

Elektronický slovník ve Virtuálním prostředí

Bc. Martin Mlčák

Diplomová práce

2005

Zde patří oficiální zadání diplomové práce.

ABSTRAKT

Abstrakt česky

Diplomová práce nastiňuje využití počítačové techniky ve vyučování a učení, a zabývá se vytvořením elektronického slovníku ve virtuálním prostředí.

Slovník je vytvořen na téma datová bezpečnost a obsahuje přehled nejčastěji používaných pojmů v oblasti datové bezpečnosti, s adekvátním vysvětlením jednotlivých pojmů. Slovník je rozdělen do dvou částí a to pro registrované a neregistrované uživatele. Práce také obsahuje návrh a řešení celého slovníku včetně všech použitých skript a zdrojových kódů.

Abstrakt ve světovém jazyce

My extended essay outlines the advantages of using computer technology by school teaching and learning. The essay is engaged in creating Electronic Dictionary in virtual setting.

The topic of the dictionary is Data Safety and it includes the summary of the most often use terms in the area of Data Safety with the explanation of each of these term. The dictionary is divided into two parts. First part is for registered users and the second part for unregistered users. My essay contains also the proposal and solution of the whole dictionary inclusive of all used Scripts and sources codes.

Děkuji mému vedoucímu práce Ing. Markovi Čandíkovi , Ph.D. za odborné vedení a pomoc při řešení této diplomové práce.

OBSAH

OBSAH	5
ÚVOD.....	7
I. TEORETICKÁ ČÁST	9
1 INFORMAČNÍ A KOMUNIKAČNÍ TECHNOLOGIE VE VZDĚLÁVÁNÍ	10
1.1 DOSAVADNÍ VÝVOJ A VYUŽITÍ ICT	10
1.2 IMPLEMENTACE ICT VE VZDĚLÁVÁNÍ	11
1.2.1 ICT a školy	11
1.2.2 ICT a učitelé	12
1.2.3 ICT a žáci	13
1.3 MULTIMÉDIA	13
1.3.1 Základní typy multimediálních informací	14
1.4 INTERNET	15
1.4.1 Využití technologií při distančním vzdělávání	15
1.4.2 On-line vzdělávání	16
1.4.3 E-learning	18
1.4.4 Jiné možnosti využití Internetu	19
1.5 TVORBA INTERAKTIVNÍHO SLOVNÍKU	20
1.5.1 Historie a vývoj internetu	20
1.5.2 WWW, HTTP, HTML	22
1.5.3 WWW a skriptování	23
1.5.4 WWW servery	23
1.5.5 Skriptovací jazyky	26
1.5.6 Databázové systémy	29
II. PRAKTICKÁ ČÁST	31
2 ZÁKLADNÍ SYNTAXE	32
2.1 HTML	32
2.1.1 CSS	33
2.2 PHP	33
2.3 MYSQL	35
2.4 PHP A MYSQL PŘÍSTUP DO DATABÁZE	39
2.4.1 Přihlašování k databázi	39
2.4.2 Výpis údajů	39
2.4.3 Výpis základních funkcí pro práci s databázemi	40
2.4.4 SQL dotazy	41
2.4.5 Datové typy	42
2.5 STRUKTURA TABULEK	44
2.6 PŘÍSTUP DO APLIKACE	46
ZÁVĚR.....	53
SEZNAM POUŽITÉ LITERATURY.....	54

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	55
SEZNAM OBRÁZKŮ	56
SEZNAM TABULEK.....	57
SEZNAM PŘÍLOH.....	58

ÚVOD

Moderní informační a komunikační technologie stále hlouběji zasahují do života člověka. Jejich přítomnost mění tvář mnoha oborů lidské činnosti. Také vzdělávání nalezlo v těchto technologiích nové výukové prostředky, které učitel uplatňuje v klasické vyučovací hodině.

Počítače a internet, dokázaly přinést zcela nové metody a formy výuky. Je tedy zřejmé, že význam vzdělání a potřebné schopnosti absolventů škol se, díky vývoji informačních a komunikačních technologií (dále již jen ICT), zásadním způsobem mění. Technologie jsou schopny podporovat jakýkoli model výuky. Jsou jakýmsi zesilovačem, který umocňuje dobré i špatné výukové metody.

O využití počítačové techniky ve výuce informatiky a příbuzných oborů není třeba příliš hovořit. Bez ní by se totiž dané předměty jen stěží daly vyučovat.

Ale nasazení výpočetní techniky do jiných předmětů přináší několik pozitiv.

Jednak dochází k menší izolovanosti informatiky jako aplikačního předmětu od ostatních disciplín a tedy posiluje pozici předmětu informatika mezi ostatními vyučovacími předměty. Jednak větší využití počítačové učebny a zájem ze strany ostatních kolegů přispívá k posílení pozice výpočetní techniky ve škole.

Způsob využití školních počítačů závisí na softwaru a službách, které škola učitelům a studentům poskytuje. Naprostou samozřejmostí by dnes měly být textové editory, tabulkové kalkulátory, prezentační programy, různé encyklopedie a slovníky na CD-ROM i výukové programy, např. jazykové. Pokud jde o internet, je běžně k dispozici Web, na některých školách je studentům poskytována i služba elektronické pošty.

Využití výpočetní techniky je nezbytnou podmínkou pro řešení mnoha projektů. Zpracování textů, obrázků, číselných údajů a nakonec i samotná závěrečná prezentace výsledků se dnes bez počítače většinou neobejde. Internet navíc přináší takhle podporované výuce ještě další netušené možnosti. Není jen zdrojem dat a místem určeným pro prezentaci výsledků práce školy. Je možno s ním pracovat též jako s nástrojem komunikačním. Budeme-li chtít internet co nejvíce využívat v kontextu sociální podstaty vzdělávání, potřebujeme do práce zapojit jeho schopnost spojovat živé lidi. Jedná se o něco poněkud jiného, než je pouhé získávání informací.

Proto v mé práci provedu rešerši možnosti využití počítačové techniky ve vyučování a v učení. Kde se blíže podíváme na nejruznější možnosti využití ICT a následně se zaměřím na jednu z možností. A to je vytvoření elektronického slovníku ve virtuálním prostředí.

I. TEORETICKÁ ČÁST

1 INFORMAČNÍ A KOMUNIKAČNÍ TECHNOLOGIE VE VZDĚLÁVÁNÍ

Informační a komunikační technologie označují výpočtové a komunikační prostředky, které podporují vzdělávání sběrem, archivací a výměnou informací [11]. K tomuto účelu využívají technické prostředky masové komunikace, jakými jsou rozhlas, televize nebo video, ale především osobní počítače, jejich vstupní a výstupní zařízení, prostředky pro digitalizaci, snímání, měření a řízení, Internet a jeho služby, integrované edukační programy, prostředky pro videokonference, elektronickou poštu, prostředky automatizační techniky, jako např. automatické snímáče, záznamníky a zařízení pro automatické vyhodnocování údajů.

Multimedia využívají informační a komunikační technologie k šíření, zpracování, prezentaci a vyhodnocení informací. Základním charakteristickým prvkem multimedií je nutnost zapojení více smyslů při jejich vnímání uživatelem. Dále se vyznačují možnostmi interaktivního přístupu k informacím za pomoci osobního počítače a možnostmi libovolného (i nelineárního) přístupu k požadovaným informacím. Výhodou aplikací využívajících multimediální prvky a efekty je jejich rychlost a přehlednost, s jakou se uživatel dostává ke hledané informaci.

1.1 Dosavadní vývoj a využití ICT

Pokusy s využitím počítačů k výukovým účelům byly prováděny již od šedesátých let 20. st.. V té době byly počítače něčím docela novým, ještě neověřeným, a panovala všeobecná představa, že pomocí nich lze zautomatizovat téměř všechny lidské činnosti. V pedagogice navíc tehdy ještě celosvětově převládal behavioristický instruktivní model učení, jehož nedostižným cílem bylo prostřednictvím co možná nejdokonalejšího ovládání všech vjemů studujícího dosáhnout určitého jeho chování. V konečném důsledku pak určitých jeho vlastností. Proto byly první výukové počítačové programy koncipovány tak, že se snažily plně řídit to, co uživatel dělá.

Trend využívání technologií řídících učení převládal do konce 80. let. Díky tomu, že tento způsob výuky do značné míry potlačoval vlastní motivaci a zájem studujícího, vyznačoval se vývoj stále rafinovanějšími a komplikovanějšími pokusy o řízení, které by si uživatel neuvědomoval. Počítačové aplikace tohoto typu však nakonec narazily na bariéru v oblasti vývoje umělé inteligence. Zatím totiž není možné vytvořit takový počítačový

program, který by byl schopen reagovat na všechny podněty uživatele a usměrňovat ho určeným směrem. Jinými slovy, počítače dosud nejsou schopny učitele nahradit.

Přesto se velké množství aplikací fungujících na tomto principu v praxi běžně používá. Jsou to programy sloužící k drilování nebo testování. I ty mají své určité místo ve výuce. Některé znalosti nebo dovednosti si lze totiž nejlépe osvojit právě tímto způsobem (abeceda, malá násobilka, vyjmenovaná slova, nepravidelná slovesa apod.) Ne vždy je počítač pro výuku tohoto typu nezbytný. Na druhou stranu má i značné výhody v tom, že je neúnavný, spravedlivý apod.

Od počátku 90. let se začíná v oblasti vzdělávacích technologií prosazovat nový směr - konstruktivismus. Vychází z podpory vlastní aktivity mozku a snaží se tudíž ve vzdělávání vytvořit takové prostředí, které by tuto aktivitu podporovalo. Základním rozdílem v případě uplatnění vzdělávacích technologií je to, že aktivita a řízení přechází ze stroje na uživatele [15]. Podobný systém ovšem může mít úspěch pouze u studentů, jejichž přirozená chuť do práce není ničím narušena a které v této činnosti podporuje celé okolí včetně učitele.

Za konstruktivní lze označit všechny výukové aplikace, které podporují vlastní aktivitu studentů. Vlastně to jsou nakonec i všechny nástroje, pomocí nichž lze cokoli vytvořit, jako různé editory, autorské systémy, programovací jazyky, nástroje na modelování apod. Patří sem však i zdroje dat, v nichž může aktivní hledající student najít potřebné informace, jako např. encyklopedie, slovníky atd. Informace dnes bývají studentům výukovými aplikacemi předkládány nejčastěji formou hypertextu. To znamená, že data mají nelineární uspořádání podle souvislostí. Je to způsob, který se snaží napodobit uspořádání vědomostí v lidském mozku a je tudíž pro studium vhodnější než klasický lineární text.

1.2 Implementace ICT ve vzdělávání

1.2.1 ICT a školy

Je zřejmé, že je třeba docílit toho, aby ve školách bylo dostatek počítačů a aby byly připojeny do Internetu. Tuto problematiku je však třeba poněkud konkretizovat. Není totiž úplně jedno jaké počítače do škol dáme, kam je umístíme a jak je připojíme do sítě.

Školy často získávají starou vyřazenou techniku jiných organizací. Tato technika jistě může být ještě dobře využitelná. Starší počítače mohou sloužit k pořizování textových dat, k E-mailování, k výuce základů informatiky apod. Jen je třeba počítat s tím, že jsou více poruchové, vyžadují stálou péči technika a rostou náklady na opravy (pokud se opravují). Dnešní školy potřebují být vybaveny též současnými multimediálními počítači, na nichž lze pracovat s grafikou, využívat CD-ROM, prohlížet WWW stránky, učit se jazyky apod.

Dalším závažným problémem je rozmístění počítačů. Dnes je u nás zvykem budovat většinou tzv. počítačové učebny určené k výuce informatiky. Ty bývají často v odpoledních hodinách nevyužité a učitelé jiných předmětů se jim zdaleka vyhýbají. Je proto třeba charakter počítačových učeben změnit na studovny. Ideálním modelem by bylo udělat mimo výuku informatiky z počítačových učeben studovnu, která by se mohla stát součástí školní knihovny. K dispozici by zde mohly být jak klasické studijní materiály, tak moderní elektronické. Nezbytnou nutností je to, aby k počítačům měli přístup též učitelé. Proto by měli mít k dispozici počítače ve sborovně a v kabinetech. Konečným cílem by pak mělo být umístit alespoň jeden počítač do každé třídy.

