie*. Pri jej navštívení sa zobrazí obsah stránky a modulom vygenerovaný obsah pre túto stránku. Skriptu vyhovel iba blok s jedinečným názvom *ubytovanie*, preto sa vložil súbor so skriptami **svk.ubytovanie**.na vygenerovanie obsahu.

Registrácia príspevkov – stránka ponúka základné informácie a modul generuje formulár pre registráciu príspevku do súťaže ŠVK. Užívateľ musí vyplniť všetky polia, pretože sú povinné. Oproti pôvodnej verzii sa vykonalo viacero zmien. Pôvodný formulár bol vytvorený cez modul *Webform*. Keďže sa vyžadovalo lepšie spracovanie dát a podrobnejšie rozdelenie prvkov, modul *Webform* bol vynechaný z používania a nahradený našim zdrojovým kódom na generovanie a obsluhu formulára.

Pre daný príspevok sa môže registrovať aj viac ako len jeden autor. Kliknutím na tlačidlo *Pridať kolónky nového autora* sa formulár síce odošle, ale kód len pridá kolónky.

V pôvodnom riešení sa tituly museli vypisovať ručne a nie vždy korektne spolu s menom a priezviskom, pričom každý autor sa musel oddeľovať špecifickým znakom. Tak isto aj univerzita a fakulta.

Výhoda súčasného riešenia spočíva v separátnom oddelení a zjednodušení zberu dát, ako pre užívateľa tak aj pre správcu. Tituly sú rozdelené podľa správnej formy písania na *Titul za menom* a *Titul pred menom*. Ak má autor viac titulov dajú sa označiť viaceré alebo ak sa tituly nenachádzajú v zozname, môže si do správnej kolónky vyplniť tituly. Nové tituly sa zapíšu do tabuľky.

V prípade, že autor je študent, skript ho automaticky nastaví ako prezentujúceho. V opačnom prípade, ak je autor učiteľ, nepovažuje sa za prezentujúceho.

V prípade výberu univerzity a fakulty ma užívateľ na výber 3 možnosti. Vyberie si zo zoznamu fakultu, ktorá patrí danej univerzite, alebo ak nie je fakulta v zozname existujúcej univerzity, stačí ak klikne na tlačidlo *Pridať fakultu* a vypíše názov alebo ak sa nenachádza univerzita v zozname klikne na *Pridať univerzitu a fakultu*. Vtedy vypíše názov fakulty a univerzity. Nové školy sa zapíšu do tabuľky.

Vzorový registračný formulár je na obr. 8.

lázov práce *			
ekcia príspevku *			
- Select -	•		
litul pred menom	Meno autora *	Zaradenie autora	Titul za menom
Bc.		Študent	CSc.
Dr.h.c. Ing.	Priezvisko autora *	Učiteľ	PhD. ThDr.
Dopinit' titul(y)			Dopinit' titul(y)
Pridať kolónky nového	autora	odderovat medzerou	
Jniverzita a Fakulta *			
- Select -	-		
Pridať novú univerzitu	a fakultu Pridať iba novú fakultu		
Abstrakt *			

Prehľadávať... Nie je zvolený súbor.

Zaregistrovať príspevok

Obr. 8 Formulár na registráciu príspevku

Ubytovanie – stránka zobrazí informácie a blok s formulárom pre rezervovanie miesta na ubytovanie. Jedinou hlavnou zmenou oproti starej verzii je vynechanie modulu *Webform* a naprogramovanie nášho formulára.

Zoznam ubytovaných – stránka zobrazí členovi *OV* zoznam zaregistrovaných rezervácií na ubytovanie s podrobnosťami.

Zoznam príspevkov v sekcii – podrobný prehľad o registrovaných príspevkoch určený pre *OV*. **Zoznam príspevkov** – zjednodušený zoznam príspevkov pre bežného užívateľa, ktorých príspevky sú už zaradené v konečnej sekcii.

Výsledky – stránka s výsledkami súťaže jednotlivých súťažiacich podľa kategórií, ktorá je prístupná aj pre bežného užívateľa.

Diplomy a potvrdenia o účasti – stránka na uloženie umiestnenia v súťaži a generovanie PDF diplomu alebo potvrdenia o účasti pre danú sekciu. Prístupné len pre *Editora*.

Zaradenie príspevkov do sekcie – presúvanie zaregistrovaných príspevkov do konečných sekcií. Práva má len *Editor*.

Úprava sekcií – vytváranie a editovanie konečných sekcií kde sa budú zaraďovať zaregistrované príspevky. Práva má len *Editor*.

Program – stránka prístupná pre každého užívateľa zobrazí podrobné informácie o čase prezentácie a poradia.

Úprava programu – podľa sekcií sa určuje poradie a čas na prezentáciu, miestnosť a komisia. Práva ma len *Editor*.

