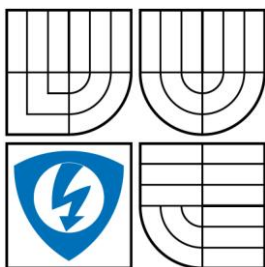


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ



FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

WEBOVÁ APLIKACE PRO VÝUKU SIMULACÍ V NS2

WEB APPLICATION FOR NS2 TRAINING

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

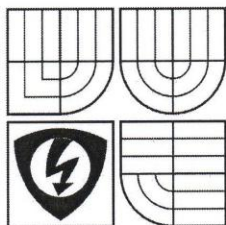
AUTOR PRÁCE
AUTHOR

BC. VÁCLAV PAVLOSEK

VEDOUCÍ PRÁCE
SUPERVISOR

ING. MILAN ŠIMEK

BRNO 2009



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ
Fakulta elektrotechniky
a komunikačních technologií
Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Pavlosek Václav, Bc.
Ročník: 2

ID: 47605
Akademický rok: 2008/09

NÁZEV TÉMATU:

Webová aplikace pro výuku simulací v ns2

POKYNY PRO VYPRACOVÁNÍ:

V programovacím jazyku PHP navrhnete webovou aplikaci pro výuku simulací v nástroji Network Simulator 2. Aplikace musí obsahovat databázi projektů včetně jejich efektivního vyhledávání. Pro studenty také připravte diskuzní fórum.

DOPORUČENÁ LITERATURA:

- [1] Flynt C.: Tcl/Tk - A Developer's Guide, Second Edition, Morgan Kaufmann Publishers, ISBN: 1-55860-802-8
- [2] J. Castagnetto, H. Rawat, S. Schuman, Ch. Scollo, D. Veliath, Professional PHP Programming, Wrox Press Ltd. 1999.

Termín zadání: 9.2.2009

Termín odevzdání: 26.5.2009

Vedoucí práce: Ing. Milan Šimek
Konzultanti diplomové práce:


prof. Ing. Kamil Vrba, CSc.
předseda oborové rady



UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Anotace

V této závěrečné zprávě jsou uvedeny informace k méj diplomové práci „Webová aplikace pro výuku simulace v NS2“. Tato aplikace je po nainstalování funkční a její zdrojové kódy jsou uloženy na přiloženém CD.

Je pojednáno o nástroji Network Simulator 2. Pomocí něj se realizují simulace sítí a o nich pak jejich autor vkládá údaje do webové aplikace. Zaregistrovaný návštěvník webu má možnost vložit do aplikace projekt. Projekt obsahuje informaci o jím vytvořené simulaci v NS2. Webová aplikace umožňuje také zobrazit si detail libovolného vloženého projektu, který je schválen administrátorem. Dále může řadit projekty, vyhledávat v nich zadaný výraz, či připojovat své příspěvky v diskusním fóru. Správce může ve svém okně projekty odevzdané uživatelem schvalovat, tím je zpřístupnit ostatním, a také je odstraňovat z databáze.

Teorie o technologiích, které jsou využity při implementaci této aplikace. Jedná se o webový server Apache, databázový server MySQL, a jazyk PHP. Dále jsou zmíněny informace o zabezpečení webových aplikací včetně příkladů možných útoků na aplikace a jejich databáze. Je uveden návrh databáze, která tvoří jádro aplikace. Tento návrh je závislý na požadavcích na aplikaci. Další kapitoly dávají čtenáři ucelený obraz o funkčnosti aplikace, jsou uvedeny ukázky grafické podoby aplikace tak, jak je ve finálním stádiu implementována, a také ukázky zdrojových kódů pro vytváření databázových tabulek aplikace.

Anotace - anglicky

There is information to my master's thesis which is called "Web application for NS2 training". This application works after installation and its source codes are saved on applied CD.

It is said about implement Network Simulator 2. It helps to realize simulation of nets and then author inserts information about them into web application. Registered web's visitor has possibility to insert project into application. The project contents information about simulation created in NS2. Web application can also visible detail of possible project which is approved of administrator. Then the visitor can sort projects, search

entered expression or connect his contribution to discussion forum. Administrator can approve user's projects in his part of application. It makes available for the others. He can also delete them from database.

Theory about technologies which are used for implementation of this application. It is talked about web Apache server, database MySQL server and programmable PHP language. There is also mentioned information about security of web application included possible attacks on applications and their database. It is presented proposal of database which creates core of application. This proposal is depended on application requirements. Next chapters give to reader whole image about functionality of application. There are mentioned samples of final graphical image of application. This document also provides the shows of source codes for creating database tables.

Klíčová slova

NS2, simulace počítačových sítí, Apache, MySQL, PHP, HTML, webová aplikace, zabezpečení webových aplikací, XSS, SQL injection, ER diagram, schéma databáze, projekt.

Klíčová slova – anglicky

NS2, simulation of computer nets, Apache, MySQL, PHP, HTML, web application, security of web applications, XSS, SQL injection, ER diagram, database scheme, project.

Bibliografická citace práce:

PAVLOSEK, V. *Webová aplikace pro výuku simulací v ns2*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 46 s. Vedoucí diplomové práce Ing. Milan Šimek.

Prohlášení

Prohlašuji, že svou diplomovou práci na téma **Webová aplikace pro výuku simulací v NS2** jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 26.5.2009

.....
podpis autora

Poděkování

Děkuji vedoucímu diplomové práce Ing. Milanu Šimkovi, odbornému asistentovi ústavu telekomunikací, za odbornou pomoc při realizaci práce.

V Brně dne 26.5.2009

.....
podpis autora

OBSAH

1	Úvod.....	6
2	Základní popis simulačního nástroje Network Simulator.....	7
2.1	Příklady příkazů v NS2	8
2.2	Zobrazení výsledků	8
3	Použité technologie při realizaci aplikace	10
3.1	Apache server.....	10
3.2	PHP	10
3.2.1.	Schopnosti, možnosti a vlastnosti	11
3.3	MySQL.....	15
4	Zabezpečení webových aplikací	16
4.1	Cíle útoků	16
4.2	XSS (Cross Site Scripting).....	16
4.2.1.	Zabezpečení proti XSS	17
4.3	Zabezpečení databáze.....	17
4.3.1.	SQL injection.....	17
4.3.2.	Metody zabezpečení	18
4.4	Zabezpečení citlivých údajů v databázi	18
4.4.1.	Jednocestné zabezpečení.....	18
4.5	Nastavení serverového prostředí používající jazyk PHP	19
5	Návrh modelu webové aplikace.....	21
5.1	Požadavky na výsledné chování.....	21
5.1.1.	Požadavky na výsledné chování aplikace	21
5.1.2.	Další požadavky.....	22
5.1.3.	Požadavky na databázi.....	22
5.1.4.	Požadavky na implementační jazyk a použité technologie	22
5.2	Návrh databáze webové aplikace	23
5.2.1.	ER diagram	23
5.2.2.	Schéma databáze.....	24
6	Funkčnost aplikace	25
6.1	Programovací jazyk.....	25
6.2	Volba internetového prohlížeče	25
6.3	Struktura html části PHP souboru	25
6.4	Význam nejdůležitějších částí aplikace	25
6.4.1.	Hlavní stránka	25
6.4.2.	Vložení nového projektu.....	26
6.4.3.	Projekty.....	26
6.4.4.	Detail projektu	26
6.4.5.	Správa systému – Administrátor.....	27
6.4.6.	Diskusní fórum	28
7	Ukázky aplikace.....	29
8	Ukázky zdrojových kódů	38
8.1	Vytvoření databázových tabulek v programu MySQL	38
	Závěr	43
	Seznam použité literatury	44

1 Úvod

Cílem této diplomové práce je navrhnout model a implementovat webovou aplikaci pro výuku simulací v NS2, která je určena pro uživatele Internetu zabývající se problematikou využití simulačního nástroje Network Simulator 2. Řešením bude vytvoření takové aplikace, pomocí které uživatelé mohou vkládat své projekty do její databáze. Umožní také získávat informace o jiných projektech vytvářených dalšími účastníky webu či připojit své příspěvky v relaci diskusní fórum. K aplikaci se může připojit jeden či více administrátorů a provést správu projektů, jako je schvalování, úprava či mazání.

Vlastnostem a využití populárního simulačního nástroje Network Simulator 2 pro modelování chování sítě se blíže věnuje následující kapitola. Další kapitoly poskytují informace k základní teorii o návrhu a implementačních nástrojích pro tvorbu obecných webových aplikací a jejich zabezpečeníh proti možným útokům na citlivé informace v databázi aplikace. Zaměření dalších kapitol je na požadavky, návrh konkrétní aplikace a funkčnost již nainstalované aplikace. V předposlední části tohoto dokumentu je možno shlédnout ukázky webové aplikace, a poslední část je věnována ukázkám kódů aplikace.

2 Základní popis simulačního nástroje Network Simulator

Nástroje, které umožňují simulovat datové sítě, hrají v dnešním světě čím dál tím významnější roli. Důvodem jejich využívání je zejména menší finanční zátěž na jejich uživatele, kteří navrhují sítě než při fyzické realizaci testovacích sítí. Jedním z takových možných nástrojů, který se pro simulaci chování sítí používá je Network Simulator 2. Dokáže simulovat sítě řízené diskretními událostmi. Byl vyvinut na univerzitě v Berkeley v objektově orientovaném jazyku C++. Lze v něm výkonně simulovat chování lokálních i rozsáhlých datových sítí.

Mezi základní operace NS2 patří:

- budování modelu sítě
- stanovení cesty v modelu sítě
- budování spojení (na úrovni vyšších vrstev OSI modelu)
- generování provozu
- modelování poruch
- monitorování provozu

Model se vytváří psaním do textového souboru. Strukturu definuje jazyk TCL. Cesta se v modelu rozlišuje dvěma typy.

Cesta v modelu rozlišuje přenosy typu unicast a multicast a pro každý typ je možné využít různé typy směrovacích algoritmů.

Další možnou operací v NS2 je definice charakteru přenosu. Syntaxi definice zahrnuje opět jazyk Tcl.

Generátor dat může pracovat na základě analytických informací, nebo navázat na dříve definovanou aplikaci.

Poruchy lze definovat deterministicky nebo vytvoříme modul chyb, u kterého určíme typ a parametry pravděpodobnostní funkce pro generování chyb.

Monitorováním provozu sledujeme události spojené s pakety (vznik, ztráta zpracování, příchod k danému uzlu, atd.) i události spojené s frontami.

Po definici všech částí lze simulaci spustit opět příkazem, který definuje syntaxe jazyka Tcl.