Jak již bylo řečeno, školní počítače by měly být propojeny do sítě. Lokální síť značně usnadňuje jejich správu a zjednodušuje administrativu. Z výukového hlediska nás však zajímá hlavně připojení do Internetu. Není totiž vůbec jedno, jakým způsobem a hlavně rychlostí je škola připojena. Nejlevnější a nejpomalejší komutované připojení totiž dovoluje jen velmi omezenou práci s Internetem. Při vhodné volbě technického řešení mohou v tomto případě všichni uživatelé lokální sítě pracovat s elektronickou poštou. Prohlížení WWW stránek je však silně komplikováno a omezeno. Proto je třeba školy do Internetu připojovat co nejrychlejší, pokud možno pevnou linkou. Pak je možno bez problémů pracovat nejen s poštou a WWW, ale škola navíc může provozovat své vlastní servery, a sama informace do Internetu poskytovat. Většina vyspělých zemí má zvláštní dohody s telekomunikačními firmami poskytujícími připojení do Internetu o mimořádných minimalizovaných cenách pro školy.

1.2.2 ICT a učitelé

Nejdůležitější podmínkou úspěšného zavádění ICT do vzdělávání je schopnost učitelů tyto nové nástroje vhodným způsobem využívat. Současná situace v této oblasti je

špatná. Myšlení učitelů i způsob jejich přípravy je většinou natolik konzervativní a rezistentní vůči změnám, že lze jednoznačně říci, že právě toto bude úplně nejtěžším úkolem celé reformy.

1.2.3 ICT a žáci

Je všeobecně známo, že děti mají mnohem bližší vztah k technologiím než dospělí. Proto se učitelé velmi často dostávají do situace, kdy mají žáci nad nimi v technických záležitostech převahu. Úloha učitele však je a vždy bude nezastupitelná. Místo toho, aby své žáky řídil, měl by se stát spíše jakýmsi poradcem, průvodcem a povzbuzovatelem k práci. Učení se totiž bez práce neobejde.

Značně důležitou problematikou je možnost zneužívání Internetu k šíření pro děti nevhodných materiálů nebo k pronikání do chráněných oblastí sítě. Takovému využívání ICT na školách je třeba zabránit. Není to však snadné. Dalším opatřením, které je možno zavést, je filtrování dat, které z Internetu do školy přicházejí. Technicky je to řešitelné podobně jako ochrana proti počítačovým virům [1]. Poněkud problematický je však celkový přínos filtrování. Může se totiž snadno stát, že nebudou přístupné i některé materiály, které jsou zcela nezávadné. Navíc může tento postup naopak u žáků vzbudit o zakázané materiály nežádoucí zájem. Nejlepším, i když asi zatím v některých případech poněkud nedostižným, cílem je vychovat děti tak, aby byly kompetentní se samy rozhodnout, co budou na Internetu dělat. Měly by mít stále dostatek konstruktivní práce, aby neměly vůbec potřebu se zabývat nevhodnými činnostmi.

1.3 Multimédia

Multimediální technologie představují souhrn postupů a prostředků pro zpracování, archivaci a přenos multimediálních informací [13]. Multimediální technologie je možno posuzovat z hlediska multimediálních technologií pro zpracování a archivaci multimediálních informací (tzv. informační technologie), multimediálních síťových a

přenosových technologií pro přenos multimediálních informací (tzv. přenosové technologie).

Na základě uvedeného posouzení se rozlišují :

- multimediální informační technologie
- multimediální přenosové technologie

Multimediální informační technologie *realizují některé operace s multimediálními informacemi* [14], a to především zpracování multimediální informace, archivaci multimediální informace, která reprezentuje realizaci základních procesů spojených s budováním multimediálních databází a realizací multimediálních serverů. Významnými subkategoriemi multimediálních informačních technologií pro archivaci multimediálních informací jsou:

- databáze
- prostředky pro vyhledávání informací
- multimediální servery

Multimediální přenosové technologie zahrnují dvě hlavní kategorie technologií, kterými jsou [15]:

- telekomunikační technologie - všechny druhy přenosových a multiplexních technologií (ATM, optických, mobilních, atd.) spolu s přístupovými technologiemi v inteligentních sítích
- síťové technologie - integrující zejména problematiku multimediálních protokolů a jejich konverze, protokolů pro rychlé (high-speed) sítě a manažování multimediálních sítí

1.3.1 Základní typy multimediálních informací

Pod pojmem multimediální informace obecně rozumíme informace zprostředkované pomocí minimálně dvou různých druhů digitálních signálů, které se současně vyskytují v daném souboru. Mezi základní typy multimediálních informací, které jsou v telekomunikačních sítích distribuovány patří:

- textové a grafické informace

- víceúrovňové statické obrazy
- pohyblivé dynamické obrazy (video)
- řečové a audio informace

Multimediální služby a multimediální technologie přinášejí do prostředí informační techniky množství nových aspektů.

1.4 Internet

Kapitolou samou pro sebe je Internet. Díky němu přibýly k původně jen informačním technologiím do označení ICT též technologie komunikační. Umožňuje nejen snadnou komunikaci mezi všemi uživateli pomocí E-mailu, ale je též zdrojem nepřeborného množství informací prostřednictvím služby World Wide Web. Až do roku 1994 byl Internet sítí čistě akademickou. Po tomto roce byl uvolněn i ke komerčním účelům a od té doby je jeho rozvoj doslova překotný. Jeho místo ve výuce je nezastupitelné. V souladu s principy moderní pedagogiky dovoluje do výuky zapojit lidi a informační zdroje doslova z celého světa. Web je totiž jednou obrovskou celosvětovou hypertextovou encyklopedií, kde lze najít téměř vše. Problémem je to, že ne vždy se podaří vhodné informace rychle najít, ne všechny jsou pravdivé a některé jsou dokonce pro výuku vysloveně nevhodné. Obrovskou výhodou je naopak poměrně snadná **možnost publikování**, kterou mají téměř všichni uživatelé Internetu. Je tedy zřejmé, že **konstruktivní výukové možnosti** jeho využití jsou obrovské.

Internet je z pohledu on-line médií klíčovou technologií. Informace uložené na internetu (na počítači který je konfigurovaný jako tzv. server, který je připojený do internetové sítě), si může prohlédnout kterýkoliv jiný uživatel Internetu. Výhodami internetového publikování jsou především kontinuálnost poskytování informací (zveřejňování informací bez časových /redakčních uzavírek, neomezený rozsah zveřejňovaných informací), využívání hypertextu a možnosti využití multimédií při zveřejňování informací.

1.4.1 Využití technologií při distančním vzdělávání

Nové informační technologie přinesly nové široké možnosti ve vícero vědních disciplínách, včetně pedagogiky. Tento přínos je viditelný především v distančních vzdělávacích technologiích, kde nové informační a komunikační technologie sehrávají

významnou roli, a to jak při tvorbě a používání distančních vzdělávacích materiálů, při vzájemné komunikaci mezi studenty, při komunikaci mezi zprostředkovateli distančního studia a studenty, při organizování a administrování distančního studia, apod. Informační a komunikační technologie nacházejí uplatnění nejen v oblasti distančního vzdělávání, ale jsou také vhodným doplňkem kombinované výuky, resp. dálkového studia.

Výukový model založený na nových informačních a komunikačních technologiích – on-line výuky, je možné ve stručnosti charakterizovat skutečností, že nemá prostorové a časové omezení. V porovnání s konvenčním, klasickým přístupem vzdělávání nabízí výukový model založený na nových informačních a komunikačních technologiích studentům i pedagogům vícero možností – podstatně větší množství informací, možnosti sdílení informací vícerym studentům současně, nové možnosti spolupráce učitelů a studentů, apod. Na druhé straně si použití nových informačních a komunikačních technologiích vyžaduje zvýšené požadavky na vzdělávání ze strany studentů, jakými jsou např. přístup k počítači a k Internetu, základní praktické zkušenosti s informačními technologiemi, silnější motivaci studenta úspěšně ukončit on-line vzdělávání, apod.

1.4.2 On-line vzdělávání

On-line vzdělávání představuje takový druh vzdělávání, který pro komunikaci mezi učitelem a žákem, resp. mezi žáky navzájem, využívá také prostředky počítačových a telekomunikačních technologií, které umožňují komunikaci v reálném čase.

Mezi základní výhody on-line vzdělávání patří:

prostorová decentralizace studia -možnost studovat z kteréhokoliv místa na světě, kde je vyhovující připojení k Internetu a dostatečné technické vybavení počítačem ke studiu.

časová nezávislost studia -on-line vzdělávací kurzy jsou uloženy na výukovém serveru, který umožňuje téměř nepřetržitý přístup k vzdělávacím materiálům, pokud je tak nakonfigurovaný¹.

¹ Některé servery mohou být nakonfigurované tak, že přístup k vzdělávacím materiálům je možný jen v definovaném čase, aby se zabránilo přetěžování komunikačních linek, resp. mimo tohoto definovaného časového intervalu nemusí být zabezpečená dostatečná kvalita přístupu ke vzdělávacím materiálům.

možnost studia vlastním tempem - možnost volnějšího časového limitu k nastudování dané problematiky, v rámci definovaných možností.

možnosti anonymity - eliminace možných diskriminujících faktorů během studia, jakými jsou věk, vzhled, rasa, apod.

přístup ke zdrojům - možnost jednoduché technické realizace tvorby odkazů na aktuální zdroje elektronických informací a materiály související s obsahovou náplní kurzu tak, aby byl studentům umožněn hlubší a rozsáhlejší rozsah edukačních materiálů.

výhody on-line komunikací - on-line vzdělávání umožňuje různé způsoby vzájemné komunikace, které jsou založené na využití některých základních služeb počítačových sítí, kterými jsou služby zprostředkovávající diskuzi ve skupinách - tzv. služby dávkového charakteru a služby zprostředkovávající diskusi ve skupinách - tzv. služby interaktivního charakteru.

komplexní podpora samostatného studia - on-line edukační materiály mají poskytovat komplexní pedagogickou a technickou podporu samostatného studia, která zahrnuje speciální vypracované studijní materiály, aktivační prvky studia, koncepční a organizační informace o studiu, manuály pro technickou podporu on-line vzdělávání, návody na způsob on-line studia, apod.

Vzhledem k výhodám, které on-line vzdělávání nabízí, disponuje on-line vzdělávání jistými nevýhodami, kterými z pohledu studentů jsou :

odlišný přístup k informačním technologiím a technické problémy (nestejné možnosti připojení studentů k Internetu – různá rychlost připojení a tím časově omezená možnost přístupu k informacím v případě pomalých přenosových linek, finanční náklady spojené s připojením k Internetu, různé technické vybavení počítačů u studentů, a především různé hardverové komponenty multimediálních sestav počítačů).

nedostatek schopnosti samostatného studia – On-line vzdělávání nabízí relativně velkou míru „volnosti“ během studia, co může u studentů s nižší motivační úrovní, resp. s nižší sebedisciplínou a zodpovědností k studiu vést k jistým problémům.

Typické aktivity studentů v rámci virtuálního on-line studia na Internetu jsou:

- samostatné studium s využitím Internetu a multimédií
- navazování kontaktů s konzultantem v interaktivním režimu

- poznávání nástrojů komunikací
- vzájemné poznání studentů na dálku v rámci virtuálních týmů
- individuální přístup při hledání a poznávání informačních zdrojů
- týmový "brainstorming" s využitím moderních síťových technologií
- koordinace práce ve skupině, konzultace s instruktorem na dálku
- příprava studentské prezentace vlastního projektu a zpracování ohlasu

1.4.3 E-learning

E-learning je moderní prostředek vzdělávání, využívající výhody on-line komunikace (on-line vzdělávání), který navíc obsahuje prostředky pro řízení studia (samostudia). Právě vlastnost řízení studia je základní vlastností, odlišující on-line studium od e-learningu.

Základními vlastnostmi e-learningu jsou :

E-learning je ve svojí podstatě multimediální interaktivní vzdělávání nové generace, praktikované pomocí počítačových kurzů, které jsou distribuované pomocí CD-ROMu, Internetu, Intranetu a pod...

Celý proces vzdělávání je tu integrovaný do písemného a mluveného slova, audio a video sekvencí, schématu, grafiky, a taktéž do různých testů.

Tento způsob vzdělávání zvyšuje efektivitu procesu učení. Student nové informace přijímá především prostřednictvím sluchových a vizuálních vjemů. Účinnost se dále zvyšuje aktivní účastí studenta na studijním programu - plněním konkrétních praktických úloh a testů.

Rozmanitost použitých prostředků a jejich grafická přitažlivost, společně s možnostmi vícecestné komunikace, zvyšuje efektivitu výuky.

Rozvíjí se nové možnosti pro zdokonalení výukového procesu:

Internet a multimédia nabízí široké spektrum znalostí a zkušeností,

světovým trendem je používání moderních komunikačních technologií ve vzdělávání, což vede k rychlému se rozvíjení zkušeností s virtuálním on-line studiem.