3.1.3 Inštalácia a konfigurácia modulu

Prihlásime sa do systému pod účtom administrátora. V hornej lište klikneme na **Modules** a v zozname modulov hľadáme názov modulu **ŠVK** zaradený do kategórie **Študentská vedecká konferencia.** Zaškrtneme políčko vedľa názvu a uložíme zmeny. Modul sa v priebehu pár sekúnd nainštaluje a je pripravený na použitie.

V administračnom rozhraní klikneme na *Configuration* > *Nastavenie modulu ŠVK* a nastavíme jednotlivé políčka. **Dĺžka prezentácie** predstavuje dĺžku prezentácie v minútach. Hodnota tejto premennej sa používa na výpočet konečného času prezentácie na stránke *Program*. **Dátum ubytovania** predstavuje termíny ubytovania, kde každý termín sa oddeľuje bodkočiarkou. Výsledné termíny sa zobrazia ako políčka na zaškrtnutie na stránke *Ubytovanie*. **Dátum na dokumente** sa používa pri generovaní na stránke *Diplomy a potvrdenia o účasti*. **Emailová adresa odosielateľa** je inými slovami "Od koho", keď zasiela systém e-maily po úspešnom odoslaní formulára zo stránok *Ubytovanie* a *Registrácia príspevkov*. **Emailové adresy adminov** – **Ubytovanie/Registrácia príspevkov** nie je nič iné, ako zoznam emailových adries, kde bude posielaná informačná pošta o novej registrácii príspevku alebo novom ubytovaní.

Nie každá stránka však bude prístupná pre každého užívateľa. Či už prihláseného alebo neprihláseného. Zabezpečíme to cez pomocný externý modul **node privacy byrole**, ktorý si stiahneme z *https://www.drupal.org/project/node_privacy_byrole* a aktivujeme. Súbežne si vytvoríme 4 typy účtov. Pomocou menu *Admin > People > Permissions > Roles* si vytvoríme *OV, Uzivatel, Editor* a *Administrator*. Do týchto rolí zaradíme členov majúcich účet v RS. Prípadne im nastavíme globálne práva čo môžu a nemôžu robiť. Podľa požiadaviek by napr. *Editor* mal mať práva na zobrazenie a úpravu stránky *Úprava sekcií, Zaradenie do sekcie* a *Úprava programu*. Toto však nenastavíme cez *Permissions* ale cez vyššie spomínaný modul

node privacy by role. Pri vytváraní novej stránky nám pod textovým poľom obsahu ponúkne, ktorým rolám by mala byť stránka prístupná. Ak vytvárame napríklad *Úprava sekcií*, zaškrtneme voľby *View* a *Edit* pre rolu *Editor* a zvyšné ostanú nezaškrtnuté.

V sekcii *Content* > *Pages* si vytvoríme jednotlivé stránky, ktoré modul obsluhuje a sú spomenuté v kapitole vyššie. Nastavíme práva pre jednotlivé role. Teraz je však najdôležitejšie prepojiť blok a stránku. Vytvárame si napríklad stránku **Zaradenie do sekcie**. Pridáme nejaký obsah stránky, ktorý sa bude informačného charakteru. Povolíme aby sa zobrazovala ako odkaz v hlavnom menu. Klikneme na **URL path settings** a sem zadáme takzvaný alias. Alias musí byť jedinečný. Pre jednoduchosť a analógiu si ho pomenujeme ako **zaradenie-do-sekcie**. Uložíme zmeny. Ak by sme zadali do webového prehliadača **localhost/svk/zaradenie-do-sekcie** zobrazí nám našu práve uloženú stránku. Ostáva už len priradiť vhodný blok ku tejto stránke.

Presunieme sa do *Structure > Blocks*. Vidíme, že už máme bloky začínajúce s názvom **Blok SVK** zaradené do regiónu *Content*. Nájdeme si blok **Blok SVK – Zaradenie do sekcie** a klikneme na *configure*. Podľa všetkého máme pre voľbu *Pages* zaškrtnutú voľbu *Only the listed pages* a predvyplnenú hodnotu. A práve táto hodnota je náš alias vytvorenej stránky. V prípade, že sa nezhoduje, prepíšeme na aktuálnu URL adresu – Alias, teda v tejto situácii zaradenie-do-sekcie. V *Content types* zaškrtneme *Pages* a v *Roles* iba tie skupiny, ktoré majú práva na zobrazovanie ako parameter *View* pri vytváraní danej stránky. Čiže *Editor*. Uložíme zmeny a naša stránka bude zobrazovať aj okrem svojho obsahu, i potrebný obsah generovaný modulom.

4 Diskusia

Najzdĺhavejším a najväčším problémom pri tvorbe modulu bolo pochopenie a naučenie sa akým spôsobom Drupal generuje rôzne systémové súčasti. Pretože má svoj vlastný spôsob generovania rôznych prvkov (napr. formulár). Ďalší strávený čas bol venovaný učeniu ako sa Drupal-ovsky vytvára daná vec. Mohli by sme vytvoriť formulár klasickým HTML spôsobom, ale sme tu od toho, aby sme využili silu RS. Je to len plus, pretože vieme niečo nové.