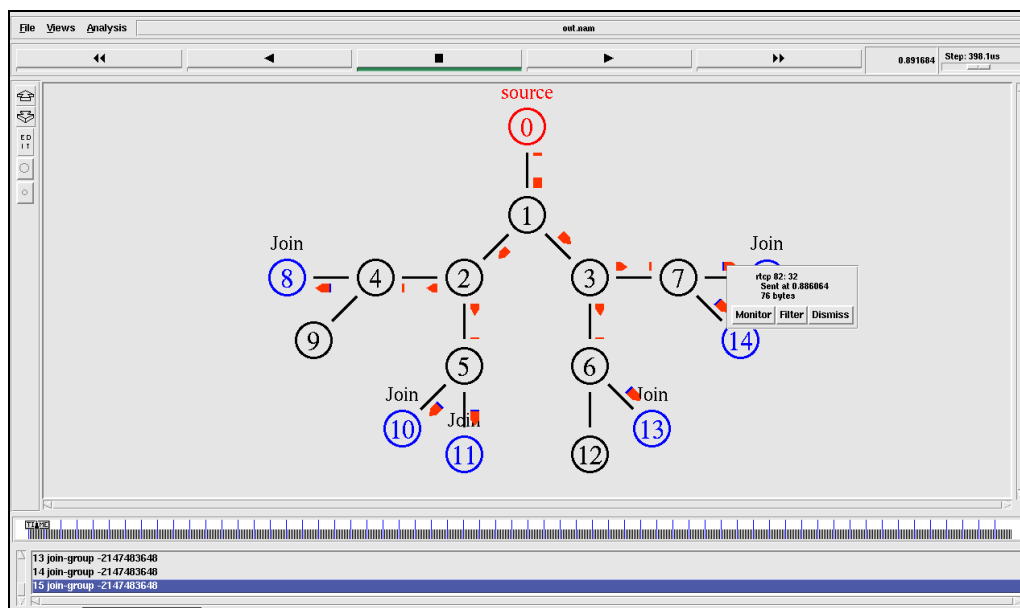
2.1 Příklady příkazů v NS2

- Vytvoření objektu reprezentujícího celý simulátor
set ns [new Simulator]
- Vytvoření linky
\$ns duplex-link \$n0 \$n1 <bandwidth> <delay> <queue_type>
- Definice přenosu využívajícího nespolehlivý transportní protokol (UDP)
set src [New Agent/UDP]
- Vytvoření zdroje, cíle a jejich propojení
set tcp [New Agent/TCP], set tcpsink [New Agent/TCPSink],
\$ns connect \$tcp \$tcpsink
- Vytvoření modelu aplikace FTP
set ftp [New Application/FTP]

2.2 Zobrazení výsledků

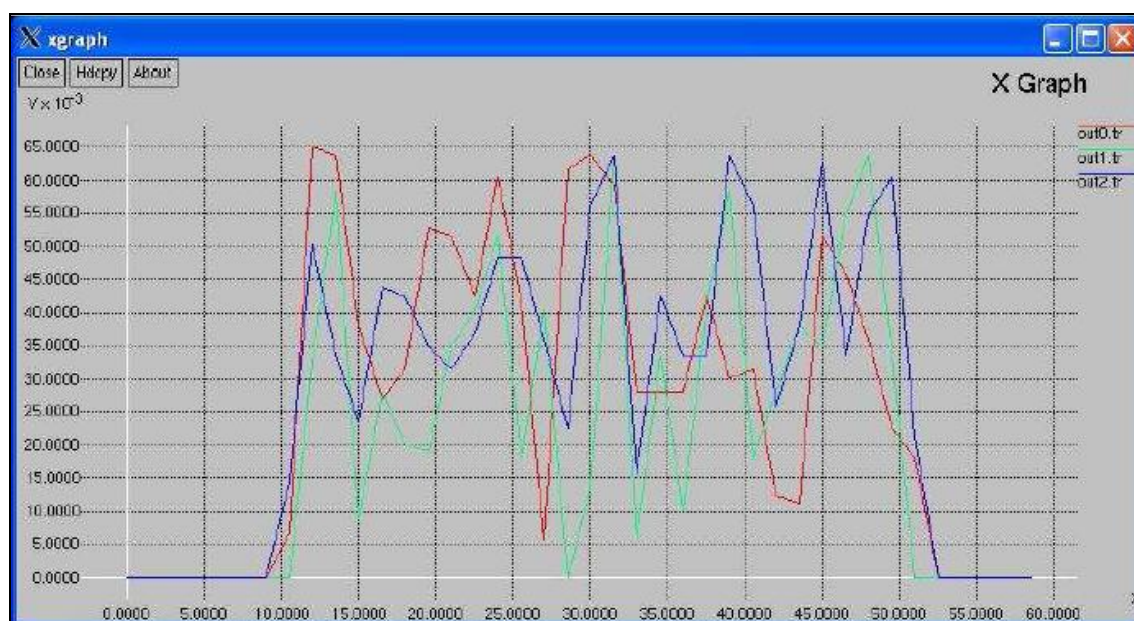
Zobrazení výsledků v NS2 se provádí nezávisle na simulačním nástroji NS2 externími aplikacemi.

S využitím nástroje Network Animator lze znázornit definovanou topologii a jednotlivé generované topologie (viz. Obr. 2.1).



Obr. 2.1: Ukázka simulace pomocí nástroje Network Animator

Časově závislé průběhy, jako například průběh plnění front atd. lze znázornit nástrojem *xgraph* (Viz. Obr. 2.2).



Obr. 2.2 : Ukázka časových průběhů zobrazených nástrojem *xgraph*.

3 Použité technologie při realizaci aplikace

3.1 Apache server

Apache je jednoduchý, ale přitom velmi výkonný web server, který je dostupný jak pro 32bitová Windows, tak i pro platformu Unix/Linux.

Prvním WWW serverem byl NCSA server, do kterého programátoři začali přidávat softwarové záplaty (anglicky a patch), aby se zlepšil výkon a funkčnost, a také v neposlední řadě i zamezilo bezpečnostním dírám. Postupem času z těchto záplat zrodil nový www server, kterému se začalo říkat Apache.

Proč je Apache tak úspěšný? Na tuto otázku si klade mnoho lidí tu samou odpověď. Apache nesrazil na kolena ani www server IIS od Microsoftu a to už je co říci. Jedna z mnoha odpovědí by mohla znít, že je zdarma, ale to je IIS také, dále, že je k dispozici zdrojový kód, možnost podílet se na vývoji, atd.

3.2 PHP

Na domovské stránce www.php.net je uvedeno, že: *"PHP je široce používaný obecně použitelný skriptovací jazyk s otevřenými zdrojovými kódy, obzvláště vhodný pro vytváření webovských aplikací."*

Jazyk PHP je podobný jazyku ASP a syntaxí k jazyku C nebo Javě a trochu i jazyku Perl. Za úspěch vděčí hlavně své jednoduchosti a lehkosti programování. Bez nadsázky se dá říci, že čas potřebný na vytvoření aplikace se zde neměří na měsíce, ale na týdny.

PHP je interpretovaný jazyk. To znamená, že program se nijak nepřekládá (nekompile) a vykonává se přímo ze zdrojového kódu. Takovým programům se někdy říká *skripty* a interpretovaným jazykům *skriptovací*. Uvedený přístup má výhodu ve své jednoduchosti a snadné modifikovatelnosti programů. Důležitou nevýhodou však je, že ke spuštění programů v PHP musíte mít nainstalován jeho interpret (tj. program, který kód vykoná). Dobrou zprávou je, že interpret PHP je volně šířen včetně zdrojových kódů (v souladu s tzv. PHP licencí) a můžete si ho bezplatně stáhnout třeba z domovské stránky jazyka.

Jazyk PHP prošel v minulosti poněkud bouřlivým vývojem. Zkratka "PHP" se kdysi četla jako *"Professional Home Pages"* (čili profesionální domovské stránky) a za léta,

kdy opustil domácí stránky a zabydlel se opravdu v profesionálním prostředí, se začalo vysvětlovat, že "PHP" je rekurzivním akronymem pro sousloví "**PHP: Hypertext Preprocessor**" (tedy PHP: hypertextový preprocesor (předzpracováváš)). Syntaxe se od prvních verzí přetransformovala a z dalšího vývoje lze vypožorovat, že se jazyk chce co nejvíce podobat Javě.

Jak už bylo zmíněno, hlavní zřetel tvůrců PHP je brán na vývojáře webovských aplikací tvořených dynamicky generovanými HTML stránkami. Příkazy PHP je možno psát přímo do samotné HTML stránky a ovlivňovat tak dynamicky její obsah. Zde je uvedena ukázka PHP kódu:

```
<html>
<head></head>
<body>
Dnes je <?php echo date(); ?>. Přejeme vám příjemný den.
</body>
</html>
```

To, co je vidět na příkladu je obyčejná HTML stránka obohacená o příkazy jazyka PHP, které se vkládají mezi značky <?php (popř. jen <?) a ?>. Ukázka vygeneruje stránku, ve které PHP vytiskne (echo) aktuální čas na serveru (date()).

Na druhou stranu omezení PHP pouze na tvorbu dynamických WWW stránek už taky není pravdou. Stejně dobře jako stránky je v něm možno psát programky podobné skriptům například v bashi na Linuxu nebo dávkovým (.BAT) souborům ve Windows.

3.2.1. Schopnosti, možnosti a vlastnosti

PHP může být provozováno na různých platformách včetně Linuxu, mnoha variant Unixu (včetně HP-UXu, Solarisu či OpenBSD), Microsoft Windows, Mac OS X, RISC OS a dalších. Dorozumí se bez problémů s mnoha dnešními web-servery a pro většinu z nich lze PHP zkompileovat jako modul (se zbytkem spolupracuje na základě CGI). V tabulce 3.1 je vidět přehled nejčastěji používaných webových serverů ve spojitosti s PHP.

Tab. 3.1: Nejpoužívanější webové servery podporující PHP

Apache	Xitami	Personal Web Server	Netscape a iPlanet servery
Microsoft Internet Information Server	OmniHTTPd	Oreilly Website Pro Server	Caudium

Výstupem z PHP je standardně soubor HTML, ale není to pouze omezeno na tento formát. Je možno psát skripty, jejichž výsledkem bude text ve formátu XHTML, XML nebo dokonce PDF. PHP umí generovat i obrázky nebo dokonce animace Flash. Znamená to tedy, že pomocí PHP je možno například udělat skript, který na požádání vytvoří graf vytížení systému v podobě GIF souboru. (vytvořené obrázky se nikam neukládají, jsou přímo *výsledkem vykonání* skriptu.)