1.4.4 Jiné možnosti využití Internetu

Velmi zajímavé možnosti poskytuje další schopnost internetu, a to možnost vzájemné komunikace všech uživatelů. Hlavní službou v této oblasti je elektronická pošta neboli e-mail, který patří k základním funkcím počítačových sítí již od jejich zrodu.

Přístup k této službě je normálně vázán na získání konta (práva přístupu) na některém počítači do internetu připojeném. Spolu s kontem bývá uživateli totiž většinou zřízena též schránka elektronické pošty a získá tak i vlastní e-mailovou adresu.

Technickou stránkou elektronické pošty se zde zabývat nebudeme. My se zaměříme spíše na to, jak tento nástroj využít ve vzdělávání. Vhodné využití elektronické pošty je totiž jedním z možných způsobů, jak do spolupráce s děním ve třídě zapojit mimoškolní prostředí. Takto lze zvýšit motivaci studentů a zajistit kontakt s jinými lidmi, kteří mohou být zdrojem dalšího poučení. Mohou to být třeba špičkoví vědci - odborníci na právě probírané téma i vrstevníci z kteréhokoli kouta světa.

E-mail má proti Webu jednu velkou výhodu. Je totiž použitelný i v případě technicky nedokonalého připojení do internetu. Teoreticky nebrání nic tomu dopisy psát i číst ve stavu, kdy připojení k síti neexistuje. V okamžiku, kdy se později toto spojení naváže, může dojít k velmi rychlé výměně pošty a spojení se může zase zrušit.

Elektronická pošta (spolu s SMS) se pomalu stává samozřejmostí. Je to nejlevnější způsob komunikace. Mnoho studentů by si dnes již bez ní vůbec neumělo představit kontakt se svou rodinou a známými (v době kdy jsou na internátech / kolejích).

Přínos internetu pro učitele je neoddiskutovatelný. Každý učitel může jistě vyhledávat nové zajímavé informace, jimiž může obohacovat svou práci. Může udržovat kontakt s kolegy nebo se rovnou přihlásit do nějakých konferencí a číst denně třeba i několik desítek zpráv, z nichž některé budou možná i dost zajímavé.

1.5 Tvorba interaktivního slovníku

1.5.1 Historie a vývoj internetu

Již začátkem 60. let v době studené války se začaly ve Spojených státech amerických objevovat myšlenky na vytvoření sítě, která by propojovala nejdůležitější vojenské, vládní a akademické počítače a která by byla schopna přežít jaderný úder. Bylo jasné, že by to měla být síť bez centra, a měla by být schopna funkce i v případě výpadku jednotlivých uzlů.

Zrod sítě ARPANET

Koncem 60. let se sdružení RAND a univerzity MIT a UCLA zabývaly myšlenkou vytvořením decentralizované sítě s přepojováním paketů 1. V roce 1968 přispěchal s finanční podporou na pomoc Pentagon, respektive jeho agentura DARPA (Defence Advanced Research Project Agency) a na podzim roku 1969 byly zprovozněny první čtyři uzly sítě, která byla nazývána ARPANET (všechny čtyři uzly byly umístěny na amerických univerzitách). V roce 1971 se ARPANET rozrostl na 15 uzlů a v roce 1972 již bylo propojeno celkem 37 počítačů. Je s podivem, že se hlavním využitím sítě ARPANET nestalo používání vzdálených počítačů, ale komunikace prostřednictvím elektronické pošty a elektronické konference. Kromě konferencí sloužících k výměně vědeckých informací se objevily první konference určené pro zábavu. První z nich konference SF-LOVERS určená milovníkům science-fiction [11].

První neamerické Instituce na síti

Rok 1973 se zapsal do dějin Sítě připojením prvních dvou neamerických institucí, britské University College of London a norské Royal Radar Establishment. V té době již bylo jasné, že myšlenka decentralizované sítě s přepojováním paketů je tím správným řešením pro síť, která by měla být robustní a snadno rozšířitelná. Již v roce 1974 byla zveřejněna první specifikace protokolu TCP/IP, jenž měl nahradit stávající přenosový protokol NCP. V roce 1983 se oficiálně přešlo od protokolu NCP k sadě protokolů TCP/IP.

Prudký rozvoj Internetu

Období mezi lety 1983-1992 by bylo možné označit jako druhou etapu rozvoje Internetu. Tato etapa byla charakterizována prudkým růstem Internetu (z přibližně tisíce počítačů v roce 1983 na více než milion počítačů v roce 1992) a především expanzí mimo americký kontinent. V prvních letech tohoto období vznikly počítačové sítě EUNET, EARN, japonská síť JUNET a britská síť JANET. Velkým krokem kupředu bylo vytvoření americké páteční sítě NSFNET (ta měla v roce 1986 na dnešní poměry směšnou kapacitu 56 kb/s), která propojovala pět nejdůležitějších amerických "superpočítačových" center. Její vznik a provoz byly financovány vládní agenturou NSF.

Předvedení Gopheru a Veroniky

Roku 1991 dosáhla NSFNET kapacity 44,7 Mb/s a vědci z Minnesotské univerzity poprvé předvedli systém Gopher, který byl posledním schůdkem pro přechod k systému WWW. Pro prohledání gopherského prostoru byl v roce 1992 na univerzitě v Nevadě vytvořen systém Veronika

U kolébky WWW

Již počátkem roku 1989 se na půdě ústavu částicové fyziky CERN objevil dokument HyperText and CERN, jenž popisoval možnosti vytvoření interního distribuovaného systému jako jednotné nadstavby nad mnoha různorodými informačními zdroji. Autorem tohoto dokumentu byl Tim Berners-Lee, který později v listopadu roku 1990 předvedl první prototyp WWW serveru (jako operační systém byl zvolen NeXT). Prudký nárůst obliby WWW se datuje k září 1993, kdy byla dostupná první funkční verze velmi populárního prohlížeče NCSA Mosaic.

Nástup komerce

Do roku 1993 zůstával Internet doménou především vědeckých a akademických pracovišť. Uživatelé a provozovatelé se vehementně bránili příchodu komerčních aktivit na Internet. Situace se začala měnit v roce 1991, kdy americký kongres přijal zákon High Performance Computing Act (jeho iniciátorem byl senátor Al Gore). Od roku 1993 se na Internetu začaly ve velkém objevovat komerční organizace, nejprve počítačové, později i

firmy z dalších oborů lidské společnosti. V mnoha státech se Internet stává běžnou součástí každodenního života.

Kdo řídí Internet

Je překvapující, že Internet navzdory počátečnímu financování agenturami DARPA a NSF nikdo nevlastní a ani není centrální autorita, která by jej nějakým způsobem řídila. Přesto existují instituce podílející se významnou měrou na fungování a dalším rozvoji Internetu. Jako první jmenuje Internet Society (ISOC), jenž sdružuje internetové uživatele. ISOC má dvě hlavní složky Internet Activities Board (IAB) a Internet Engineering Task Force (IETF). Obě tyto složky spolupracují s nejvýznamnějšími počítačovými firmami na tvorbě standardů potřebných pro další rozvoj Internetu.

Další institucí, která od poloviny roku 1994 dbá zejména na rozvoj služby WWW, je WWW Consortium (W3C). Sdružuje lidi, kteří se podíleli v ústavu CERN na prvních krůčcích "děťátka jménem WWW", techniky z MIT a z francouzského institutu INRIA. Ředitelem konsorcia není nikdo jiný než tvůrce WWW Tim Berners-Lee. Na webovém serveru W3C můžete nalézt spoustu zajímavých materiálů o jeho činnosti a internetových standardech, jež spoluvytvářejí. Tento server najdete na adrese: <http://www.w3c.org/>

V neposlední řadě musím jmenovat organizaci InterNIC dohlížející na přidělování tzv. IP adres počítačům a doménových jmen.

Budoucnost Internetu

Rozvoj Internetu se samozřejmě nezastavil. Zejména v zemích "Internetově nevyspělých" dochází ke stejně prudkému rozvoji jako v počátcích devadesátých let v Americe. Průzkumy ukazují, že počet počítačů připojených do Internetu se každým rokem zdvojnásobuje. Trend růstu počtu připojených do roku 2000 zachycuje graf. Je z něj patrné, že nás v roce 2000 bylo téměř hrozné číslo, reprezentující přes 100 milionů připojených počítačů.[11]

1.5.2 WWW, HTTP, HTML

Služba WWW (World Wide Web zkráceně web) vznikla zhruba v roce 1990. Jedná se o síťový systém pracující jako klient-server. Byla vyvinuta ve Švýcarském CERNu poblíž Ženevy. A právě v době vzniku WWW je vyvinut jazyk HTML, který je i dnes

ústřední technologií Webu. Druhou nezbytnou technologií je přenosový protokol http. Ten zajišťuje přenos HTML stránek z WWW serveru do prohlížeče. Třetí nezbytnou součástí k přenosu je URL . V překladu to znamená něco jako „jednotný vyhledávací prostředek“. Každý objekt umístěný na Webu má svoji jedinečnou URL adresu, která slouží k vytváření odkazů na daný objekt. V praxi jde o to, že uživatel vyšle URL(požadavek) na WWW server, ten jej zpracuje, vyhledá dokument s tímto URL odkazem a odpoví ve formě HTML dokumentu. O celou tuto cestu se postará přenosový protokol HTTP.

1.5.3 WWW a skriptování

HTML, HTTP a URL sice představuje silný nástroj Internetu, ale z pohledu dnešního uživatele neumožňuje nějaký velký komfort. Uživatel mohl pouze pasivně prohlížet WWW stránky. K tomu, aby Web byl interaktivní bylo nutné využít možnosti automatického generování těchto stránek. HTML stránka je vlastně soubor uložený na disku (na WWW serveru) a my můžeme vznést URL dotaz na jakýkoliv spustitelný soubor. Ten je generován WWW serverem. K tomu, aby se tak dělo, bylo zapotřebí vytvořit nějaké rozhraní, které by se staralo o přenášení dat mezi WWW serverem a programem. Toto rozhraní se jmenuje CGI (Common Gateway Interface).A jedná se o tzv. CGI skripty.CGI skripty se nejčastěji píší v Perlu, příkazovém Shellu UNIXU, C, C++, Javě, ASP, PHP....

1.5.4 WWW servery

WWW servery jsou důležitým prvkem pro zobrazování internetových stránek a komunikaci s uživatelem. WEB server je program, který dostane požadavek na určitou HTML stránku nebo nějaký soubor, server požadavek analyzuje a odešle jako výsledek v určitém formátu (HTML stránka, obrázek, soubor ke stažení...). Komunikace probíhá pomocí http protokolu. WWW servery mohou zpracovávat i virtuální účty, zpracovávat skripty na „straně serveru“, omezovat počet spojení, zabezpečovat přístup k určitým adresářům atd.

Apache

Mezi velmi používané a populární WWW Servery patří Apache. Je to bezesporu dáno svou univerzálností, stabilitou, funkčností, možností rozšíření.... Apache se přímo podílel na vzniku a vývoji některých jazyků (např. PHP), které jsou ve svém prvním počátku

vyladěny hlavně pro tento server. Historie tohoto serveru sahá do roku 1994, kdy skupinka programátorů a vývojářů v NCSA začala pracovat na jednoduchém a volně šiřitelném webserveru. Jejich snaha byla úspěšná a tak v dubnu 1995 vyšla na svět první verze serveru Apache. Server byl a je i dnes volně šiřitelný a je distribuován ve formě zdrojových kódů. Původní platformou byl UNIX, ale „portováním“ na další platformy na sebe nenechalo dlouho čekat. Dnes si z adresy <http://www.apache.org> můžete stáhnout jak již zmíněné zdrojové kódy, tak i distribuční balík pro MS Windows či NLM moduly pro Netware.

Geniální myšlenkou vývojářů tohoto serveru byla možnost zavést do paměti pouze jádro Apache a poté provést zavedení pouze těch dalších přídatných modulů rozšiřujících jeho funkce, které jsou potřeba. Tím je umožněno v podstatě neomezené rozšiřování funkcí serveru, aniž by bylo nutné znovu a znovu instalovat a konfigurovat nové verze serveru. Stačí spustit potřebný modul a hotovo. Programátoři znalí C++ si dokonce můžou napsat své vlastní přídatné moduly, dokumentace je volně dostupná na již zmiňované adrese.

Na menší problém narazí správci „klikáři“, kteří jsou zvyklí naklikat konfiguraci serveru v okenních manažerech a wizardech. Bohužel ,ale nic takového Apache nemá. Jediné místo (až na výjimky), které ukrývá konfiguraci serveru, je prostý textový soubor HTTPD.CONF, který lze editovat v Notepadu.

Samozřejmostí je podpora PHP, Perlu, ASP (jako externí modul) a dalších funkcí, které jsou od dobrého serveru požadovány.

