

Aktualizace informačního systému firmy

Update of the company information system

Michal Bajer

Bakalářská práce
2008



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav aplikované informatiky

akademický rok: 2007/2008

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Michal BAJER**

Studijní program: **B 3902 Inženýrská informatika**

Studijní obor: **Informační technologie**

Téma práce: **Aktualizace informačního systému firmy**

Zásady pro vypracování:

1. **Analýza současného stavu.**
2. **Návrh na zdokonalení a zefektivnění používaného systému.**
3. **Řešení navržených vylepšení.**
4. **Nasazení nového systému do provozu a jeho srovnání s původním systémem.**

Rozsah práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tisková/elektronická**

Seznam odborné literatury:

1. **STANÍČEK, P.** CSS - hotová řešení. Computer Press, 2006, ISBN: 80-251-1031-1.
2. **LACKO, L.** PHP a MySQL - hotová řešení. Computer Press, 2006, ISBN: 80-251-1249-7.
3. **PROKOPOVÁ, Z.**: Databázové systémy MySQL+PHP. FAI UTB Zlín, s. 126, 2006, Vysokoškolská skripta. ISBN 80-7318-486-9.
4. **RIORDAN, Rebecca M.** Vytváříme relační databázové aplikace. Praha : Computer Press, 2000. 280 s. ISBN 80-7226-360-9.
5. **SCHNEIDER, R.D.** MySQL - Oficiální průvodce tvorbou, správou a laděním databází. Grada, ISBN: 80-247-1516-3.
6. **KRČMÁŘ JAKUB** - Adobe Photoshop - praktický webdesign. Grada, ISBN: 80-247-1423-X.

Vedoucí bakalářské práce:

doc. Ing. Zdenka Prokopová, CSc.

Ústav aplikované informatiky

Datum zadání bakalářské práce:

20. února 2008

Termín odevzdání bakalářské práce:

5. května 2008

Ve Zlíně dne 20. února 2008

prof. Ing. Vlastimír Vašek, CSc.
děkan



doc. Ing. Ivan Zelinka, Ph.D.
ředitel ústavu

ABSTRAKT

Tato bakalářská práce se zabývá analýzou stávajícího informačního serveru firmy Jinlong s.r.o. a následným navržením nového a efektivnějšího systému.

Hlavním cílem práce je nalezení výhodnějšího řešení a jeho praktická realizace v podobě webového serveru. Pro tento případ byl vybrán databázový systém MySQL ve spolupráci s programovacím jazykem PHP. V závěru práce je porovnán stávající a minulý informační server a uvedeny jeho klady oproti minulému.

Klíčová slova: webové stránky, databázový systém, MySQL, PHP, webdesign

ABSTRACT

This bachelor work is engaged in analysis of the current information server of the firm Jinlong s.r.o. and the resulting suggestion of a new and a more effective system.

The main target of the work is the finding of a more convenient solution and its practical realization in the form of the web server. For this case it was chosen the database system MySQL, in cooperation with the programming language PHP. In the summary are compared the current and the last information server and there are given its positives against the last one.

Keywords: websites, database system, MySQL, PHP, webdesign

Děkuji tímto mojí vedoucí bakalářské práce, doc. Ing. Zdence Prokopové, Csc. za odborné rady a cennou pomoc, kterou mi poskytla při psaní této bakalářské práce.

Prohlašuji, že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků, je-li to uvolněno na základě licenční smlouvy, budu uveden jako spoluautor.

Ve

.....

Zlíně

Podpis diplomanta

OBSAH

ÚVOD	8
I. TEORETICKÁ ČÁST	9
1 DEFINICE WEB DESIGNU	10
1.1 PYRAMIDA WEB DESIGNU	11
1.2 WEB JAKO MÉDIUM	12
1.3 TYPY WEBOVÝCH SERVERŮ A JEJICH DĚLENÍ	13
1.3.1 Abstraktní dělení	13
1.3.2 Vizuální dělení	15
1.4 TYPY DATABÁZÍ	16
1.5 HISTORIE RELAČNÍHO MODELU	17
1.6 ANATOMIE RELAČNÍ DATABÁZE	18
1.7 VZTAHY	21
1.8 PHP	24
1.8.1 Historie PHP	24
1.9 MYSQL	26
II. PRAKTICKÁ ČÁST	28
2 ANALÝZA SOUČASNÉHO STAVU SYSTÉMU	29
2.1 JEDNOTLIVÉ SEKCE WEBOVÉHO SERVERU	31
2.1.1 Restaurace	31
2.1.2 Bar	32
2.1.3 Fitness, squash, bowling	33
3 NÁVRH NOVÉHO WEBOVÉHO SERVERU	34
3.1 GRAFICKÝ NÁVRH	34
3.2 STRUKTURA A NÁVRH DATABÁZÍ	39
3.2.1 Sekce Sport Bar	39
3.2.2 Sekce restaurace	43
3.2.3 Sekce bowling, squash, fitness a solárium	45
ZÁVĚR	46
ZÁVĚR V ANGLIČTINĚ	47
SEZNAM POUŽITÉ LITERATURY	48
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	49