V PHP je možno psát podle procedurálního paradigmatu nebo objektově. Velikou výhodou jazyka je množství vestavěných knihovných funkcí. Jsou to například tyto (kompletní seznam můžete nalézt v on-line dokumentaci): matematické funkce pro práci s libovolně přesnými čísly, *práce s kompresí ZIP (jen ke čtení)*, *Bzip2*, kalendářové funkce, *funkce pro datum a čas*, podpora COM ve Windows, *funkce pro práci se soubory*, propojení PHP s Javou, *kryptovací a hashovací funkce*, síťové funkce, *funkce pro přístup na FTP*, HTTP funkce, *funkce pro IMAP, POP3, NNTP a odesílání elektronické pošty*, funkce pro práci s procesy, *spouštění programů*, podpora semaforů, sdílené paměti a meziprocesorové komunikace, *funkce standardu POSIX*, regulární výrazy, *Shockwave Flash*, různé funkce pro práci s řetězci, *XML, XSLT, DOM XML* a mnoho dalších.

Velkým kladem PHP je, že zvládá spolupráci s mnoha databázemi (viz. Tab. 3.2).

Tab. 3.2.: Základní seznam databází, které podporuje PHP

Adabas D	dBase	Direct MS-SQL	Empress
FilePro (jen pro čtení)	FrontBase	Hyperwave	IBM DB2
Informix	Ingres	InterBase	mSQL
MySQL	Oracle (OCI7 a OCI8)	Ovrimos	PostgreSQL
Solid	Sybase	Unix dbm	Velocis

Navíc je pro PHP vyvinuta schopnost práce s ODBC (**O**pen **D**ata**B**ase **C**onnection), takže je možnost se z PHP připojit k jakékoliv další databázi podporující tento světový standard.

Napsání webových stránek pracujících s nějakým databázovým systémem je velice jednoduché. Pro ilustraci je zde uveden bohatě komentovaný příklad získání dat z MySQL:

```
<?php
```

```
// připojení k MySQL (v případě neúspěchu vypíše chybové
```

```
// hlášení a skončí)
```

```
$link = mysql_connect('localhost', 'uzivatel', 'heslo')
```

```
or die('Nemohu se připojit k MySQL!');
```

```
// výběr konkrétní báze dat
```

```
mysql_select_db('prace')
```

```
or die('Nepodařilo se vybrat bázi dat!');
```

```
// realizace SQL dotazu (získání všech dat z tabulky
```

```
// "pracovnici", kteří se jmenují "Josef")
```

```
$vysledek = mysql_query( 'SELECT * FROM pracovnici WHERE  
jmeno="Josef"');
```

```
// cyklus, ve kterém se v každém kroku zpracuje jeden řádek
```

```
// získaných dat
```

```
while ($radek = mysql_fetch_array($vysledek)) {
```

```
// tisk příjmení
echo $radek['prijmeni'] . '<BR>';
}
?>
```

A jak se v něm programuje? Je nutno říct, že především *rychle*. PHP jakožto skriptovací jazyk tolik nebazíruje na typech, proměnné ani funkce se nemusí deklarovat, jednoduše jsou použitelné. Například při použití proměnné \$promenna, je jenom na programátorovi, jestli do ní uloží číselnou hodnotu, řetězec, pole nebo objekt (a jestli tak k ní bude i přistupovat). Konverze mezi základními datovými typy (pokud to lze) je automatická, takže se při programování spíš soustředíme na samotný algoritmus než na to, jak jej v programovacím jazyku vyjádřit.

Jelikož se nikde nic nealokuje (lépe: PHP si vše alokuje samo), nemusí se ani hlídat velikosti místa pro proměnné: do řetězce lze stejně jednoduše uložit jeden znak jako 3 kB textu a totéž platí pro pole: pokud se programátor rozhodne uprostřed programu přidat do pole další položku, není nic snazšího: jednoduše ji *použije* v přiřazovacím příkazu a interpret PHP se sám postará o to, aby pro ni bylo v paměti místo.

Takových a podobných dalších vymožeností typických pro interpretované jazyky je v PHP spousta. Velice silným nástrojem (ač to na první pohled možná nevypadá) je možnost používat jako indexy pole kromě *jakýchkoliv* čísel i řetězce nebo dokonce hodnotu jakéhokoliv typu. Rázem tak dostává programátor do rukou mocný nástroj *hashovacích tabulek* (nebo *asociativních polí*, chcete-li), který je dobrým pomocníkem na každém kroku.

```
<?php
```

```
// Proměnné
```

```
$pocet = 3;
```

```
// Pokud použijete číselnou proměnnou v řetězcovém kontextu,
```

```
// PHP její hodnotu na řetězec automaticky převede
```

```
// Operátor tečky (".") znamená spojení řetězce
```

```
$retezec = 'Máme doma ' . $pocet . ' kočky.';
```

```
// Vytvoření nového pole
```

```
$pole = array('bílá', 'modrá', 'zelená');
```

```
// Přidání další položky do pole


$pole[] = 'černá';


// Přístup k položkám pole


$pole[0] = $pole[1];



// Indexem pole může být i řetězec


$pole['počet koček'] = $pocet;


?>
```

3.3 MySQL

- MySQL je relační databázový systém typu DBMS (database management system)
- MySQL je výkonný SQL server, kde je uložena SQL databáze, která je přístupná z PHP skriptů a využívá se pro ukládání a vyhledávání ve strukturovaných datech.
- každá databáze v MySQL je tvořena z jedné nebo více tabulek, které mají řádky a sloupce
- v řádcích rozeznáváme jednotlivé záznamy (řádek=záznam)
- sloupce mají jméno a uvozují datový typ jednotlivých polí záznamu (sloupec=pole)
- práce s databázemi, tabulkami a daty se provádí pomocí příkazů, respektive dotazů
- dotazy vycházejí z deklarativního programovacího jazyka SQL (Structured Query Language)
- systém MySQL je využitelný v C, C++, Java, Perl, PHP, Python, Tcl, Visual Basic, .NET
- MySQL je šířen jako Open source: <http://www.mysql.com/downloads/>, dokumentace: <http://www.mysql.com/documentation/>

4 Zabezpečení webových aplikací

Zabezpečení webových aplikací je důležitým krokem při návrhu. Důvodem je mimo jiné skutečnost zvyšujících se nároků programátora na vývojové nástroje, jejichž proměnné se nemusí při deklaraci vázat s datovým typem. Tyto programovací prostředky jsou užitečné z hlediska příjemnějšího programování, což lze s výhodou použít při implementaci zejména webových aplikací, avšak kvůli absenci typové kontroly lze poměrně snadno z nezabezpečené aplikace získávat data, která by neměla být přístupná. Mezi tyto programovací prostředky patří zejména v klientské části jazyk JavaScript, na straně serveru skript, jehož syntaxi definuje zejména norma jazyka PHP, případně ASP, a podobně. V této kapitole budou popsány základní možnosti útoků, jejich typů a zabezpečení proti nim.

4.1 Cíle útoků

Mezi cíle útoků ve většině případů patří tzv. **session ID**. Je to jednoznačný identifikátor relace mezi klientem a serverem. Používá se z toho důvodu, že protokol http je standardně nestavový, a identifikace relace je významná pro rychlejší zpracování dat na serveru. Tento identifikátor se ukládá do paměti na straně serveru i klienta. Když jej potenciální útočník získá, může na straně serveru vystupovat jako původní klient a získat o něm informace.

4.2 XSS (Cross Site Scripting)

Jedná se o útok prostřednictvím vložení nebezpečného kódu v klientském skriptovacím jazyce – např. JavaScript, JScript, VBScript apod. do „bezpečného“ obsahu webové stránky. Spočívá v jakési důvěře přístupu zejména neznalého uživatele, který si není vědom rizik spočívajících ve vstupních akcích (např. kliknutím na klamavou reklamu, což způsobí vykonání útočného kódu, atd.). Rozlišujeme tři druhy tohoto útoku:

1. Okamžitý útok (non-persistent) - je založen na okamžitém zobrazení (vykonání) nebezpečného skriptu.

2. Perzistentní útok - využívá možnosti běžného uživatele vkládat vlastní obsah na web (diskusní fóra, komentáře atp.).
3. Lokální útok - je velmi podobný charakteristikám okamžitého útoku, ale ke zpracování nebezpečného kódu se zneužije existujícího klientského skriptu. Tento skript může přistupovat k souborům na lokálních discích, což může mít za následek např. odeslání osobních údajů vzdálenému uživateli apod.

4.2.1. Zabezpečení proti XSS

Zabezpečení proti těmto typům útoků spočívá v zajištění nevykonání skriptovacích konstrukcí ve veřejných částech webu. To lze nejlépe docílit překódováním značek `<` `>` na entity. V PHP se k tomuto účelu používá funkce `htmlspecialchars()`. Další používanou funkcí je `strip_tags()`, která však nepřevádí entity, což u znaku `&` může způsobit chybné zobrazení stránky. `htmlspecialchars()` je doporučeno používat všude tam, kde se očekává neformátovaný text (příp. text formátovaný jinak, než pomocí HTML).

4.3 Zabezpečení databáze

Většina dotazů na webový server se děje pomocí jazyka SQL (Structured Query Language). Tento dotazovací jazyk je jednoduchý, obsahuje čtyři základní dotazy pro manipulaci s daty (SELECT, INSERT, UPDATE, DELETE) a je používán u většiny databázových systémů. Nebezpečí útoku na databázi spočívá v manipulaci s dotazem.

4.3.1. SQL injection

Nebezpečný uživatel (útočník) může z nezabezpečené databáze získat informace prostřednictvím SQL injection útoku. Jedná se o to, že například do vstupního uživatelského pole, kde se očekává pouze jednoznačná položka přidá i kus kódu, který ve výsledné podobě při zpracování na serveru může vypadat úplně jinak.

Příklad:

- Ve vstupním textovém poli se očekává jméno uživatele, pomocí něhož server vrátí klientovi číslo občanského průkazu.
Útočník však zadá: Novak' or 'a'='a
Tento řetězec se uloží do proměnné \$data.
- Na webovém serveru se nachází kód
mysql_query("Select číslo_OP where Jméno='\$data'");
- K databázovému server se dostane dotaz:
"Select číslo_OP where Jméno='Novak' or 'a'='a'"
- K uživateli se dostanou čísla OP všech jmen uložených v db tabulce.

4.3.2. Metody zabezpečení

Abychom zabezpečili databázi proti útoku popsaném v předchozí kapitole, je nutné převést kontrolní značky na data (před značky ' , " , \ a NULL zařadit zpětné lomítko \).