Novell Enterprise Web Server

Firma Novell se při vyvíjení (či spíše nevyvíjení) webserveru vydala zvláštní cestou. Koupila produkt firmy Netscape nazvaný Netscape Server for Netware a dodávala jej spolu s Novell Netware 4.11 a vyšším. Na svou dobu to byl výborný a spolehlivý server. Jeho vývoj nebyl tak bouřlivý jako vývoj internetu a server začal zaostávat za svými konkurenty. Dnes je dodáván s Novell Netware 5.x a volitelně i s Novell Netware 6 jako Novell Enterprise Web Server 4.5 (i když jeho jádro se stále hlásí jako Netscape 3.5 for Netware).

Výbornou vlastností serveru je jeho úzká provázanost s NDS, takže přístup k jednotlivým částem webu lze nastavit jednoduchým přidáním či odebráním přístupových

práv na adresáře či soubory. Není třeba udržovat žádnou další databázi uživatelů a hesel, i když takovou možnost server také nabízí.

Konfigurace a správa webserveru se provádí na první pohled možná trochu těžkopádně přes webovské rozhraní (a pouze přes něj, neexistuje žádná další utilita), které zdaleka není intuitivní a může nezalého administrátora spíše zmást. Omezené jsou možnosti konfigurace domovských adresářů jednotlivých uživatelů, kde nelze nastavit privilegovaný přístup. Také správa aliasů k adresářům je na první pohled těžkopádná a nepřehledná a server jako takový má i další drobné vady na kráse.

Samozřejmostí je podpora SSL, ASP, NSP, JSP, JavaBeans, JavaServletů a Perlu. V balíku také najdete news-server a FTP server. Výbornou vlastností serveru je zakomponovaný modul pro vyhledávání na webu, jenž může správce jednoduše aktivovat, aniž by byl nucen psát složité prohledávací skripty.

Server nepodporuje používání dalších přídatných modulů, čímž ztrácí možnost pružně reagovat na překotný vývoj webu. Podle dostupných informací je další vývoj tohoto serveru zastaven. V Novell Netware 6 je již plně podporován server Apache, server Novell Enterprise Web Server je dodáván pouze jako volitelný a doplňkový modul.

Internet Information Server

Drtivá obchodní politika firmy Microsoft způsobila, že tento webserver se po nesmělých začátcích dostal mezi velmi populární. Třebaže jeho první dvě verze byly v podstatě nepoužitelné, (IIS 2.0 se dodával s MS Windows NT 4.0, až teprve SP3 ho povýšil na verzi s pořadovým číslem 3), postupem času se propracoval mezi nejpoužívanější servery a následoval Option Pack 4 ve kterém byl obsažen IIS verze 4.0. Díky tomu, že je dodáván jako součást operačního systému, je téměř dokonale provázán se systémem, na kterém běží, a dokáže využívat všechny jeho vlastnosti.

Konfiguraci serveru lze provádět přes Microsoft Management Console nebo přes webové rozhraní. Dokumentace a návody, jak spravovat server, je i na českém trhu dost a dost.

Možnosti tohoto serveru jsou opravdu rozsáhlé. Kromě již standardních a každým předpokládaných služeb, jimiž je podpora ASP, SSL, Perlu, PHP a dalších je integrován také jednoduchý SMTP server. IIS podporuje tzv. ISAPI filtry, které, jak již sám název napovídá, fitrují obsah dat přenášných k uživateli či od uživatele a provádějí určitou akci nebo data modifikují. Příkladem ISAPI filtrů jsou například velmi známá serverová

rozšíření programu FrontPage. K dispozici je také obsáhlá dokumentace pro vývojáře, kteří potřebují napsat svůj vlastní ISAPI filtr vyhovující právě jejich potřebám.

Velká popularita s sebou nese nepříjemnou stránku, kterou je nutnost neustále bdít nad aktualizací především bezpečnostních "záplat".

PinkNetWeb Server

Pochází z dílny českých vývojářů a více informací o něm naleznete na <http://pnws.pinknet.cz>. Server podporuje PHP, Perl, SSL, ISAPI (nikoli ASP). Má v sobě zabudovaný SMTP server, podporuje POP3, je také FTP serverem a PROXY serverem. Jak vidíte, k dokonalosti mu schází opravdu málo.

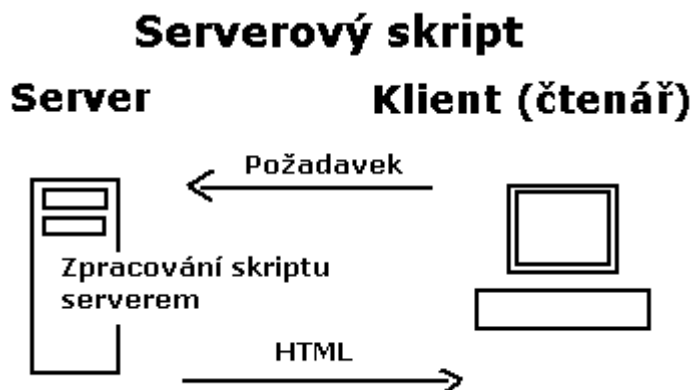
Další typy WWW serveru jsou : PNWS, Boa, Thttpd, Abyss web server, Gaudium.

1.5.5 Skriptovací jazyky

PHP

Vývoj PHP začal v roce 1994. Jeho autorem a jedním ze současných vývojářů je Rasmus Lerdorf. Základy PHP jsou v jazyku C a Perlu, proto jsou si taky velmi podobné. V současnosti je nejrozšířenější verze PHP 4.0. Ale vzhledem k tomu že už je k dispozici delší dobu verze PHP 5.0, dá se očekávat přechod na tuto novější verzi.

PHP je skriptovací jazyk vykonávaný na straně serveru vkládaný do běžného HTML kódu. Co to znamená? Každou stránku, která obsahuje PHP skripty, server nejprve vezme a vykoná všechny příkazy v PHP, které jsou ve stránce uvedené, poté pošle klientovi již čistý HTML kód, který je výsledkem běhu skriptu. Server může PHP skripty teoreticky hledat ve všech odesílaných souborech, ale zpravidla je nakonfigurován tak, aby je hledal v souborech s příponami .php, .php3 nebo .phtml. Příkazy PHP jsou vkládány přímo do HTML kódu a jsou od něj odděleny tagy <? a ?> (nebo <?php a ?>).



Obr. 1. Ukázka zpracování PHP skriptu.

Do PHP je momentálně zabudována podpora těchto databází: Adabas D, dBase, Direkt MS-SQL, Express, FilePro (jen pro čtení), FrontBase, HYperware, IBM DB2, Informix, My SQL, m SQL, Oracle....

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). Bezproblémové je použití PHP se servery Apache, Microsoft Internet Information Server, Personal Web Server, Oreilly Website Pro Server, Gaudium, Xitami[7,21,27]

JavaScript a JScript

JavaScript vyvinula společnost Netscape. Microsoft vytvořil JScript, který je svou syntaxí téměř shodný s JavaScript. Jde o jednoduchý objektově orientovaný skriptovací jazyk. Oproti PHP však pracuje „na straně klienta“. Je totiž součástí internetových prohlížečů. Ke zpracování tedy nedochází na serveru, nýbrž v prohlížeči uživatele. JavaScript je vhodný zejména pro obsluhu událostí (např. ověření akce záznamu, kontrola vyplnění formuláře...). Nevýhodou může být, že v některých prohlížečích se nemusí zobrazovat a že každý, kdo si takové stránky prohlíží, má přístup ke kompletnímu zdrojovému kódu.[8]

ASP

Aktive Server Pages (původním kódovým označením *Denali*) je technologie firmy Microsoft pro vytváření aplikací běžících „na straně“ webových serverů. Nejde o programovací jazyk, ale o sadu objektů Windows NT, která zprostředkovává jednoduchý přístup k http požadavkům klientů a dalším objektům v rámci webového serveru. Těmito objekty se manipuluje pomocí skriptovacího jazyka. Mezi nejvíce používané skriptovací jazyky pro vytváření ASP stránek patří:

- VBScript – jazyk na základě Microsoft Visual Basicu (nejčastěji)
- JScript – skriptovací jazyk velmi podobný JavaScriptu
- PerlScript – skriptovací verze Perlu od společnosti ActiveState

K provozu potřebujete platformu Windows NT s webovým serverem Microsoft Internet Information Server (IIS). Windows 9x jsou podporovány pouze pro účely testování a nehodí se pro běžný provoz.[2]

Java

Java je programovací jazyk pocházející od firmy Sun Microsystems. Jedná se o objektově orientovaný jazyk vycházející z C++, ke kterému má syntaxi velmi blízko. Velkou výhodou Javy je její hardwarová nezávislost, neboť je překládána do speciálního mezikódu (bytecode), který je na konkrétním počítači nebo zařízení (PC, handheld, mobilní telefon apod.) interpretován do nativního kódu (tzv. JIT – Just-In-Time compilerem). Programátor tedy může napsat Java program například na PC pod Windows a spustit ho na PC s Linuxem, Applem, SGI, DEC (tam, kde je k dispozici Java runtime).[6]

Visual Studio (Net.)

Jedná se o velmi mocný celek, který je momentálně pouze na platformě Microsoft Windows. Umožňuje programování internetových dynamických stránek, ale i lokálních či síťových programů. Visual Studio podporuje .NET Framework (základní jádro pro tvorbu a provoz webových aplikací a služeb). V rámci prostředí Visual Studio.NET lze používat mnoho programátorských jazyků (Basic, C#, Visual C++ a Visual J#). Skripty se vykonávají na „straně serveru“.

1.5.6 Databázové systémy

MySQL

Systém vznikl ve švédské firmě MySQL AB. Mezi hlavní vlastnosti patří vysoká rychlost, spolehlivost a výkonný relační databázový systém, který umožňuje zpracování velkého množství dat. MySQL je open-source projekt, který je šířen pro nekomerční použití zdarma. V případě komerčního využití je nutno zaplatit určitou částku (cca od 200\$ do 500 \$) [9]. Tím získá uživatel možnost zásahu do zdrojových kódů, modifikovat ho a následně si zkompilovat vlastní distribuci. Tímto způsobem lze získat distribuci, která bude maximálně vyhovovat vlastním požadavkům, a navíc lze navýšit výkon až o 40% .

Výhodou je nezávislost na platformě. Podle oficiální dokumentace je možné databázi provozovat na operačních systémech AIX, Amiga, BSDI, Caldera, DEC, FreeBSD, Linux, HP-UX, Mac OS X, NetBSD, Novel, OpenBSD, MS Windows.... Nevýhoda je téměř nulová kontrola složitějších integritních omezení. Vazby mezi tabulkami systém nehlídá a programátor si musí dát pozor, aby vztahy nebyly porušeny.

MSSQL

Jde o SQL Server společnosti Microsoft. Obsahuje v sobě grafické prostředí a spoustu integrovaných snadně konfigurovatelných správců i pro vytváření reportů. Jeho nevýhodou je cena a vyšší hardwarová náročnost. Cena se pohybuje v desítkách až stovkách tisíc korun. Stejně tak je možné provozovat MS SQL Server pouze pod operačním systémem MS Windows.

PostgreSQL

Je objektově-relační systém řízení báze dat, založený na projektu POSTGRES verze 4.2. Berkeleyské katedry počítačových věd Kalifornské univerzity v USA. Má obsáhlou podporu SQL konstrukcí. PostgreSQL obsahuje i transakce a vnořené SELECTy, je možnost si definovat vlastní funkce a trigger. K zajištění integrity dat si můžete vytvořit různé integritní omezení včetně cizích klíčů. Zamykání tabulek nahradil systém spravující více verzí dat. Další vlastností je možnost dědičnosti definice tabulek nebo vytvářet vlastní datové typy.

Oracle

Je největším komerčním konkurentem firmy Microsoft. Jde o velmi výkonný, škálovatelný a bezpečný databázový systém. Nová verze (Oracle 10g) umožňuje tzv. grid computing. Jde o spojení (clustering) více serverů, které se potom jeví jako jeden velký počítač a dynamicky rozděluje výpočetní výkon podle aktuální potřeby jednotlivých aplikací. Obsahuje Oracle Enterprise Manager, který by měl umožňovat snadnou správu databází. Tak jako MySQL nebo PostgreSQL je ho možné provozovat na více systémech (Windows NT/2000/XP, Linux, Solária, HP-UX, Tru64, AIX).

II. PRAKTICKÁ ČÁST

2 ZÁKLADNÍ SYNTAXE

2.1 HTML

Základní struktura HTML dokumentu je následující:

```
<html>
  <head>
    <title>Nadpis stranky</title>
  </head>
  <body>
    <center>Text který se zobrazí v <b>prohlížeči</b>.</center>
  </body>
</html>
```

Html - Začíná a končí celý dokument. Veškerý další obsah musí být uvnitř. Jedná se o značku nepovinnou, většina prohlížečů si ji domyslí. Pokud ale chcete mít soubory v souladu s normou, <html> tag používejte. Tag html nemá žádné atributy.