Dotýkali sme sa s problémom, keď zdrojový kód podľa vzorových príkladov vybraného háčika napríklad _menu nevygeneroval požadovaný odkaz. Bolo potrebné obísť systém vyrovnávacej pamäti (cache), ktorý zapríčiňuje, že sa niektoré časti kódu (háčiky) vykonajú až po vymazaní či aktualizovaní vyrovnávacej pamäte RS. Poslúžil nato modul *Devel* i keď nie vždy spolupracoval ako mal. Našťastie pri vývoji modulu ŠVK sme nevytvárali kód, ktorý by do značnej miery potreboval aktualizovať vyrovnávaciu pamäť. Stačilo len zavolať súbor **update.php**, ktorý je v **root-e** RS alebo vymazať vyrovnávaciu pamäť Drupalu. Síce to trvalo dlhšie ako generovanie stránok, no vývoj modulu radikálne nespomaľoval.

Ďalším problémom bolo volanie háčikov, ich závislosť na sebe a prepojenosť. Jemným príkladom môže byť, že háčik **_submit** sa zavolá až po úspešnej validácii háčikom **_validate**. Teda háčik **_submit** sa nevykonal vždy, čo v očiach začiatočníka vyvolalo otázku: "*Prečo*?".

Referenčná príručka Drupalu v tomto smere nie je až taká obsiahla, preto bolo potreba sa inšpirovať, poučiť, skúšať nové veci a kódy z iných stránok.

Pri robení praktickej časti záverečnej práce sa neustále vyskytli možnosti ako spraviť niektoré veci jednoduchšie. V rámci možností a času zmeny boli aplikované. Po funkčnej stránke by mal vyhovovať požiadavkám, odstrániť nedostatky a čo možno v najväčšej miere uľahčiť jeho integrovanie a obsluhu.

5 Záver

Cieľom celej práce bolo oboznámenie sa s tvorbou vlastného modulu pre redakčný systém Drupal, pričom sme využili moduly jadra i externé moduly. Potrebovali sme si nainštalovať lokálny webserver s podporou skriptovacieho jazyka PHP a databázy MySQL. Boli však vyžadované znalosti PHP a práce s databázou.

Na začiatok sme museli analyzovať súčasný stav stránok ŠVK a navrhnúť vhodné riešenie pre zjednodušenie ovládania, implementácie a optimalizácie.

Funkcia celého modulu spočívala vo vytvorení rozdielnych blokov, v ktorých sa generoval presne stanovený obsah. Následne sme dané bloky priradili do presne stanovených stránok vytvorených prostredníctvom redakčného systému.

Zoznam použitej literatúry

- [1] Open Source redakčné systémy čo sú zač, prečo sú zadarmo a ako ich využiť, Dostupné na internete: <u>http://www.zajtra.sk/programovanie/231/open-source-redakcne-systemy-co-su-zac-preco</u> <u>-su-zadarmo-a-ako-ich-vyuzit</u>, On-line, 20.5.2015
- [2] Writing module .info files (Drupal 7.x), Dostupné na internete: <u>https://www.drupal.org/node/542202</u>, On-line, 20.5.2015
- [3] *Writing .install files (Drupal 7.x)*, Dostupné na internete: https://www.drupal.org/node/876250, On-line, 20.5.2015
- [4] How to write your own module in Drupal 7, Dostupné na internete: <u>http://www.justinphelan.com/post/how-write-your-own-module-drupal-7</u>, On-line, 20.5.2015
- [5] Drupal 7 How to Create a Module Part 1, Dostupné na: <u>http://www.wdtutorials.com/drupal-7/drupal-7-how-to-create-module-part-1#.VV9pvkZ</u> <u>LqSr</u>, On-line, 20.5.2015
- [6] 7. Basic form with submit handler, Dostupné na internete: https://www.drupal.org/node/717740, On-line, 20.5.2015
- [7] Form API reference, Dostupné na internete: <u>https://api.drupal.org/api/drupal/developer!topics!forms api reference.html/7</u>, On-line, 20.5.2015
- [8] Správne používanie akademických a vedecko-pedagogických titulov na Slovensku, Dostupné na internete: <u>http://www.cardiology.sk/casopis/407/pdf/13.pdf</u>, On-line, 20.5.2015
- [9] Metodické usmernenie k používaniu titulov, čestných titulov a označení funkcií vysokoškolských učiteľov, Dostupné na internete: http://www.fhi.sk/files/predpisy/Metodicke_usmernenie_k_pouzivaniu_titulov.doc, On-line, 20.5.2015
- [10] Vysoké školy na Slovensku a v Čechách,
 Dostupné na internete: <u>http://www.univerzita.net</u>, On-line, 20.5.2015

Prílohy

Príloha A: CD médium – práca v elektronickej podobe, zdrojové súbory modulu, SQL štruktúra databázových tabuliek s preddefinovanými hodnotami