Příklad řešení v PHP:

- Použití funkce mysql_real_escape_string(), která se o převod znaků stará.
- Použití direktivy magic_quotes_gpc - automaticky nahrazuje speciální znaky ve všech příchozích (POST, GET, COOKIE) datech. Je nutné, pokud je tato direktiva zapnutá, ošetřování neprovádět nebo jej odstranit pomocí funkce stripslashes().V PHP pomocí funkce get_magic_quotes_gpc() zjistíme její nastavení.
- Použití databázové vrstvy, která umí oddělit v SQL dotazu data od zbytku dotazu (Příklad: MySQLi nebo PearDB).
- Vytvoření vlastní vrstvy

4.4 Zabezpečení citlivých údajů v databázi

4.4.1. Jednocestné zabezpečení

Provádí se nejčastěji pomocí hashovacích algoritmů. Vstupem hashování funkce je jakýkoliv řetězec jakékoliv délky, výstupem řetězec pevné délky nazývaný hash.

Používané hashování algoritmy jsou např. MD2, MD4, MD5, RIPEMD, SHA-0, SHA-1, SHA-256/224, SHA-512/384)

Požadavky na hashování algoritmy:

1. **bezkoliznost** – Dva různé řetězce mají stejný hash - v praxi nemusí vždy platit (převádí se libovolně dlouhý řetězec na řetězec omezené délky obsahující znaky pouze z určité množiny).
2. **lavinovitý efekt** - v originále nazýván jako *avalanche effect* – jde o vlastnost algoritmu, kdy malá změna na vstupu způsobí nesrovnatelně větší změnu na výstupu (typicky se při převrácení hodnoty jednoho bitu ve vstupním řetězci změní více než polovina bitů ve výsledném otisku).

Prolomit jednocestné zabezpečení je pro útočníka problém. Nezbyvá mu jiná možnost než využití metody síly (brute force). V případě, že heslo je kombinací několika znaků, např. malých písmen, je velmi vysoká pravděpodobnost, že útočník heslo odhadne. Milióny hashů, které jsou online k dispozici mohou sloužit k prolomení databáze zakódovaný systémem MD5. Dnes se od MD5 ustupuje kvůli jeho náchylnosti ke kolizím a doporučuje se využívat SHA-1 (nebo lépe SHA-256/224 a vyšší).

V PHP lze použít funkce md5() a sha1(). Tyto funkce přímo generují výsledný hash na základě parametru.

4.5 Nastavení serverového prostředí používající jazyk PHP

Pokud používáme jazyk PHP, je nutné v první řadě správně nastavit prostředí, na kterém pracuje. V naprosté většině případů jde o webový server Apache. Je nutné pomocí konfigurace znemožnit např. prohlížení zdrojových kódů PHP přímo z webu a takto zabezpečit všechny ostatní datové typy, které by mohly způsobovat problémy. Také výpis adresáře pomocí HTTP požadavku je dobré dovolovat jen lokálně a to pomocí souboru .htaccess s místním nastavením.

Velmi důležité je mít správně nastavenou adresářovou strukturu a manipulovat s oprávněními s extrémní opatrností.

Co se týče samotného skriptovacího jazyka, zde je především nutné správně nakonfigurovat direktivy. Následující direktivy by měly být z bezpečnostního hlediska **vypnuty**:

- `safe_mode` – nastavuje bezpečný režim, je však nespolehlivá a do PHP 6 se s ní nepočítá.
- `display_errors` – zobrazování chyb je nebezpečné, může útočníkovi značně pomoci. Pokud je to nutné, nastavovat lokálně pomocí `error_reporting()`.
- `magic_quotes_sybase` – mění zásadním způsobem ošetřování speciálních znaků v SQL dotazu.
- `register_globals` – opět velmi nevhodné, otevírá možnost útoku pomocí vložených proměnných.

Naopak, následující direktivy je vhodné **používat** (nastavit):

- `open_basedir` – zamezuje ze skriptu přístup mimo zadaný adresář
- `disable_functions` – můžete specifikovat zakázání určitých funkcí (`exec`)
- `session.use_only_cookies` – nebudou se brát v potaz session ID přicházející z GET nebo POST dat.
- `session.save_path` a `upload_tmp_dir` – nastavují cestu k ukládání sessions a s dočasných dat. Vhodné nastavit na bezpečné místo.

5 Návrh modelu webové aplikace

5.1 Požadavky na výsledné chování

5.1.1. Požadavky na výsledné chování aplikace

- Uživatel bude moci přistupovat k aplikaci zadáním DNS (Domain Name System) adresy ve webovém prohlížeči
- Stránky webové aplikace přístupné **běžnému uživateli**:
 - Informace o NS2 – popis simulačního nástroje, hlavní charakteristika
 - Projekty – Seznam odevzdaných projektů a datum jejich vložení
 - **Zobrazení detailu** o kterémkoliv odevzdaném projektu (Jméno autora, abstrakt, možnost stáhnutí zdrojových souborů a obrázků na svůj lokální disk)
 - Vložení nového projektu – bude obsahovat formulář, ve kterém uživatel vyplní:
 - Název projektu
 - Abstrakt
 - Vybere ze svého disku soubor s obrázkem (.jpg,...)
 - Vybere ze svého disku soubor se zdrojovým kódem (.tcl)
 - Vybere ze svého disku soubor s dokumentací (.*)
 - Do textových polí napíše své jméno a příjmení (povinné), fakultu, ročník, e-mail (povinné), ICQ
 - Stisknutím tlačítka „Odevzdat ke schválení“ uloží projekt na server a zároveň pošle administrátorovi zprávu, aby jej schválil
 - Diskusní fórum
- Stránky webové aplikace přístupné **pouze administrátorovi** aplikace:
 - Stránka administrátora, kde může nastavovat a měnit tyto data:
 - E-mail administrátora

- Heslo administrátora
- Aktivace a deaktivace odevzdaného projektu
- Úprava odevzdaných projektů, pokud ještě nebyly aktivovány (zpřístupněny ke zobrazení detailu pro běžného uživatele)
- Úplné vymazání projektu z databáze a z diskového prostoru
- Přidání nového administrátora

5.1.2. Další požadavky

- Zabezpečení webové aplikace
- Zařazení projektů do jednotlivých kategorií
- Vyhledávání v projektech pomocí klíčového slova
- Řazení projektů podle příjmení autora, názvu projektu, data

5.1.3. Požadavky na databázi

- Uchování informací o studentech (autorech) projektů (Jméno, příjmení)
- Uchování informací o projektu (Název, Datum vložení, příznak aktivity,...)
- Uchování názvů souborů obrázků
- Uchování názvů souborů zdrojových kódů
- Uchování informací o kategorii a protokolu, které daný projekt využívá
- Uchování informací o administrátorovi

5.1.4. Požadavky na implementační jazyk a použité technologie

- Operační systém: Linux
- Implementační jazyk: PHP
- Webový server: Apache
- Databázový server: MySQL

5.2 Návrh databáze webové aplikace

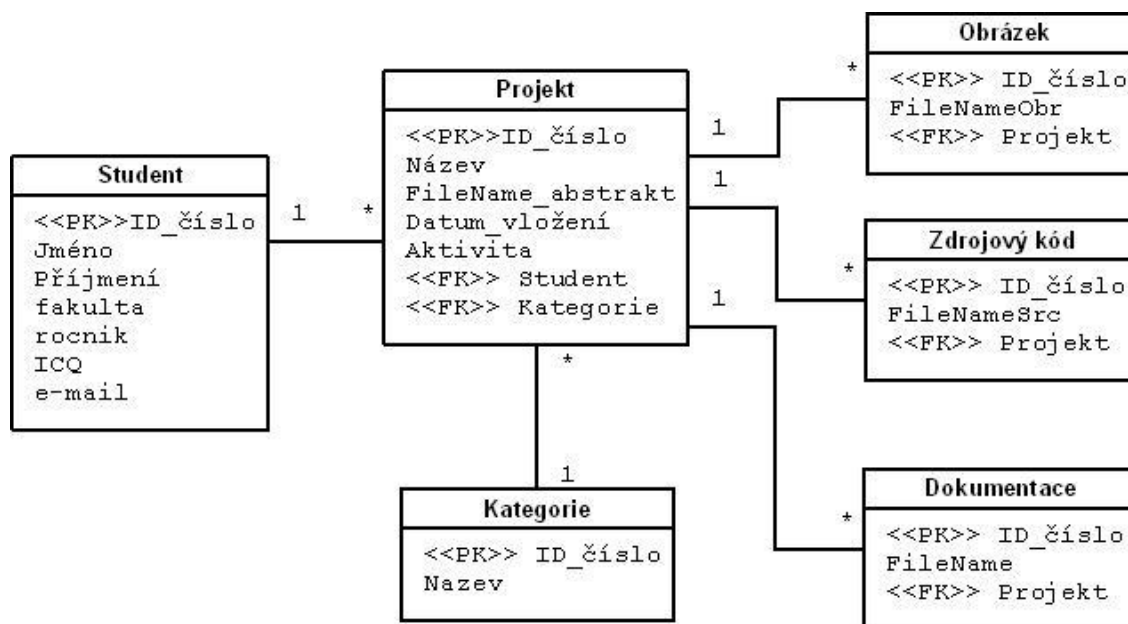
Návrh databáze se popisuje nejčastěji tzv. *ER diagramem* a *schématem databáze*.

ER diagram se definuje pomocí objektů – entit (Entity) a vztahů mezi nimi (Relationship). Někdy bývá označován jako ERA (A – označuje atributy, které daná entita obsahuje).

ER diagram lze převést na *schéma databáze*, které slouží k informaci o struktuře jednotlivých tabulek databáze, ze které lze snadno vytvořit programový model například v prostředí MySQL pro vytvoření tabulek databáze.

Oba zmíněné popisy závisí na požadavcích na aplikaci (viz. kap. 5.1.1, 5.1.2), a jsou dále uvedeny v této kapitole.