Head - Hlavička dokumentu, která se nezobrazuje. Obsahuje nepovinně další tagy (title, meta, link, style, script aj.). Pokud v hlavičce použijete prostý text, v některých prohlížečích se zobrazí na začátku stránky. Tag head nemá žádné atributy.

Title - Titulek stránky. Párový tag. Mezi značkami <title> a </title> je text, který bude chápán jako titulek. Titulek má několik málo známých funkcí, například:

zobrazuje se v záhlaví okna a na hlavním panelu jako nadpis aplikace

zobrazuje se jako nadpis ve vyhledávacích

je prioritní při indexování stránky vyhledávacími roboty

mnoho softwaru ho používá jako nativní odkaz

Body - Tělo dokumentu. Obsahuje veškerý zobrazovaný obsah stránky.

2.1.1 CSS

V roce 1996 došlo k rozšíření jazyka HTML o funkci CSS (Cascading Style Sheets). Tato technologie umožňuje přiřadit jednotlivým HTML příkazům řadu charakteristik stylu textu (například písmo, barva, řádkování, styl tabulek v dokumentu, pozadí stránek a podobně). Díky této funkci lze vytvářet HTML dokumenty, které budou vzhledově konzistentní. [5]

2.2 PHP

PHP lze vkládat přímo do HTML stránek. Jsou podporovány tři druhy syntaxe:

```
<? ... ?>
<?php ... ?>
<SCRIPT language="php"> ... </SCRIPT>
```

Ukázkou použití PHP skriptu v HTML stránce je následující příklad:

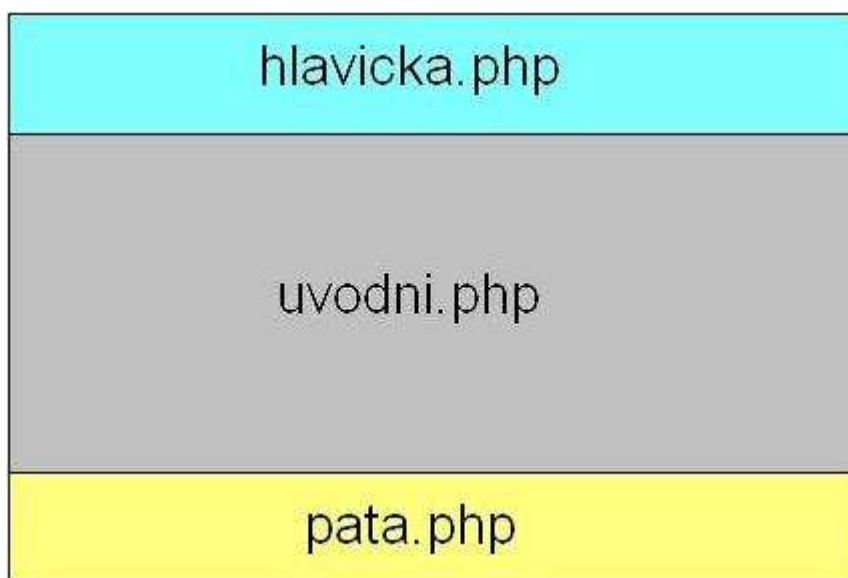
```
<html>
<head>
<title>Ukázka</title>
</head>
<body bgcolor="#FFFFFF" text="#000000">
<center><font face="Arial CE, Arial" size="5">
Aktuální čas: <?php echo Date ("H:i:s"); ?>
</font></center>
</body>
</html>
```

A jak skript pracuje? Důležité je si všimnout, že jde v podstatě o klasickou HTML stránku, která navíc obsahuje jeden PHP příkaz, a to `echo Date ("H:i:s")`, který je od okolního HTML kódu oddělen značkami `<?php` a `?>`. Server nejprve vezme požadovaný soubor `ukazka.php`, vidí, že má příponu `.php`, a proto ho nejprve „prožene“ interpretem PHP a vykoná všechny příkazy - ty hledá právě mezi oddělovacími značkami `<?php` a `?>`. Až narazí na příkaz `echo`, který slouží k zápisu do výsledného souboru.

Funkce `Date` vrací datum a čas, v závorce jsou její parametry, které v tomto případě určují formát, ve kterém má být čas zobrazen (hodiny, minuty a sekundy oddělené

dvojtečkami). Následující středník slouží k oddělení více příkazů od sebe - v tomto případě je nadbytečný, protože příkaz je jen jeden, ale je dobré zvyknout si ho psát. Interpret PHP tedy namísto příkazu PHP zapíše do stránky aktuální čas a server poté stránku odešle návštěvníkovi. Pokud si dáte zobrazit zdroj výsledku skriptu, opravdu uvidíte pouze čisté HTML, žádné PHP.

Pomocí skriptování je také možné vytvářet strukturu dokumentu. Dříve byl programátor odkázán pouze na používání rámců v HTML nebo musel na každé stránce doplňovat celý HTML kód (což bylo neefektivní). Díky PHP si můžeme opakující se části nadefinovat do externího souboru a ty volat ve všech následujících stránkách. Toho jsem využil i v případě této aplikace (Obr.2).



Obr. 2. Struktura stránek .

Při snaze nahrát skript index.php dojde k vložení do stránky nejdříve souboru hlavicka.php (modrá část). Poté dojde k vložení uvodni.php (šedá část) a na závěr se vloží pata.php (žlutá část). Samozřejmostí je, pokud index.php obsahuje ještě nějaké další kódy, ať už HTML nebo php, tak se provedou v závislosti na umístění ve zdrojovém kódu index.php .

Načtení jednotlivých zdrojových kódů obstarává příkaz :

```
<? readfile ('hlavicka.php')?>
```

... text ...

```
<?readfile ('hlavicka.php')?>
```

V souboru hlavicka.php se nachází hlavní ovládací menu (Obr.3). Strukturu dokumentu uzavírá soubor pata.php (Obr.4).



Obr. 3. Hlavní menu dokumentu



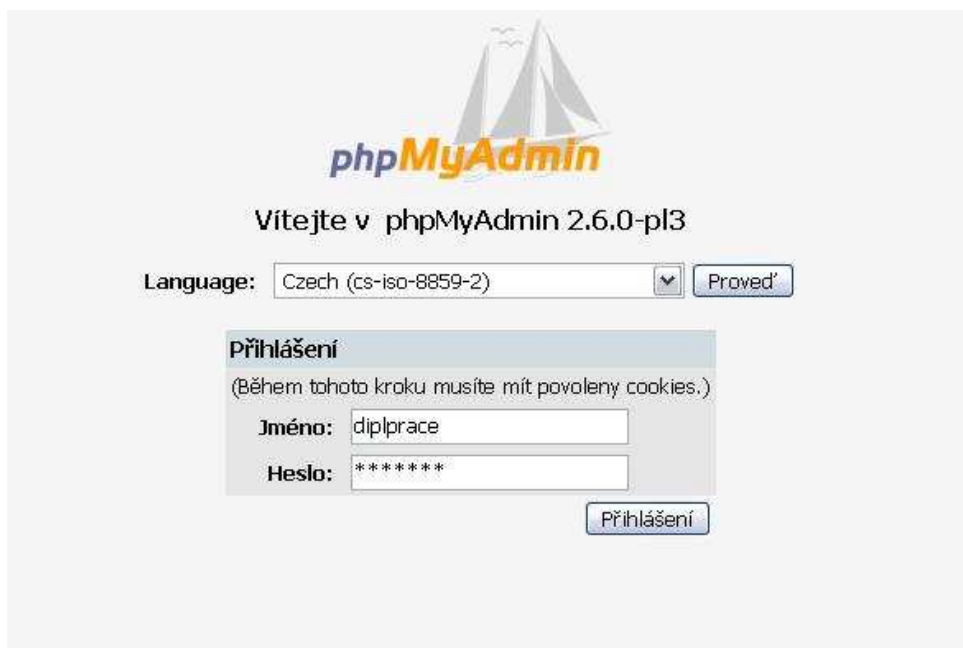
Obr. 4. Spodní menu dokumentu.

2.3 MySQL

V nových verzích se již objevila možnost využití transakcí a zamykání tabulek. Jedná se o možnost vložení příkazů do bloku transakce. Tyto příkazy se pak promítnou do databáze jediné tehdy, když všechny příkazy v transakci budou úspěšně provedeny. Pro zajištění bezpečnosti má v sobě MySQL zabudovaný systém uživatelů a práv. Přístup k databázím je chráněn hesly, přičemž každý uživatel může mít definovanou množinu operací, které je oprávněn vykonávat (tyto se definují v databázi mysql). Pro zvýšení ochrany se hesla přenášejí výhradně v šifrovaném tvaru. Od verze MySQL 3.23 je možné vytvářet tabulky o velikosti až 8 miliónu terabajtů (2^{63} bajtu).

Pokud je databázový server nastartovaný, můžeme k němu přistupovat. MySQL má v sobě implementovaný nástroj na její správu z příkazové řádky (MS Windows) nebo terminálu (Unix, Linux).

V našem případě jsem využil web hosting na serveru webzdarma.cz, kde je podpora jak MySQL tak PHP. Přístup do databáze je v tomto případě jednoduchý přes webové rozhraní na adrese <http://mysql.webzdarma.cz>. (Obr.5)



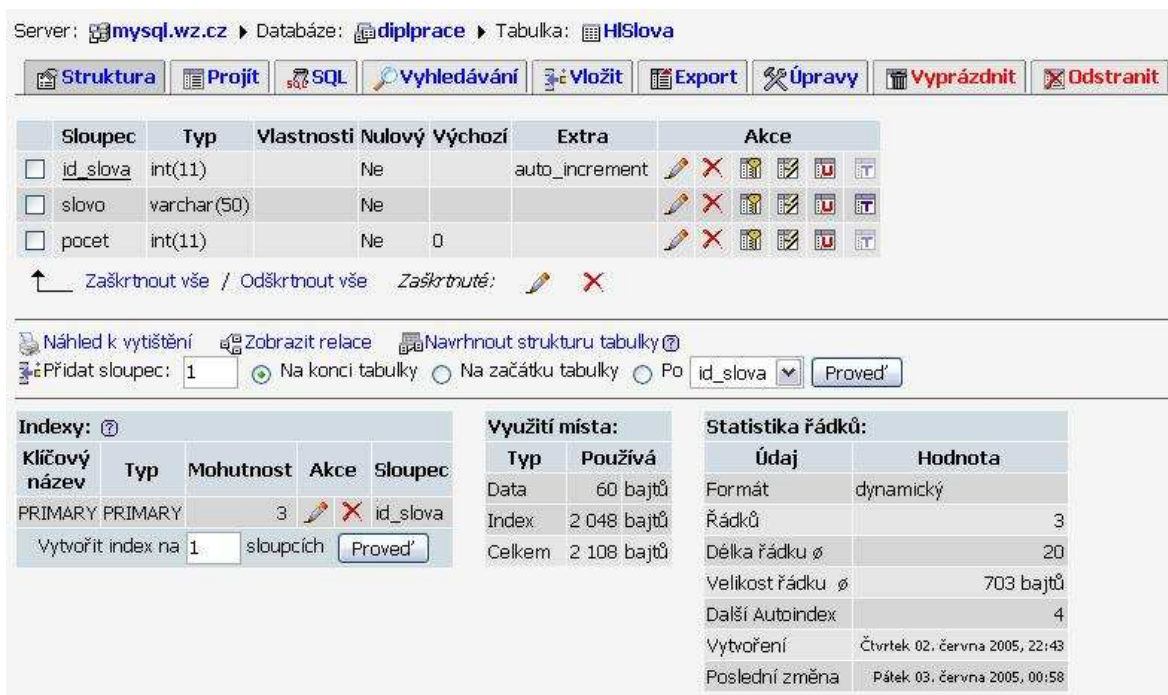
Obr. 5. Přihlášení na MySQL server.

kde po zadání jména a hesla se dostanete do webového rozhraní (Obr.6), kde můžeme provádět správu databáze dvěma způsoby:

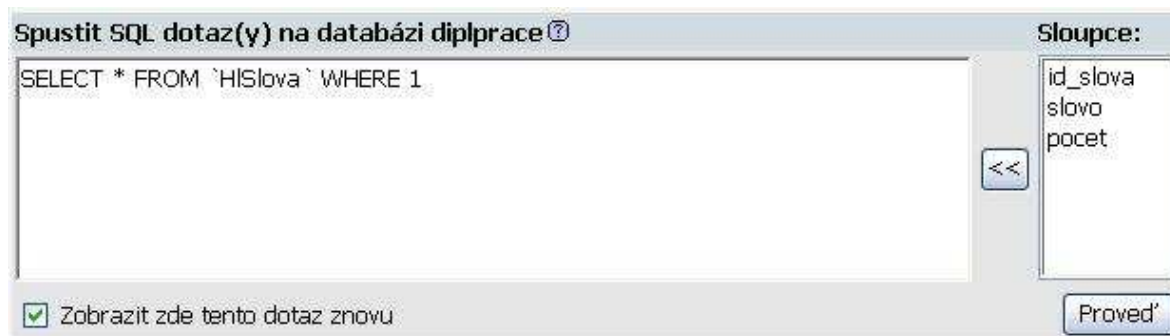
- Pomocí jednoduchého webového rozhraní (Obr.7).
- Klasickými příkazy jazyka MySQL (Obr.8).