5.2.1. ER diagram



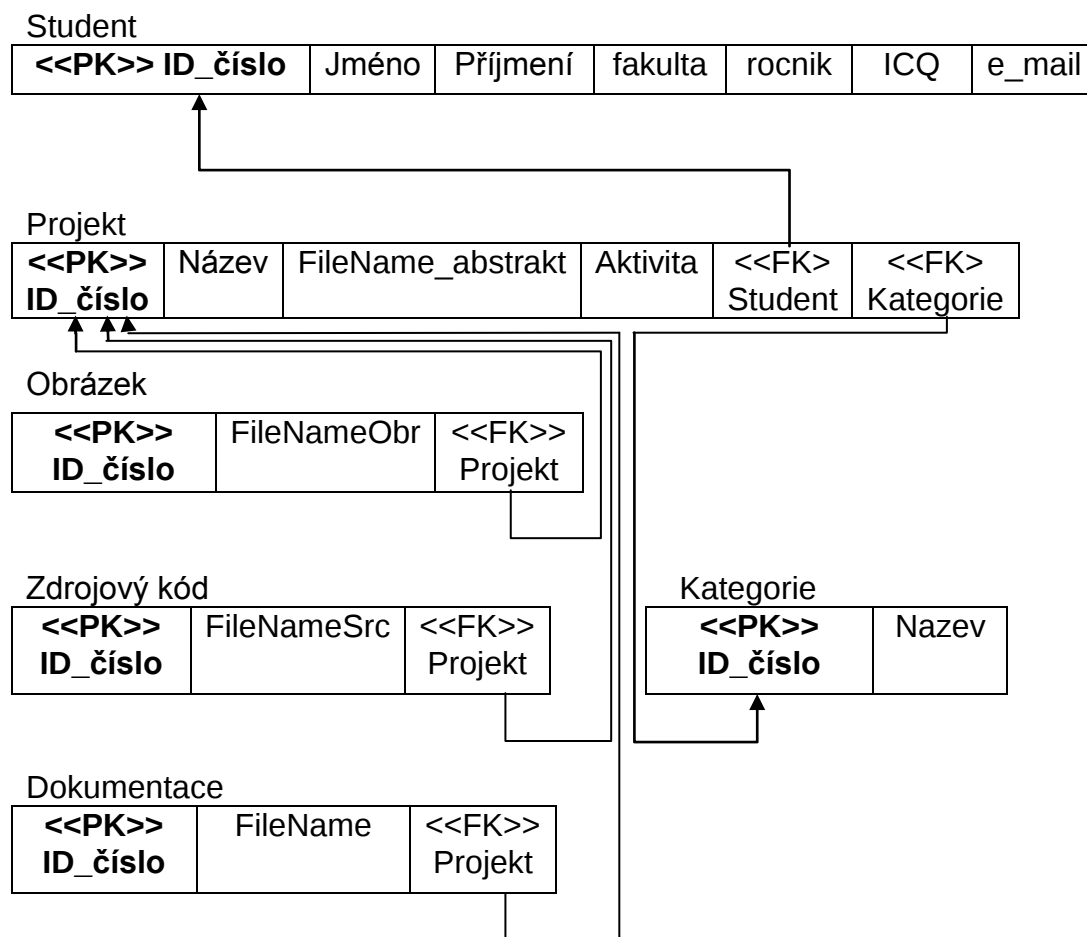
Obr. 5.1: ER diagram navrhované webové aplikace.

Popis obrázku 5.1:

Entita student obsahuje atributy ID_číslo, které jej jednoznačně identifikuje, Jméno, Příjmení, fakulta, ročník, ICQ a e-mail. Student může vlastnit více projektů. Projekt je

jednoznačně identifikován ID_číslem, a následují další atributy. Název, FileName_abstrakt, který definuje název textového souboru, kde je uložen abstrakt, dále pak datum vložení projektu, aktivitu, která nabývá dvou stavů podle toho, zda administrátor projekt schválil, či nikoliv. Od projektu jsou vedeny vazby jednak na entitu Obrázek, ve které FileNameObr definuje název souboru obrázku, obdobně jsou vazby na Zdrojový kód, kde je název souboru zdrojového kódu určen atributem FileNameSrc a vztah k entitě Dokumentace. Ke každému projektu se vztahuje jedna kategorie, do které projekt náleží.

5.2.2. Schéma databáze



Obr. 5.2: Shéma databáze navrhované webové aplikace vycházející z ER diagramu.

6 Funkčnost aplikace

V této kapitole budou popsány konkrétní části již nainstalované a běžící aplikace na serveru (instalace aplikace bude uvedena v příloze).

6.1 Programovací jazyk

Programovacím jazykem aplikace byl dle požadavků v zadání práce zvolen PHP (podrobnější popis je uveden v kapitole 3.2).

6.2 Volba internetového prohlížeče

Aplikace byla otestována a běží bez problému v MS Internet Explorer verze 6.0.2900. Proto je tento prohlížeč doporučeným nástrojem pro užívání aplikace. Taktéž byla testována v internetovém prohlížeči Mozilla Firefox verze 2.0.0.20. V této verzi vykazovala drobné problémy se znakovou sadou u aplikační části diskusní fórum. Tuto disfunkci lze v aplikaci potlačit pomocí menu Zobrazit – Znaková sada – Unicode (UTF – 8).

6.3 Struktura html části PHP souboru

Aplikace je tvořena tak, že při kliknutí na jednu položku aplikace se vyvolá jeden soubor obsahující hypertextovou část a PHP kód. Strukturu hypertextové (html) části představují html značky (tzv. tagy). Tělo webové stránky obsahuje blokové elementy (div) využívající kaskádové styly (CSS) pro grafický design aplikace. Tyto elementy jsou z důvodu kompatibility s webovým prohlížečem Mozilla Firefox ještě v některých souborech členěny do tabulky (table).

6.4 Význam nejdůležitějších částí aplikace

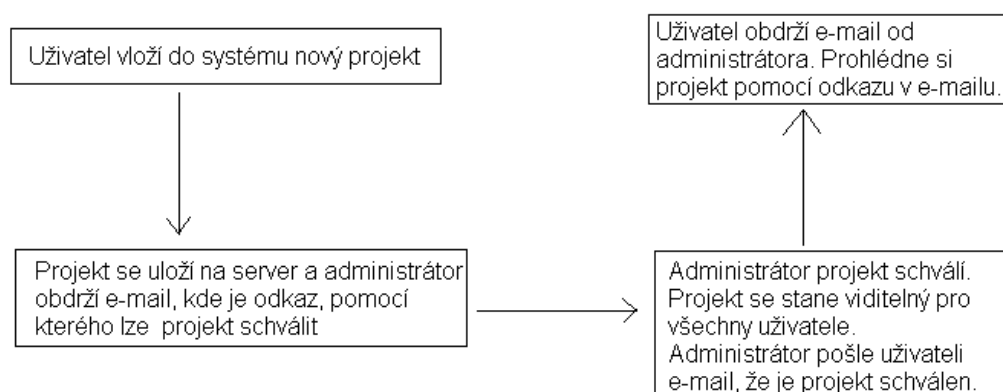
6.4.1. Hlavní stránka

Hlavní stránka aplikace obsahuje informaci o nástroji Network Simulator 2.

6.4.2. Vložení nového projektu

Tato internetová stránka slouží běžnému uživateli ke vkládání nového projektu do systému. To znamená, že když uživatel vkládá projekt poprvé, informace o něm zadané v poli Jméno, Příjmení, e-mail a podobně se uloží do databázové tabulky, informace o projektu se uloží do dalších tabulek (schéma databáze viz kap. 5.2.2.). Pokud tentýž autor ukládá další projekt, tak se nejdříve zjistí existence záznamu o něm v databázové tabulce, pokud záznam existuje tak se vloží pouze informace o projektu, a soubory se uloží na server do adresáře „UPLOADED_PROJECTS/Prijmeni/Datum_a_cas“ stejně jako v předchozím případě.

Reakci aplikace na vložení nového projektu zobrazuje následující obrázek.



Obr. 6.1. Reakce aplikace na vložení nového projektu.

6.4.3. Projekty

Po kliknutí myši v menu na položku „Projekty“ se v levé části obrazovky zobrazí seznam aktivních to znamená schválených projektů administrátorem. Nad nimi lze pomocí rádiových tlačítek provést řazení těchto projektů podle příjmení autora, názvu projektu a data vložení. Také je zde zavedena funkce hledání výrazu napsaného v textovém poli. Po kliknutí na název projektu v seznamu projektů se zobrazí jeho detail.

6.4.4. Detail projektu

Tato stránka podobně jako stránka „Projekty“ popsána v předchozí kapitole zobrazuje seznam projektů a taktéž detail projektu, na který uživatel kliknul ve stránce „Projekty“. Detail projektu, který lze shlédnout v pravé části obrazovky sestává z dat vložených při vložení nového projektu. Výčtově se jedná o název projektu, abstrakt, kategorie, do

které projekt patří (IP síť, bezdrátové IP síť, bezdrátové senzorové síť), dále pak obrázek, zdrojový kód, dokumentaci a informaci o autorovi.

6.4.5. Správa systému – Administrátor

K práci s tímto oknem se administrátor dostane kliknutím na odkaz „admin“ z libovolné stránky aplikace. Systém si vyžádá přihlašovací údaje e-mail a heslo. Pro prvotní použití po instalaci aplikace jsou nastaveny tyto přihlašovací parametry:

- e-mail: simek@feec.vutbr.cz
- heslo: bbb

Heslo je z důvodu bezpečnosti ukládáno v hashované podobě do databázové tabulky.

Ukázka kódu ověřování e-mailu a hesla:

```
$conn = mysql_connect($db_server,$db_login,$db_heslo);
```

```
if(!$conn) echo "Chyba prihlasovani";
```

```
$SQL_text = "select ID_cislo,email,password from diplomka.admin_info ";
```

```
$RES = mysql_query($SQL_text, $conn) or die("Fail execution query select  
password admin !");
```

```
if((!empty( $_POST['e-mail_admin']))&& (!empty($_POST['password_admin'])))
```

```
{
```

```
    $nalezen=false;
```

```
    while($r = mysql_fetch_row($RES))
```

```
    {
```

```
if(((($_POST['email_admin']==$r[1])and(md5($_POST['password_admin'])==$r[2]))) {
```

```
    $nalezen=true;
```

```
    $ID_admin = $r[0];
```

```
    break;
```

```
}
```

```
}
```

```
if($nalezen==true)
```

```
{
```

//souhlasí, nastavíme session, může jich být více a můžeme v nich přenášet třeba jméno
//přihlášeného...

```
$_SESSION['prihlasen']=1;
$_SESSION["ID_admin"]=$ID_admin;
//a znovunačteme stránku (předpokládám že je to index v kořeni webu
header("location: ./web_admin.php?zvyrazni_ID=".$_POST['zvyraz_ID']);
exit;
}
else
{
//nesouhlasí
$sprava='<p>Litujeme, přihlašovací údaje nejsou platné.</p>';
echo $sprava;
}
}
if(!isset($_SESSION['prihlasen'])) {
$sprava='<p>Litujeme, přihlašovací údaje nejsou platné.</p>';
echo $sprava;
exit;
}
?>
```