Obr. 6. webového rozhraní pro správu databáze.



Obr. 7. jednoduchého webového rozhraní.



Obr. 8. Menu pro příkazy jazyka MySQL.

Tvorba a editace tabulek pomocí webového rozhraní je jednoduchá a intuitivní. Proto se zaměřím na tvorbu a editaci pomocí příkazů MySQL.

Nyní ukáži, jak vytvořit tabulku, kterou budeme pro diplomovou práci potřebovat. Vzhledem k podobné tvorbě všech tabulek, uvedu tvorbu jen jedné tabulky. Tedy příkaz pro vytvoření tabulky bude vypadat takto:

```
CREATE TABLE Slova(
id_slova int NOT NULL AUTO_INCREMENT,
slovo varchar(50) NOT NULL,
url varchar(100) NOT NULL,
nazev varchar(25) NOT NULL,
popis text NOT NULL,
PRIMARY KEY (id_slova)
);
```

Tento kód vytvoří tabulku *Slova* s pěti sloupci *id_slova*, *slovo*, *url*, *nazev*, *popis*. Dále je ze zápisu zřejmé že, položka *id_slova* bude typu int, nesmí nabývat nulové hodnoty a bude se automaticky s každým novým údajem zvětšovat o jedničku. A v posledním řádku říkáme že *id_slova* bude tzv.primární klíč (což je jedinečná hodnota).

2.4 PHP a MySQL přístup do databáze

2.4.1 Přihlašování k databázi

Před zahájením práce s databází pomocí PHP skriptů je nutné se k databázi přihlásit. K tomu slouží funkce *mysql_connect* nebo *mysql_pconnect*. První funkce se přihlásí k databázi a po změně stránky dojde ke zrušení spojení. Tak je potřeba se připojovat na každé stránce znova. V druhém případě otevřené spojení s databází permanentní. Při připojování k databázi je nutné ještě zadat název *mysql_select_db*.

```
<?
$localhost="server";
$username="uzivatel";
$userpass="heslo";
$databasename="dabaze";

mysql_connect("$localhost","$username","$userpass");
mysql_select_db("$databasename");
?>
```

V mé aplikaci je přihlašování prováděno z externího souboru, který si načítá každý skript. Pokud nedojde k úspěšnému přihlášení do databáze či k serveru, stránka se nenačte a zobrazí se chybové hlášení.

2.4.2 Výpis údajů

Po úspěšném přihlášení k databázovému serveru a databázi, již můžeme využívat ostatní funkce. Výpis údajů lze provádět dvěma způsoby. V praktických příkladech ukáži oba dva:

```
<?
$sql = " SELECT  id, slovo FROM tabulka";
$dotaz = mysql_query ($sql); //vykonání dotazu
while ($pole = mysql_fetch_assoc ($dotaz)) {
    echo $pole["id"]; echo $pole["slovo"];}
mysql_free_result($result);
?>
```

Nebo

```
<?
$sql = " SELECT  id, slovo FROM tabulka";
$dotaz = mysql_query ($sql); //vykonání dotazu
for ($i = 0; $i<mysql_Num_Rows ($dotaz); $i++) {
    echo mysql_result ($dotaz,$i,'id' );
    echo mysql_result ($dotaz,$i,'slovo' );
}?>
```

Využití závisí od způsobu výpisu. Pokud např. chcete vypisovat tabulku s velkým počtem záznamů, je vhodnější použít první způsob (pomocí cyklu while a s využitím pole). Vykonání by mělo být rychlejší. Je možné také použít funkce *mysql_fetch_array* nebo *mysql_fetch_row*. Druhý způsob se hodí např. pro výpis jen některých údajů, ale od tohoto způsobu se již upouští.

2.4.3 Výpis základních funkcí pro práci s databázemi

<i>mysql_affected_rows</i>	- Vrátí počet změněných záznamů v MySQL po posledním dotazu.
<i>mysql_close</i>	- Ukončí spojení s MySQL Serverem.
<i>mysql_connect</i>	- Vytvoří spojení s databází.
<i>mysql_create_db</i>	- Vytvoří MySQL databázi.
<i>mysql_db_query</i>	- Proveďte MySQL dotaz.
<i>mysql_error</i>	- Vrátí text chybové zprávy předchozího MySQL příkazu.
<i>mysql_fetch_array</i>	- Načte výsledný řádek do asociativního, čísel.pole nebo obojí.
<i>mysql_fetch_assoc</i>	- Načte výsledný řádek do asociativního pole.
<i>mysql_row</i>	- Načte následný řádek do pole.
<i>mysql_free_result</i>	- Uvolní výsledek z paměti.
<i>mysql_list_dbs</i>	- Načte všechny databáze dostupné na MySQL serveru.
<i>mysql_list_fields</i>	- Načte výsledek s obsahem položky.

mysql_list_tables	- Načte všechny tabulky v MySQL databázi.
mysql_num_fields	- Vrátí počet položek ve výsledku.
mysql_num_rows	- Vrátí počet záznamů ve výsledku.
mysql_pconnect	- Otevře persistentní spojení s MySQL serverem.
mysql_query	- Pošle MySQL dotaz.
mysql_result	- Načte výslednou hodnotu jedné položky.
mysql_select_db	- Vybere MySQL databázi.
mysql_tablename	- Načte jméno tabulky.

2.4.4 SQL dotazy

Jazyk SQL vznikl v 70.letech z výzkumného projektu o relačních databázích firmou IBM. Byl přijat jako oficiální standart pro relační databáze. Používá se pro čtení, vkládání, aktualizaci a odstraňování dat. Nejvíce používané příkazy jsou následující:

SELECT – slouží jako výběrový dotaz do databáze

příklady:

```
SELECT * FROM Slova;
```

Vybere všechny záznamy z tabulky slova.

```
SELECT MAX(id)FROM Slova;
```

Vybere největší identifikační číslo (id) z tabulky slova.

```
SELECT * FROM Uzivatele WHERE jmeno 'Martin'
```

Vypíše všechny položky, kde je ve sloupci jméno položka Martin.

INSERT – přidání řádku do tabulky

příklady:

```
INSERT INTO Slova VALUES ('$jmeno','$url','$jmeno','$popis')
```

Vloží údaj od tabulky Slova, tento SQL dotaz je nutné použít v rámci PHP dotazu do databáze, který namísto '\$jmeno','\$url','\$jmeno', '\$popis' nahradí požadované údaje z vyplněného formuláře při vložení nových údajů do databáze.

UPDATE – modifikace hodnoty v tabulce

příklady:

```
UPDATE article SET title = '$title', message = '$article_text' WHERE id_article = $id_article"
```

Zjistí jaké má uživatel práva, a podle toho dosadí hodnotu do '\$title' podle toho mu dovolí / nedovolí číst zprávy.

DELETE – vymazání řádků z tabulky

příklad:

```
DELETE FROM $tabulka WHERE id = $id
```

Záznam z číslem \$id bude z tabulky \$tabulka smazán.

2.4.5 Datové typy

Jako všechny databázové systémy i MySQL nabízí ve standardní výbavě několik datových typů. Různé verze mohou obsahovat více či méně možností, proto uvedu jen nejdůležitější z nich.

Tab. 1. Číselné datové typy.

Typ	Velikost
TINYINT	1 byte
SMALLINT	2 bytes
MEDIUMINT	3 bytes
INT	4 bytes
INTEGER	4 bytes

BEGINT	8 bytes
FLOAT	4 bytes
DOUBLE	8 bytes
DOUBLE PRECISION	8 bytes
REAL	8 bytes

Tab. 2. Časové datové typy.

Typ	Velikost
DATE	3 bytes
DATETIME	8 bytes
TIMESTAMP	4 bytes
TIME	3 bytes
YEAR	1 byte

Tab. 3. Číselné datové typy.

Typ	Počet znaků
CHAR(M)	Max. velikost 255
VARCHAR(M)	Proměnlivý CHAR, max. velikost 255
TINYBLOB, TINYTEXT	2^8
BLOB, TEXT	2^{16}
MEDIUMBLOB, MEDIUMTEXT	2^{24}
LOB, LONGTEXT	2^{32}

2.5 Struktura tabulek

Article - tabulka slouží pro ukládání poznámek.

```
CREATE TABLE `article` (
  `id_article` int(11) NOT NULL auto_increment,
  `title` varchar(100),
  `message` text,
  `publish_date` int(11),
  `uzivatele` int(11),
  `archive_date` int(11),
  PRIMARY KEY (`id_article`)
);
```

Hlslova – zde se ukládají údaje o počtu kolikrát bylo které slovo vyhledáváno

```
CREATE TABLE HlSlova(
  `id_slova` int NOT NULL AUTO_INCREMENT,
  `slovo` varchar(50) NOT NULL,
  `pocet` int(11) NOT NULL,
  PRIMARY KEY(id_slova)
);
```

Online – do tabulky se ukládají informace o aktuálních přihlášených uživateli

```
CREATE TABLE `online` (
  `uz_id` int(11) NOT NULL default '0',
  `cas` int(14) NOT NULL default '0',
  sent_on TIMESTAMP,
  PRIMARY KEY (`uz_id`)
) TYPE=MyISAM;
```

Slova – zde se ukládají jednotlivé slova a jejich významy

```
CREATE TABLE `Slova` (
  `id_slova` int NOT NULL AUTO_INCREMENT,
  `slovo` varchar(50) NOT NULL,
  `url` varchar(100) NOT NULL,
  `nazev` varchar(25) NOT NULL,
```

```
`popis` text NOT NULL,  
PRIMARY KEY (`id_slova` )  
);
```

Uzivatele – slouží pro ukládání údajů o registrovaných uživateli

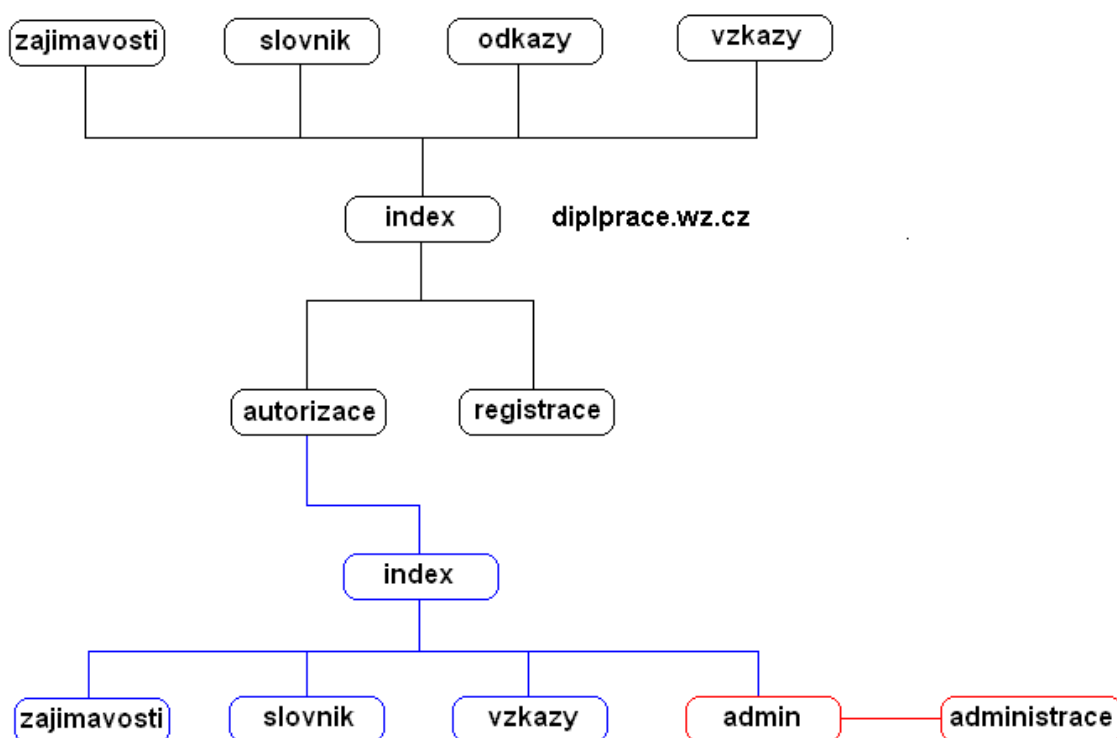
```
CREATE TABLE `uzivatele` (  
  `id` int(11) NOT NULL auto_increment,  
  `jmeno` varchar(30) NOT NULL default "",  
  `celejmeno` varchar(50) NOT NULL default "",  
  `heslo` varchar(32) default NULL,  
  `prava` tinyint(2) NOT NULL default '0',  
  `email` varchar(50) NOT NULL default "",  
  UNIQUE KEY `jmeno` (`jmeno`),  
  KEY `id` (`id`)  
) TYPE=MyISAM AUTO_INCREMENT=5 ;
```

Kniha – slouží pro ukládání odkazů z neregistrované části

```
CREATE TABLE `kniha` (  
  `id` int(11) NOT NULL auto_increment,  
  `jmeno` varchar(150) NOT NULL default "",  
  `datum` varchar(150) NOT NULL default "",  
  `prispevek` text NOT NULL,  
  `mejl` varchar(150) default NULL,  
  PRIMARY KEY (`id`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

2.6 Přístup do aplikace

Ještě než se pustíme do popisu celé aplikace je nutné si ukázat strukturu celé aplikace (Obr.9). Černá barva označuje část aplikaci po které se můžou pohybovat neregistrovaní uživatelé. Modrá barva znázorňuje možnosti pohybu po webu pro registrované uživatele. No a červená barva ukazuje části které jsou přístupné pouze administrátorovi.



Obr. 9. Struktura systému z hlediska uživatelů.

Ted' už můžeme popsat celý web. Po zobrazení webové stránky na adrese diplprace.wz.cz se nám objeví úvodní stránka (Obr.10). Zde se můžeme volně pohybovat pomocí menu.

Máme zde k dispozici odkazy „ZAJIMAVOSTI“ kde jsou odkazy na zajímavé články. Další možností je „SLOVNÍK“ kde se můžeme pustit rovnou do vyhledávání položek v databázi. Samozřejmě je tato část vyhledávání podstatně jednodušší oproti vyhledávání ve slovníku v registrované části.



Obr. 10. úvodní stránka.

V položce „ODKAZY“ jsou uvedeny některé zajímavé stránky, zabývající se tematikou datové bezpečnosti, virů... Jako poslední v menu je položka „VZKAZY“ kde může nezaregistrovaný uživatel zanechat jednoduchou zprávu.

V levé části webové stránky se nám nabízejí dvě možnosti. Buď se zaregistrovat, a nebo se přihlásit do systému. Pokud se chceme přihlásit, slouží nám k tomu formulář, kde zadáme svůj login a své heslo (které jsme si vytvořili při registraci).

Pro registraci slouží tlačítko registrace. Registrační část je tvořena třemi kroky, kdy v prvním kroku (Obr.11) zadáme „login“ pod kterým se budeme přihlašovat do systému, a pak zadáme svůj email, na který nám přijde potvrzení „loginu“, a zároveň i vygenerovaný registrační kód.

Ve druhém kroku (Obr.12) budeme vyzváni k zadání registračního kódu, přístupového hesla a svého jména a příjmení. Pokud vše proběhne správně, tak ve třetím kroku bude sděleno že registrace proběhla úspěšně, a systém odešle na uživatelský email zprávu o ukončení registrace a uživatelský login a heslo.

1. krok v registraci

Pro úspěšné zaregistrování zvolte uživatelské jméno a e-mail, kde Vám bude poslán ověřovací registrační kód.

login:

e-mail:

pokračuj

Obr. 11. první krok registrace.

2. krok

Login byl potvrzen. Nyní pokračujte zadáním registračního kódu a hesla.

Právě Vám byl zaslán e-mail s registračním kódem. Opište kód z e-mailu a potom si zvolte heslo, pro kontrolu zadejte heslo 2x.

registrační kód:

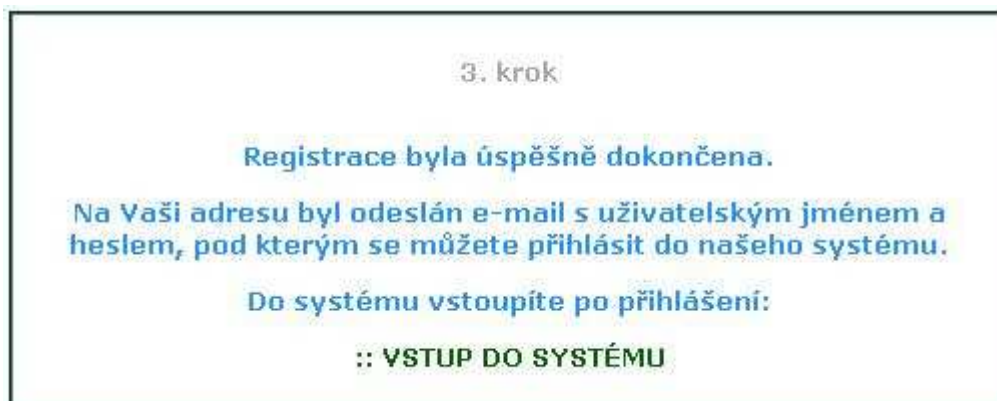
heslo:

heslo znovu:

jméno a příjmení:

pokračuj

Obr. 12. druhý krok registrace.



Obr. 13. třetí krok registrace.

Po přihlášení se dostaneme do registrované části (Obr.14) která je vizuálně velmi podobná uživatelské části. V levé prostřední části je zobrazeno jméno přihlášeného uživatele, a tlačítko s možností se odhlásit od systému. V pravé části je pak zobrazena informace o počtu on-line přihlášených uživatelů, včetně našeho loginu i s celým jménem. V případě dalších přihlášených uživatelů dojde k jejich zobrazení i jejich loginu a jména.



Obr. 14. Registrovaná část.

V horním menu se pak nacházejí odkazy na pohyb po registrované části. Položka „ZAJÍMAVOSTI“ zůstává prozatím stejná (do budoucna se předpokládá její rozšíření pro registrované členy).

Ale „SLOVNÍK“ už je v tomto případě obohacen o možnosti zobrazit celou tabulku, a tak si prohlédnout všechny výrazy které tabulka obsahuje (Obr.15). Samozřejmě zde zůstává i klasické vyhledávání, které je ale navíc vylepšeno i o zobrazením URL odkazu (pokud je vytvořen) na podrobnější popis hledaného slova.



Obr. 15. možnosti vyhledávání ve slovníku

Další položkou v menu jsou „VZKAZY“ (Obr.16), kde může každý uživatel zanechat svůj vzkaz, který může číst jen daný uživatel a samozřejmě administrátor. Vzkazy je možné editovat a také smazat. Vzkaz je opatřen datumem i časem vložení a zůstává uložen až do doby než jej administrátor nebo autor smažou. Což je jedna z výhod oproti vzkazům v neregistrované části, kde jsou všechny vzkazy veřejné.

Vložit poznámku:

předmět:

text:

datum:

odeslat

Administrace poznámek

ahoj

tak už to frčí

Administrator /7. 6. 2005 09:41/

Upravit **Smazat**

Zobrazit stranu: 1

Obr. 16. Formulář pro napsání vzkazu

Poslední položkou v menu je „ADMIN“, která slouží jako vstup do části administrátorské, ve které je možné zadávat a editovat údaje v databázi. Ta to část je chráněna heslem, aby bylo zamezeno neoprávněnému vkládání a upravování údajů v databázi. Heslo je realizované pomocí java skriptu, který je uveden zde:

```
<SCRIPT LANGUAGE="JavaScript">
<!--
function Login(){
var done=0;
var username=document.login.username.value;
var password=document.login.password.value;
if (username=="login1" && password=="alfa") {
window.location="http://diplprace.wz.cz/admin/administrator"; done=1; }
```

```
if (done==0) { alert("Bylo zadáno špatné jméno nebo heslo!"); }  
}  
  
// -->  
</SCRIPT>
```

Po zadání hesla se otevře administrativní část (Obr.17), kde je možné buď zobrazit obsahy tabulek, které je samozřejmě možné mazat či editovat. Nebo můžeme zobrazit formulář pro zadávání nových údajů do databáze.



Obr. 17. Administrativní část databáze

ZÁVĚR

Hlavním cílem této diplomové práce bylo vytvoření virtuálního slovníku na téma datová bezpečnost a zaměřit se na možnosti využití výpočetní techniky při učení a vyučování.

V teoretické části práce je možné se nejprve seznámit se historií i současností ve využívání informačních technologií ve vyučování a učení. Byla zde popsána základní rozdělení informačních technologií jejich výhody i úskalí.

Dále se je možné blíže seznámit se základním prostředím pro tvorbu databázových aplikací. Je zde věnován dostatečný prostor k popisu používaných webserveru. Jsou zde uvedeny vhodné programovací jazyky, jejich výhody i nevýhody. Popsány jsou také v současnosti používané databázové systémy.

V praktické části byl pak celý systém navržen. Jsou zde podrobně popsány základní syntaxe použitých programovacích jazyků. Vše je seřazeno chronologicky. Nejprve je proveden nástin struktury HTML. Poté je popsán úvod do skriptovacího jazyka PHP. Kde jsou popsány základní příkazy a vlastnosti jazyka PHP. Dále pak následuje popis databázového systému SQL. Opět s ukázkami jednoduchých příkazů.

V poslední části je pak popsána celá aplikace. Ta se skládá ze tří částí, a to pro registrované a neregistrované uživatele a administrátora. Neregistrovaní uživatelé mají možnost se pohybovat po základní části aplikace, která je podobná registrované, ale oproti registrované části, mají zde uživatelé menší možnosti. Pro registrované uživatele je zde lepší možnost vyhledávání, možnost zanechat vzkaz administrátorovi, který je pro ostatní uživatele skrytý. Součástí webu je pak možnost podívat se na zajímavé odkazy a články. Jediný administrátor má pak možnost vkládat a editovat údaje které obsahuje slovník.

SEZNAM POUŽITÉ LITERATURY

- [1] SPAINHOUR, S. , ECKSTEIN R. *Webmaster v kostce – pohotová referenční příručka* Praha : Computer Press 1999, ISBN 80-7226-251-3
- [2] KOSEK. *HTML – tvorba dokonalých WWW stránek*, Grada Publishing, Praha 1998
- [3] PÍSEK SLAVOJ. *Javascript – efektní nástroj oživení www stránek*. Praha : Grada Publishink, 2001. 231 s. ISBN 80-247-0014-X
- [4] <http://www.root.cz>
- [5] Souček J. Aktive Server Pages (2000), <http://kuba.spika.cz/asp.htm>
- [6] Toman P. Programovací jazyk Java, <http://dione.zcu.cz/java/uvod.html>
- [7] Oficiální stránky PHP, <http://www.php.net>
- [8] Oficiální stránky WEB serveru Apache, <http://www.apache.org>
- [9] Oficiální stránky MySQL, <http://www.mysql.com>
- [10] Kosek J. Materiály k vyuce,
<http://badame.vse.cz/izi218/materialy/kapitola.skriptovani.html>
- [11] Historie a vývoj internetu[online]. 2004 – [cit. 2004-08-09]. Dostupný z WWW:
<http://www.pcsvet.cz/art/article.php?id=4725>
- [12] Vývoj PHP - cesta správným směrem? [online]. 2004 – [cit. 2005-01-24].
Dostupný z WWW: <http://interval.cz/clanek.asp?article=3752>
- [13] HALSALL, F. *Multimedia Communications*. New York: Pearson Addison Wesley, 2000. ISBN 02-013-9818-4.
- [14] HEATH, S. *Multimedia and Communications Technology*. Campbell: Focal Press, 1999. ISBN 02-405-1529-3.
- [14] BURNETT, R. *Perspectives on Multimedia: Communication, Media and Information Technology*. Mississauga: John Wiley & Sons, 2004. ISBN 04-708-6863-5.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

API	Aplications Program Interface
ASP	Aktive Server Pages
CSS	Cascade Style Sheets
CGI	Common Gateway Interface
DARPA	Defence Advanced Research Project Agency
DNS	Domain Name Server
EARN	European Academic and Research Network
EUNET	European UNIX Network
FTP	File Transfer Protocol
HTML	Hypertext Markup Language
HTTP	HyperText Transfer Protocol
IAB	Internet Activities Board
IETF	Internet Engeneering Task Force
IIS	Internet Information Server
IP	Internet Protocol
ISOC	Internet Society
JANET	Joint Academic Network
NSF	National Science Foundation).
PHP	Personál Home Page Tools (PHP Hypertext Preprocesor)
PNWS	PinkNet Web Server
SSL	Secure Societ Layer
SQL	Structured Query Languste
TCP	Transfer Control Protocol
URL	Uniform Resource Locator

SEZNAM OBRÁZKŮ

Obr. 1. Ukázka zpracování PHP skriptu.	27
Obr. 2. Struktura stránek	34
Obr. 3. Hlavní menu dokumentu	35
Obr. 4. Spodní menu dokumentu.	35
Obr. 5. Přihlášení na MySQL server.	36
Obr. 6. webového rozhraní pro správu databáze.	37
Obr. 7. jednoduchého webového rozhraní.	37
Obr. 8. Menu pro příkazy jazyka MySQL.	38
Obr. 9. Struktura systému z hlediska uživatelů.	46
Obr. 10. úvodní stránka.....	47
Obr. 11. první krok registrace.....	48
Obr. 12. druhý krok registrace.....	48
Obr. 13. třetí krok registrace.....	49
Obr. 14. Registrovaná část.....	49
Obr. 15. možnosti vyhledávání ve slovníku	50
Obr. 16. Formulář pro napsání vzkazu	51
Obr. 17. Administrativní část databáze.....	52

SEZNAM TABULEK

Tab. 1. Číselné datové typy.	42
Tab. 2. Časové datové typy.....	43
Tab. 3. Číselné datové typy.	43

SEZNAM PŘÍLOH

- P I Seznam všech použitých zdrojových kódů
- P II Ukázka zdrojového kódu index.php (uživatelská část)
- P III Ukázka zdrojového kódu admin.php

P I Ukázka zdrojového kódu

Zde je seznam všech skriptů které jsem použil v mé aplikaci.

Uživatelská část:

clanky.php	Skript se stará o zobrazení článků.
config.inc.php	Konfigurační soubor systému, obsahu styl a vzhled webu.
error.php	Skript vypisující chybové hlášení systému.
function.php	Obsahuje pomocné funkce systému (např. pro zobrazování poznámek).
hlavicka.php	Má na starosti zobrazení hlavního menu.
index.php	Úvodní stránka systému pro neregistrovaného uživatele.
login.php	Skript pro obsluhu přihlášení uživatele.
logout.php	Skript pro obsluhu odhlášení uživatele.
msg.php	Skript starající se o výpis oznamovacích informací v systému (např. poznámka byla vložena, byl jste odhlášen ze systému).
odkazy.php	Zobrazuje odkazy na jiné stránky.
pata.php	Zobrazuje spodní menu.
reg.php	Tento skript má nestarosti registraci nových uživatelů.
slovník.php	Skript obstarávající vyhledávání ve slovníku.
sqlconnect.php	Připojení k SQL databázi.
telo.php	Hlavní tělo aplikace.
vzkazy.php	Skript sloužící pro zanechání vzkazu v neregistrované části.
zajimavosti.php	Skript zobrazující zajímavé články.

Registrovaná část:

admin.php	Skript pro správu slov (mazání, zobrazování, editaci).
administrator.php	Slouží pro vkládání a editaci údajů do databáze.
form.php	Vkládací formulář pro poznámky.

hlavicka1.php	Má na starosti zobrazení hlavního menu.
hlslova.php	Skript zobrazující kolikrát bylo které slovo hledané.
checklogin.php	Skript vkládaný do všech registrovaných stránek. Obsahuje autorizaci uživatele a načtení informací o uživateli s jeho nastavením.
index.php	Tento skript je uložen ve složce, určuje strukturu webu pro přihlášené uživatele.
logadmin.php	Má na starosti zaheslování přístupu ke skriptu administrator.php.
pata1.php	Zobrazuje spodní menu.
pridej.php	Skript obstarávající přidávání nových slov.
update.php	Skript pro modifikaci poznámek.
vzkazy.php	Slouží pro vkládání vzkazů v registrované části.

P IVI Ukázka zdrojového kódu index.php (uživatelská část)