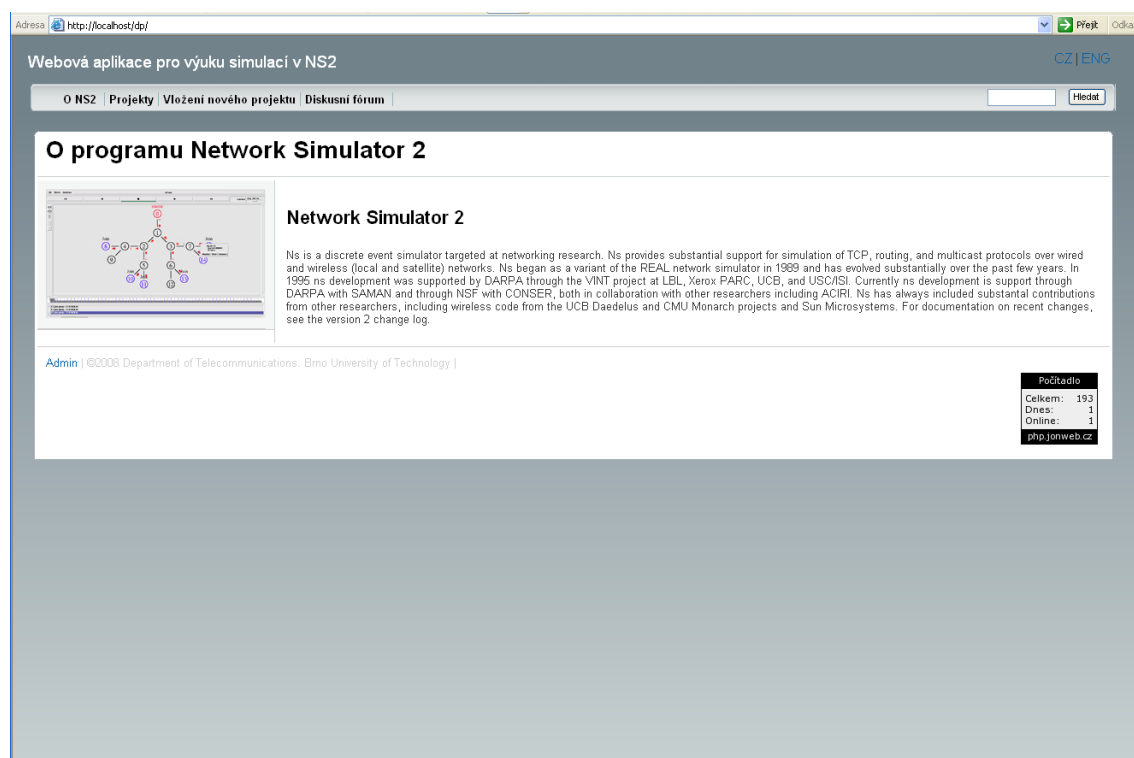
Hlavním úkolem administrátora je v tomto režimu schvalování projektů, jejich případná modifikace a mazání. Projekt je neaktivní (nepřístupný uživatelům k nahlédnutí), pokud je u projektu ve sloupci „stav“ zobrazen křížek. Projekt lze učinit aktivním, když administrátor klikne na křížek. Pokud je u projektu ve sloupci „stav“ fajfka, je projekt přístupný uživatelům k náhledu v menu sekci „Projekty“ a autorovi tohoto projektu se doručí e-mail s oznámením, že jeho projekt byl administrátorem schválen.

Dále si administrátor může měnit své přihlašovací údaje a přidávat nového administrátora.

6.4.6. Diskusní fórum

Poslední částí aplikace je diskusní fórum. Umožňuje registrovaným a přihlášeným uživatelům psát příspěvky. Dále zobrazuje statistiku, a hledat zadaný výraz.

7 Ukázky aplikace



Obr. 7.1: Úvodní strana aplikace.

V této kapitole jsou uvedeny ukázky grafické podoby aplikace vypadající ve finálním stavu. Na obrázku Obr. 7.2 je uvedena internetová stránka, která slouží uživateli pro vkládání jeho projektu.

Po stisknutí tlačítka „Odeslat ke schválení“ (viz. Obr. 7.2) je administrátorovi aplikace odeslána zpráva, aby potvrdil registrovaný projekt.

Administrátor obdrží e-mail uvedený na obrázku 7.3.

Po kliknutí na odkaz v e-mailu se administrátorovi v systému zobrazí zvýrazněný řádek s neaktivním projektem (viz. Obr. 7.5). Ten lze potvrdit kliknutím na políčko ve sloupci „stav“. Poté se projekt stane viditelným pro ostatní uživatele.

Kliknutím tlačítka myši na název projektu se zobrazí jeho detail (viz. Obr. 7.9).

--Úvodní strana--Projekty--

Vložení nového projektu - student

Název projektu*

Abstrakt (max. počet znaků je 500)*

Kategorie*

1) IP síť

Načíst obrázek ('.jpg', '.gif', '.bmp', '.png', max. velikost 1.5MB) *

Vyberte soubor:

Načíst zdrojový kód ('.tcl', '.txt', max. velikost 1.5MB) *

Vyberte soubor:

Načíst dokumentaci (*.*, max. velikost 5MB) *

Vyberte soubor:

Jméno autora*:

Příjmení autora*:

Fakulta:

Rocník:

Kontakt

E-mail*:

ICO:

Údaje označené hvězdičkou jsou povinné!

Obr. 7.2: Stránka aplikace sloužící ke vkládání nových projektů.

Datum: Sun, 09 Dec 2007 16:31:42 +0100
Od: pavlosek@kn.vutbr.cz
Komu: pavlosek@kn.vutbr.cz
Předmět: Potvrďte projekt studenta Majer Petr
Potvrzení provedete na této stránce: http://90.176.45.1/dp/web_admin.php?zvyrazni_ID=12

Obr. 7.3: Obsah e-mailu, který se uloží do e-mailové schránky administrátora po odevzdání projektu.

Webová aplikace pro výuku simulací v NS2 CZ ENG

[O NS2](#) [Projekty](#) [Diskusní fórum](#)

Přihlásit se jako administrátor

E-mail:

Heslo:

[Admin](#) | ©2008 Department of Telecommunications, Brno University of Technology |

Počítadlo

Celkem: 193
Dnes: 1
Online: 1
php.jonweb.cz

Obr. 7.4: Stránka pro přihlášení administrátora systému.

Webová aplikace pro výuku simulací v NS2 CZ ENG

[O NS2](#) [Projekty](#) [Diskusní fórum](#)

Správa systému - Administrátor

Implementace knihovny UMTS
Simulace multicastového směrování BST
Simulace unicastového směrování OSPF
Simulace hybridní bezdrátové IP sítě 802.11
Simulace bezdrátové IP sítě 802.11
Kvalita služeb pro službu VoIP
Kvalita služeb pro službu Triple Play

Zpráva o schválení nebyla doručena autorovi projektu

NASTAVENÍ INFORMACÍ O ADMINISTRÁTOROVI

Název	Hodnota	
E-mail	pavlosek@kn.vutbr.cz	✓
Heslo do tohoto systému	...	✓

PŘIDAT NOVÉHO ADMINISTRÁTORA

E-mail: Heslo:

Vložené projekty

Příjmení a jméno studenta	Název projektu	Datum vložení projektu	Stav	Úplně vymazat
Kovář Petr	Implementace knihovny UMTS	04.01.2009	✗	✗
Kovář Petr	Simulace multicastového směrování BST	04.01.2009	✓	✗
Šimek Milan	Simulace unicastového směrování OSPF	04.01.2009	✓	✗
Šimek Milan	Simulace hybridní bezdrátové IP sítě 802.11	04.01.2009	✓	✗
Šimek Milan	Simulace bezdrátové IP sítě 802.11	04.01.2009	✓	✗
Šimek Milan	Kvalita služeb pro službu VoIP	04.01.2009	✓	✗
Šimek Milan	Kvalita služeb pro službu Triple Play	04.01.2009	✓	✗

Počítadlo

Celkem: 193
Dnes: 1
Online: 1
php.jonweb.cz

[Odhlásit se](#) | ©2008 Department of Telecommunications, Brno University of Technology

Obr. 7.5: Stránka přístupná pouze administrátorovi. Jsou vidět aktivní a neaktivní (ve sloupci stav je křížek) projekty.

Webová aplikace pro výuku simulací v NS2

CZ ENG

O NS2

Projekty

Diskusní fórum

Úvodní strana

Projekty

Detail projektu

Název	Autor	Datum
Simulace multicastového směrování BST	Kovář	04.01.2009
Simulace unicastového směrování OSPF	Šimek	04.01.2009
Simulace hybridní bezdrátové IP sítě 802.11	Šimek	04.01.2009
Simulace bezdrátové IP sítě 802.11	Šimek	04.01.2009
Kvalita služeb pro službu VoIP	Šimek	04.01.2009
Kvalita služeb pro službu Triple Play	Šimek	04.01.2009

Název projektu:

Simulace multicastového směrování BST

Abstrakt:

Simulace funkcí multicastového směrování BST

Kategorie:

1) IP sítě

Obrázek:

Zdrojový kód:

80211-2.tcl

Dokumentace:

Jméno autora:

Petr

Příjmení autora:

Kovář

Fakulta:

FEKT, ÚTKO

Kontakt

E-mail:

simsek@feec.vutbr.cz

ICQ:

Aktualizovat

Obr. 7.6: Detail projektu zobrazený v administrátorském režimu. Administrátor může údaje měnit a uložit stisknutím tlačítka „Aktualizovat“.

Webová aplikace pro výuku simulací v NS2 CZ ENG

O NS2 | Projekty | Vložení nového projektu | Diskusní fórum

--Úvodní strana--

Projekty

Seřadit podle:

☐ Příjmení autora
☒ Název projektu
☐ Data vložení

Název	Autor	Datum
Kvalita služeb pro službu Triple Play	Šimek	04.01.2009
Kvalita služeb pro službu VoIP	Šimek	04.01.2009
Simulace bezdrátové IP sítě 802.11	Šimek	04.01.2009
Simulace hybridní bezdrátové IP sítě 802.11	Šimek	04.01.2009
Simulace multicastového směrování BST	Kovář	04.01.2009
Simulace unicastového směrování OSPF	Šimek	04.01.2009

Admin | ©2008 Department of Telecommunications, Brno University of Technology

Počítadlo

Celkem: 193
 Dnes: 1
 Online: 1
 php.jonweb.cz

Obr. 7.7: Seznam projektů, autoři a datum vložení.

Webová aplikace pro výuku simulací v NS2 CZ ENG

O NS2 | Projekty | Vložení nového projektu | Diskusní fórum

--Úvodní strana--

Projekty

Seřadit podle:

☐ Příjmení autora
☐ Název projektu
☐ Data vložení

Výsledky hledání

[Kvalita služeb pro službu Triple Play](#)
[Kvalita služeb pro službu VoIP](#)

Autor: Milan Šimek
 Projekt je schválen

Název	Autor	Datum
Kvalita služeb pro službu Triple Play	Šimek	04.01.2009
Kvalita služeb pro službu VoIP	Šimek	04.01.2009
Simulace bezdrátové IP sítě 802.11	Šimek	04.01.2009
Simulace hybridní bezdrátové IP sítě 802.11	Šimek	04.01.2009
Simulace multicastového směrování BST	Kovář	04.01.2009
Simulace unicastového směrování OSPF	Šimek	04.01.2009

Admin | ©2008 Department of Telecommunications, Brno University of Technology

Počítadlo

Celkem: 193
 Dnes: 1
 Online: 1
 php.jonweb.cz

Obr. 7.8: Výsledky hledání výrazu.

--Úvodní strana--Projekty--

Detail projektu

Název	Autor	Datum
Simulace multicastového směrování BST	Kovář	04.01.2009
Simulace unicastového směrování OSPF	Šimek	04.01.2009
Simulace hybridní bezdrátové IP sítě 802.11	Šimek	04.01.2009
Simulace bezdrátové IP sítě 802.11	Šimek	04.01.2009
Kvalita služeb pro službu VoIP	Šimek	04.01.2009
Kvalita služeb pro službu Triple Play	Šimek	04.01.2009

Název projektu:

Kvalita služeb pro službu Triple Play

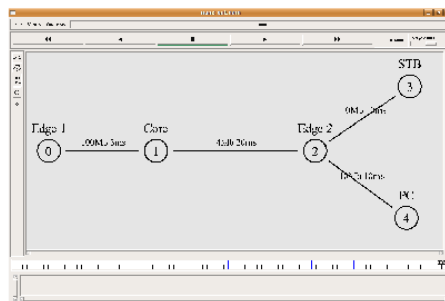
Abstrakt:

Simulace přístupové sítě se službou IPTV, VoIP a s datovými přenosy. Z teoretického úvodu se dozvíte jak fungují metody obsluhy front Round Robin a Weighted Round Robin. Dočtete se jaké způsoby označování (Policy) paketů v NS2 existují a jak probíhá počítání pravděpodobnosti zahazování. V praktické části budete simulovat síť pro IPTV provoz, kterou následně upravíte na síť poskytující Triple play.

Kategorie:

IP sítě

Obrázek:



Zdrojový kód:

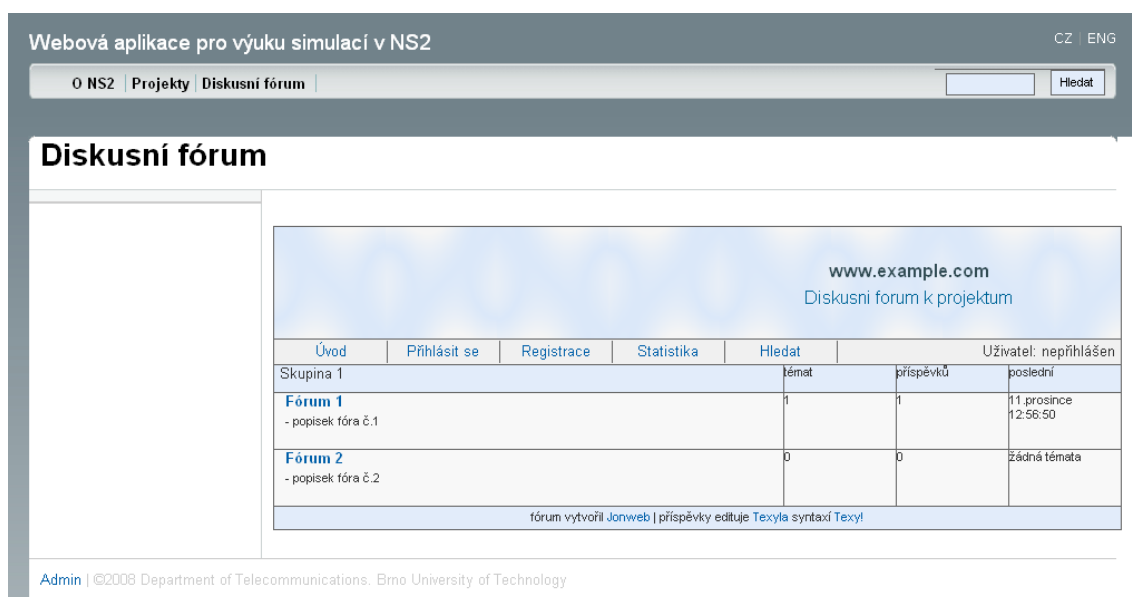
[lab2-castA.tcl](#)

Dokumentace:

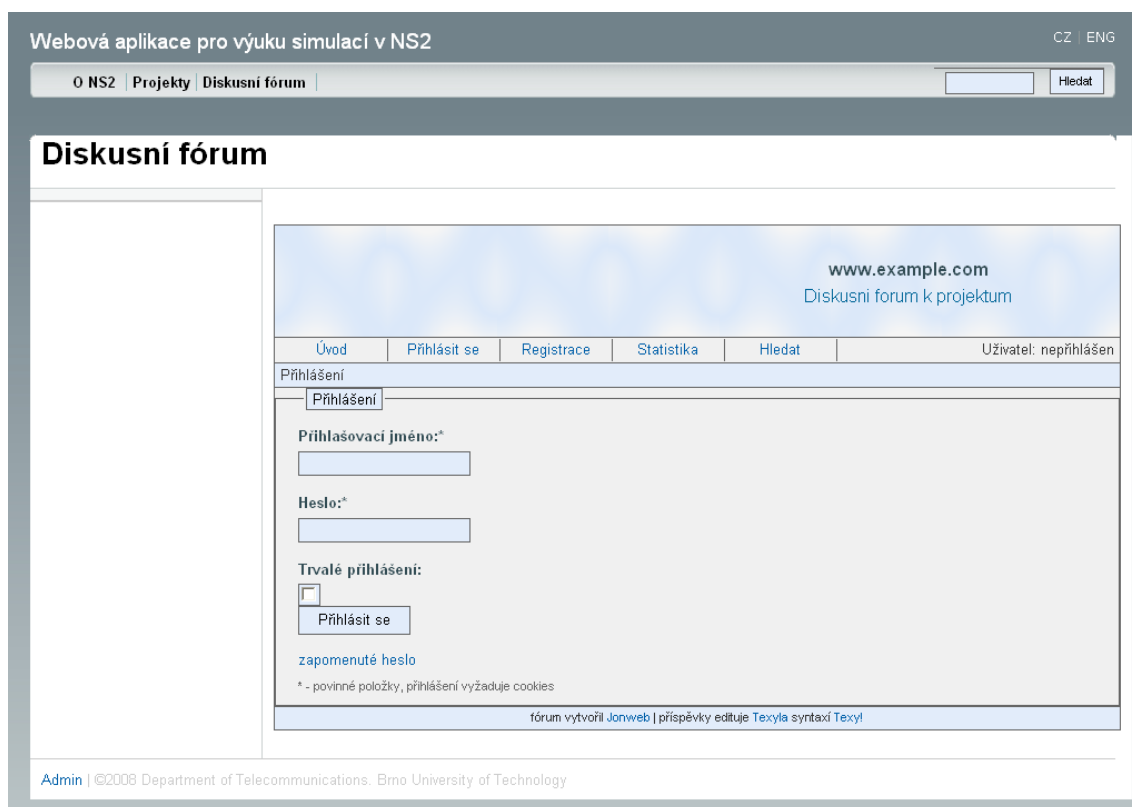
[\[\]](#)

Jméno autora: Milan
Příjmení autora: Šimek
Fakulta: FEKT, ÚTKO
Kontakt:
E-mail: simek@feec.vutbr.cz
ICQ:

Obr. 7.9: Detail projektu. Podrobněji o detailu projektu viz kapitola 6.4.4.



Obr. 7.10: Hlavní stránka diskusního fóra.



Obr. 7.11: Přihlašovací formulář diskusního fóra.

Úvod	Přihlásit se	Registrace	Statistika	Hledat	Uživatel: nepřihlášen
----------------------	------------------------------	----------------------------	----------------------------	------------------------	-----------------------

Registrace nového uživatele

[Registrace](#)

Povinné údaje:

Jméno:

Heslo:

Heslo znovu:

Opište číslici: * šest

Nepovinné údaje:

WWW stránky:

Email:

ICQ:

Jabber ID:

Skype:

MSN Messenger:

Obr. 7.12: Registrace uživatele diskusního fóra.

Webová aplikace pro výuku simulací v NS2 CZ ENG

[O NS2](#) [Projekty](#) [Diskusní fórum](#)

Diskusní fórum

www.example.com
Diskusni forum k projektu

Úvod	Přihlásit se	Registrace	Statistika	Hledat	Uživatel: nepřihlášen
----------------------	------------------------------	----------------------------	----------------------------	------------------------	-----------------------

Vyhledávání

Hledaný výraz:

fórum vytvořil Jonweb | příspěvky edituje Texyla syntaxí Texyl

[Admin](#) | ©2008 Department of Telecommunications, Brno University of Technology

Obr. 7.13: Vyhledávání.

Webová aplikace pro výuku simulací v NS2

CZ ENG

O NS2

Projekty

Diskusní fórum

Hledat

Diskusní fórum

www.example.com

Diskusní forum k projektu

Úvod

Přihlásit se

Registrace

Statistika

Hledat

Uživatel: nepřihlášen

Statistika fóra

Přístupnost:

on-line	přístupů dnes	přístupů celkem	zobrazení stránek
1	0	34	152

Obsah fóra:

témat	příspěvků	registrovaných uživatelů
1	1	1

Nejaktivnější uživatelé:

jméno	příspěvků

Nejdiskutovanější témata:

téma	odpovědí

Nejnávštěvovanější témata:

téma	zhlédnutí
Téma 1	4

fórum vytvořil [Jonweb](#) | příspěvky edituje [Texyla](#) syntaxí [Texyl](#)

Obr. 7.14: Statistika využití diskusního fóra.

8 Ukázky zdrojových kódů

8.1 Vytvoření databázových tabulek v programu MySQL

Zdrojové kódy pro vytvoření jednotlivých tabulek korespondují s obrázkem 5.2 uvedeným v kapitole 5.2.2.

Pro vytvoření SQL tabulky se užívá syntaxe:

```
create table Jméno_tabulky ( Název_prvního_sloupce datový_typ  
[doplňující_informace], Název_druhého_sloupce datový_typ [doplňující_informace],  
...  
Název_N-tého_sloupce datový_typ ,[doplňující_informace], Výčet_klíčů )  
Typ_tabulky;
```

Ukázka zdrojového kódu pro vytvoření tabulky Student:

```
create table Student (ID_cislo SMALLINT UNSIGNED NOT NULL  
AUTO_INCREMENT,  
Jmeno varchar(10),  
Prijmeni varchar(25),  
fakulta varchar(20),  
rocnik varchar(10),  
ICQ varchar(20),  
e_mail varchar(40) NOT NULL,  
PRIMARY KEY (ID_cislo)  
) TYPE=INNODB;
```

Stručný popis:

Tabulka „Student“ reprezentuje uživatele, kteří do systému vložili svůj projekt. První sloupec „ID_cislo“ je důležitý pro jednoznačnou identifikaci každého uživatele a je

celočíselného typu, neprázdný (NOT NULL) a při naplnění tabulky záznamem je automaticky inkrementován. Následují další sloupce, jejichž datovými typy jsou řetězce proměnné délky (varchar) za nimiž je v závorce uvedena jejich délka. Na předposledním řádku je uveden primární klíč, kterým je sloupec „ID_cislo“.