```
<?php
Header("Pragma: No-cache");
Header("Cache-Control: No-cache, Must-revalidate");
Header("Expires: ".GMDDate("D, d M Y H:i:s")." GMT");
require "./sqlconnect.php";
@$res=connect_db();
$order='testname';
if (isset($_REQUEST['sort'])) {
    switch ($_REQUEST['sort'])
    {
        case 0:
            $order = 'testname';
            break;
        case 1:
            $order = 'filesize DESC';
            break;
        case 2:
            $order = 'filetime DESC';
            break;
        default:
            //vyber souboru a jejich vypis
            $order = 'testname';
            break;
    } //konce Issetu sort
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html><head><title>Slovník- datová bezpečnost</title>
<meta http-equiv=Content-Type content="text/html; charset=windows-1250">
<link rel="stylesheet" type="text/css" href="./my.css">
<meta content="MSHTML 5.50.4134.600" name=GENERATOR>
</head>
<body>
<div id=hlavicka>
<?
readfile ('hlavicka.php')
?>
```

```

</div>
<div id=obal>
<!--leve odkazy-->
  <div id=left-menulogin>
    <br><br>
    <div class="nadpisek">Přihlášení do systému:</div>
    <br>
    <fieldset class="prihlaseni">
      <form name="login" method="post" action="login.php">
        <input type="hidden" name="sent" value="">
        <table border="0" cellpadding="0" cellspacing="0">
          <tr valign="middle">
            <td align="right"> <small><strong>jméno:&nbsp;</td>
            <td align="left"> <input type="text" name="login" size="15" value="<?php if(IsSet($login)) echo
$login; ?>"> </td> </tr>
            <tr valign="middle">
              <td align="right"> <small><strong>heslo:&nbsp;</td>
              <td align="left"> <input type="password" name="password" size="15"> </td> </tr>
              <tr valign="middle"> <td></td><td>
                <BUTTON TYPE=submit name="send" class="tlacitko">
                <IMG SRC="/images/prihlasit.png"></BUTTON>
              </td> </tr>
            </form></table> </fieldset>
          <br><br><br>
          Pokud jste se ještě nezaregistrovali, tak prosím zde:
          <a href="/reg.php"> </a>
        </div>
        <!--center text-->
        <div id=center>
          <br>
          <? readfile ('telo.php') ?>
        </div>
        <? readfile ('pata.php') ?>
      </body>
    </html>

```

P III Ukázka zdrojového kódu admin.php

```
<?php

// vkládání, úprava poznamek
// zabrání cashování
Header("Pragma: No-cache");
Header("Cache-Control: No-cache, Must-revalidate");
Header("Expires: ".GMMdate("D, d M Y H:i:s")." GMT");

//ini_set('session.use_trans_sid', 1); // zajisti prenos session id

require "../checklogin.php"; // je autor prihlasen ?
//require "../sqlconnect.php"; // pripoji k databazi
require "../function.php"; // nacte soubor s dulezitymi funkcemi
require "../msg.php"; // chybove hlasky a zpravy

//připojení k databázi
@$res=connect_db();

?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html><head><title>Datová bezpečnost - slovník</title>
<meta http-equiv=Content-Type content="text/html; charset=windows-1250">
<link rel="stylesheet" type="text/css" href="/my.css">
<meta content="MSHTML 5.50.4134.600" name=GENERATOR>
</head>
<body>
<?
readfile ('hlavicka1.php')
?>
</div>
<div id=obal>
    <div id=center_u>
        <div id=hledani>
            <h2>Zobraz celý slovník</h2><br>
            Tabulka "Slova" obsahuje všechny slova v databázi.<br>
            Tabulka "Hl.slova" nám zobrazí, kolikrát bylo hledání použito, a kolikrát byly jednotlivé slova hledány.
            <form>
            <select name="tabulka">
            <option value="0">---Vyberte tabulku---</option>
            <option value="1">Slova</option>
            <option value="2">Hl.Slova</option>
            </select>
            <input type="submit" value="Zobraz tabulku">
            </form>
            <?
            if($tabulka=="1"):
            $dotaz = Mysql_query("SELECT * FROM Slova");
            echo "<table border=1>";
            echo "<tr><td>\n";
            echo "Slovo\n</td><td>\n";
            //echo "Url\n</td><td>\n";
            echo "Popis\n</td><td>\n";
            while($zaznam = Mysql_fetch_array($dotaz)):
            echo "<tr><td>\n";
            echo $zaznam["slovo"]."\n</td><td>\n";
            //echo $zaznam["url"]."\n</td><td>\n";
            echo $zaznam["popis"]."\n</td><td>\n";
```

```

endwhile;
echo "</table>";
elseif($tabulka=="2"):
$dotaz = Mysql_query("SELECT * FROM HlSlova");
echo "<table border=1>";
echo "<tr><td>\n";
echo "Slovo\n</td><td>\n";
echo "Hledáno\n</td><td>\n";
while($zaznam = Mysql_fetch_array($dotaz)):
echo "<tr><td>\n";
echo $zaznam["slovo"]."\n</td><td>\n";
echo $zaznam["pocet"]."x\n</td><td>\n";
endwhile;
echo "</table>";
endif;
//akce, které skoro nic nedělají - pouze se ptají
if($action=="check"):
$dotaz = Mysql_query("SELECT * FROM $tabulka");
$zaznam = Mysql_fetch_array($dotaz);
if($tabulka=="Slova"):
?>
<form method="post">
<input type="hidden" name="action" value="uprav">
<input type="hidden" name="tabulka" value="Slova">
<input type="hidden" name="id" value="<?echo $id?>">
Jméno : &nbsp;<input type="text" name="jmeno" value="<?echo $zaznam["slovo"]?>"><br>
Url : &nbsp;<input type="text" name="url" value="<?echo $zaznam["url"]?>"><br>
Popis : &nbsp;<input type="text" name="popis" value="<?echo $zaznam["popis"]?>"><br>
<input type="submit" value=" OK "><br>
</form>
<?
else:
?>
<form method="post">
<input type="hidden" name="action" value="uprav">
<input type="hidden" name="tabulka" value="HlSlova">
<input type="hidden" name="id" value="<?echo $id?>">
Slovo : &nbsp;<input type="text" name="slovo" value="<?echo $zaznam["slovo"]?>"><br>
Kolikrát : &nbsp;<input type="text" name="pocet" value="<?echo $zaznam["pocet"]?>"><br>
<input type="submit" value=" OK "><br>
</form>
<?
endif;
endif;
?>

</div>

<div>
<div id=right>
<h2>Hledání slova</h2>
<form action="admin.php?akce=najit" name="najdi" method="post">
<b>Zadej hledané slovo: </b> <input type="text" name="slovo"><br>
<input type="checkbox" name="diakritika" value="true"> Ignorovat diakritiku <BR>
<input type="submit" value="Najdi to!">
</form>
<hr>
<?
if($akce=="najit"):
//prohledej databazi jestli v ní je slovo $slovo
$prkaz = mysql_query("SELECT * FROM Slova WHERE slovo LIKE '$slovo%'");

```

```

if($diakritika):
$slovo = strtr($slovo,"ěščřžýáíéüöäõňĚŠČŘŽÝÁÍÉÓŇÜÓŇ","escrzyaieuoaonESCRZYAIEUOAON");
endif;

```

```

//pokud jsi v databázi našel výskyt slova $slovo tak vypiš všechno co obsahuje $slovo

```

```

$num = mysql_num_rows($prikaz);
if($num > 0):
echo "<h3>Výsledky hledání</h3>";
while($zaznam = mysql_fetch_array($prikaz)):
$url = $zaznam['url'];
$nazev = $zaznam['nazev'];
$popis = $zaznam['popis'];
echo "<h5><a href=\"\$url\">$nazev</a> - $popis<br>\n</h5>";
endwhile;
//jinak vypiš že není možné najít slovo $slovo
else:
echo "<h3>Není možné najít slovo $slovo !!</h3><br>\n";
endif;
endif;
//prohledej tabulku HISlova a pokud tam již takovéto slovo je,
//tak k počtu přičti 1 a pokud není tak ho vlož s počtem 1
$dotaz = mysql_query("SELECT * FROM HISlova WHERE slovo = '$slovo'");
$pocet = Mysql_num_rows($dotaz);

```

```

if($pocet!=0) mysql_query("UPDATE HISlova SET pocet = pocet + 1 WHERE slovo = '$slovo'");
else mysql_query("INSERT INTO HISlova VALUES ('','$slovo', 1)");

```

```

?>

```

```

</div>

```

```

<br><br><br><br><br>

```

```

<?

```

```

readfile ('pata1.php')

```

```

?>

```

```

</div>

```

```

</body>

```

```

</html>

```