Ukázka zdrojového kódu pro vytvoření tabulky Kategorie:

```
create table Kategorie ( ID_cislo SMALLINT UNSIGNED NOT NULL
AUTO_INCREMENT,
                        Nazev varchar(60) NOT NULL,
                        PRIMARY KEY (ID_cislo)
                        ) TYPE=INNODB;
```

Stručný popis:

V tabulce „Kategorie“ budou uloženy záznamy s názvy kategorie, do které je jednotlivý projekt zařazen. Tyto záznamy budou v této tabulce tři: 1) IP síť, 2) Bezdrátové IP síť (WiFi, WiMAX, ad-hoc, bluetooth ...), 3) Bezdrátové senzorové síť.

Ukázka zdrojového kódu pro vytvoření tabulky Projekt:

```
create table Projekt (ID_cislo SMALLINT UNSIGNED NOT NULL
AUTO_INCREMENT,
                        Nazev varchar(50),
                        FileName_abstract varchar(20),
                        Datum char(14) NOT NULL, /* je ve formátu
YYYYMMDDHHmmss */
                        Aktivita BOOL NOT NULL,
                        Student SMALLINT UNSIGNED NOT NULL ,
                        Kategorie SMALLINT UNSIGNED NOT NULL ,
```

```

INDEX par_Student (Student),
INDEX par_Kategorie (Kategorie),
PRIMARY KEY (ID_cislo),
FOREIGN KEY (Student) REFERENCES Student(ID_cislo) ON
UPDATE CASCADE ON DELETE CASCADE,
FOREIGN KEY (Kategorie) REFERENCES Kategorie(ID_cislo)
ON UPDATE CASCADE ON DELETE CASCADE

) TYPE=INNODB;

```

Stručný popis:

Ze zdrojového kódu pro vytvoření tabulky „Projekt“ lze vidět kromě sloupce jednoznačně identifikující projekt také další sloupce. Sloupec „Nazev“ je řetězec proměnné délky s maximální délkou 50 znaků, ve sloupci „FileName_abstract“ jsou uloženy jména souborů, ve kterých je uložen abstrakt projektu, sloupec „Datum“ označuje datum vložení projektu a je v něm uložena časová hodnota ve formátu rok, měsíc, den, hodina, minuta a sekunda. Další položkou je „Aktivita“, která určuje jakýsi stav schválení projektu administrátorem. Data uložena v tomto sloupci nabývají buď hodnoty *true* nebo *false*, v závislosti na tom, zda je projekt schválen administrátorem či nikoliv. Položky „Student“ a „Kategorie“ jsou cizími klíči odkazujícími se na stejnojmenné tabulky.

Ukázka zdrojového kódu pro vytvoření tabulky Obrazek:

```

create table Obrazek (
ID_cislo SMALLINT UNSIGNED NOT NULL
AUTO_INCREMENT,
FileName varchar(20),
Projekt SMALLINT UNSIGNED NOT NULL ,

INDEX par_Projekt (Projekt),
PRIMARY KEY (ID_cislo),

```

```
FOREIGN KEY (Projekt) REFERENCES Projekt(ID_cislo) ON  
UPDATE CASCADE ON DELETE CASCADE ) TYPE=INNODB;
```

Stručný popis:

Tabulka „Obrzek“ slouží k uchování informace o názvu souboru s obrázkem pro příslušný projekt. Sloupec FileName určuje jméno souboru a sloupec Projekt je cizím klíčem tabulky „Projekt“ a jednoznačně identifikuje, o který projekt se jedná.

Ukázka zdrojového kódu pro vytvoření tabulky Zdrojovy kod:

```
create table Zdrojovy_kod (  
                ID_cislo SMALLINT UNSIGNED NOT NULL  
                AUTO_INCREMENT,  
                FileName varchar(20),  
                Projekt SMALLINT UNSIGNED NOT NULL ,  
  
                INDEX par_Projekt (Projekt),  
                                PRIMARY KEY (ID_cislo),  
                FOREIGN KEY (Projekt) REFERENCES Projekt(ID_cislo) ON  
UPDATE CASCADE ON DELETE CASCADE ) TYPE=INNODB;
```

Stručný popis:

Tabulka „Zdrojovy_kod“ slouží k uchování informace o názvu souboru se zdrojovým kódem pro příslušný projekt. Sloupec FileName určuje jméno souboru a sloupec Projekt je cizím klíčem tabulky „Projekt“ a jednoznačně identifikuje, ke kterému projektu se zdrojový kód vztahuje.

Ukázka zdrojového kódu pro vytvoření tabulky Dokumentace:

```
create table Dokumentace (  

```

```

        ID_cislo SMALLINT UNSIGNED NOT NULL
AUTO_INCREMENT,
        FileName varchar(50),
        Projekt SMALLINT UNSIGNED NOT NULL ,

        INDEX par_Projekt (Projekt),
                                PRIMARY KEY (ID_cislo),
        FOREIGN KEY (Projekt) REFERENCES Projekt(ID_cislo) ON
UPDATE CASCADE ON DELETE CASCADE    ) TYPE=INNODB;

```

Stručný popis:

Tabulka „Dokumentace“ obsahuje informace o názvu souboru s dokumentací pro příslušný projekt. Sloupec FileName určuje jméno souboru a sloupec Projekt je cizím klíčem tabulky „Projekt“ a jednoznačně identifikuje, ke kterému projektu se zdrojový kód vztahuje.

Závěr

V této závěrečné zprávě jsou uvedeny informace k méj diplomové práci „Webová aplikace pro výuku simulace v NS2“. Řešení bylo realizováno podle pokynů pro vypracování. Aplikace je po nainstalování funkční a její zdrojové kódy jsou uloženy na přiloženém CD.

V kapitole 2 je pojednáno o nástroji Network Simulator 2. Pomocí něj se realizují simulace sítí a o nich pak jejich autor vkládá údaje do webové aplikace. Zaregistrovaný návštěvník webu má možnost vložit do aplikace projekt. Projekt obsahuje informaci o jím vytvořené simulaci v NS2. Webová aplikace umožňuje také zobrazit si detail libovolného vloženého projektu, který je schválen administrátorem. Dále může řadit projekty, vyhledávat v nich zadaný výraz, či připojovat své příspěvky v diskusním fóru. Správce může ve svém okně projekty odevzdané uživatelem schvalovat, tím je zpřístupnit ostatním, a také je odstraňovat z databáze.

Teorie o technologiích, které jsou využity při implementaci této aplikace, jsou také popsány v kapitole 3. Jedná se o webový server Apache, databázový server MySQL, a jazyk PHP. Dále jsou zmíněny informace o zabezpečení webových aplikací včetně příkladů možných útoků na aplikace a jejich databáze. V kapitole 5.2 je uveden návrh databáze, která tvoří jádro aplikace. Tento návrh je závislý na požadavcích na aplikaci. Kapitoly 6,7 a 8 dávají čtenáři ucelený obraz o funkčnosti aplikace, jsou uvedeny ukázky grafické podoby aplikace tak, jak je ve finálním stádiu implementována, a také ukázky zdrojových kódů pro vytváření databázových tabulek aplikace.

Seznam použité literatury

- [1] FALL, Kevin, VARDHAN, Kannan. *The ns Manual*, Berkley, 2003.
- [2] MOLNÁR, Karol. *Prezentace přednášek z předmětu Hardware počítačových sítí*. FEKT VUT Brno, 2007.
- [3] ZENDULKA, Jaroslav. *Prezentace přednášek z předmětu Databázové systémy*. FIT VUT Brno, 2004.
- [4] KACÁLEK, Jan, MALÝ, Jan. *Publikované internetové články Zabezpečení webových aplikací*. <<http://www.elektrorevue.cz>>.
- [5] MEHDI, Achour a kol. *PHP Documentation, 1997-2007*. <<http://www.php.net>>.
- [6] *Oficiální internetová stránka webového serveru Apache*. <<http://www.Apache.org>>.
- [7] *Oficiální internetová stránka databázového serveru MySQL*. <<http://www.mysql.com>>.

Abecední přehled použitých zkratk, veličin a symbolů

- NS2 – Network Simulator 2
- PHP – Hypertext Preprocessor
- HTML – Hyper Text Meta Language
- XSS – Cross Site Scripting
- SQL – Structured Query Language
- ER – Entity Relationship

Seznam příloh

- CD s elektronickou verzí závěrečné zprávy a webové aplikace
- Prohlášení o shodě listinné a elektronické formy práce
- Licenční smlouva

Vlastní přílohy

Příloha – Instalace webové aplikace

POSTUP INSTALACE WEBOVÉ APLIKACE PRO VÝUKU SIMULACÍ V NS2 – shodné s obsahem textového souboru ctime.txt v kořenovém adresáři na přiloženém CD

1. Verze, pod kterou byla aplikace testována:

Server WAMP 5 Version 1.7.2, který obsahuje:

Apache - verze 2.2.4 (Win32)

PHP - verze 5.2.3

MySQL - verze 5.0.41

2. Zkopírování tohoto adresáře:

Celý adresář, ve kterém se nachází tento soubor ctime.txt, včetně jeho podadresářů zkopírujte do libovolného adresáře na váš webový server.

3. Vytvoření databázových tabulek:

V programu mysql.exe se spustí tyto 2 skripty nacházející se v kořenovém adresáři aplikace:

vytvor_tabulky.sql

vytvor_tabulky1.sql

4. Editace souboru config.php nacházející se v kořenovém adresáři aplikace:

Zde se nastaví jméno databázového serveru, login a heslo do databáze, dále IP adresa serveru, a e-mailová adresa, ze které se bude odesílat autorům projektu informace o schválení

5. Diskusní fórum (volně šiřitelný PHP skript z <http://www.php.jonweb.cz>)

Instalace probíhá podle informací v souboru /forum/ctime.txt

6. Počítadlo přístupů (volně šiřitelný PHP skript z <http://www.php.jonweb.cz>)

Instalace probíhá podle informací v souboru /poc/ctime.txt

7. Spuštění skriptu Vloz_heslo_admin.php

Spustit tento soubor ve webovém prohlížeči na serveru.

Tím se vloží do databáze tyto údaje potřebné pro přihlášení administrátora:

e-mail: simek@feec.vutbr.cz

heslo: bbb

Tyto údaje slouží k prvotnímu přihlášení administrátora do systému.

Po přihlášení si je administrátor může v aplikaci změnit.