SEZNAM OBRÁZKŮ.....	50
SEZNAM TABULEK	51
SEZNAM PŘÍLOH.....	52

ÚVOD

V dnešní době je velice důležité prosazování na trhu. Firmy využívají různých možností marketingu, který jim pomáhá se v dnešní době prosadit. Mezi ty nezákladnější patří reklama jak v podobě různých letáků, bannerů a dalších předmětů, které firma nabízí lidem tak webový marketing, který je v dnešní době velice užitečný.

Pod slovem webový marketing si můžeme představit jakousi prezentaci firmy na internetu. Prezentace se většinou dělá pomocí webových stránek, nebo také pomocí různých bannerů, které jsou na různých stránkách umístěny. Každá firma, která chce dobře reprezentovat na trhu by měla mít takovou webovou prezentaci. Hlavním důvodem je velký rozvoj internetu a také velký nárůst uživatelů, kteří internet využívají čím dál více k vyhledávání informací jak o firmách tak i např. o internetových obchodech, službách.

Bakalářská práce je rozdělena do dvou částí. V první teoretické části je popsán a vysvětlen co je to webdesign, který je při vytváření webové prezentace velice důležitý. Dále se zmíníme o systémech, které se využívají při vytváření takové prezentace. Mezi ně patří SQL a programovací jazyk PHP. Druhá část se zabývá praktickou stránkou problému. Firma Jinlong s.r.o. se rozhodla svoji stávající webovou prezentaci předělat a vylepšit. V prvním kroku práce se zabýváme analýzou současného stavu systému, nalezením jeho nedostatků a v druhé části navrhne lepší, efektivnější informační systém.

Hlavním cílem je vytvoření nového webového serveru pro firmu Jinlong s.r.o. a jeho praktická realizace.

I. TEORETICKÁ ČÁST

1 DEFINICE WEB DESIGNU

Existuje pět oblastí pokrývajících aspekty web designu:

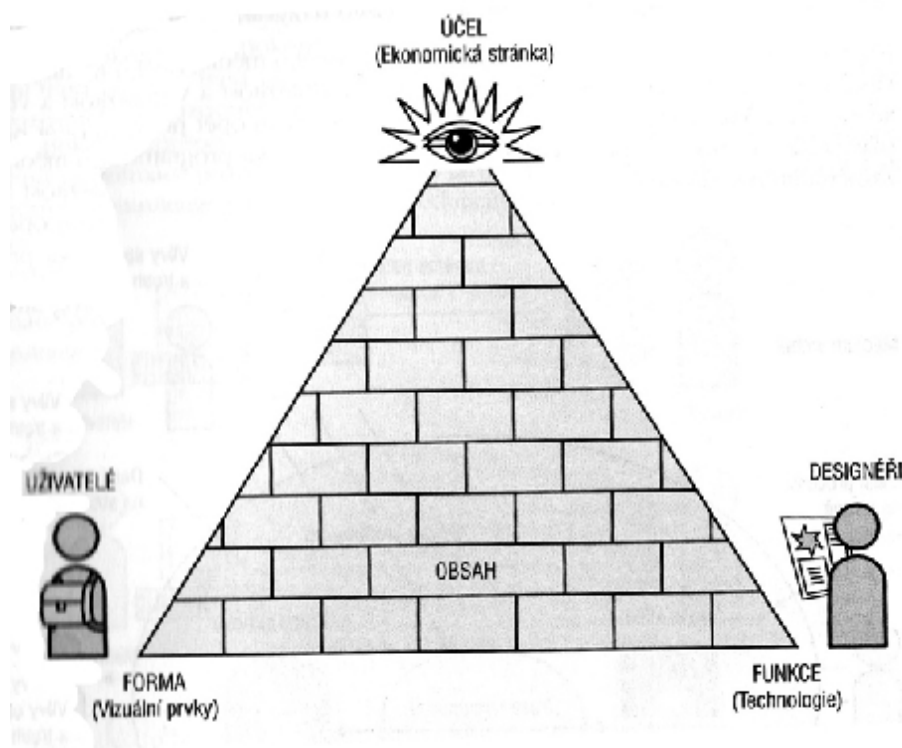
- **Obsah** – zahrnuje formu a uspořádání obsahu webového serveru. Může se jednat například o způsob psaní textu, jeho uspořádání, prezentaci a strukturu. Forma a uspořádání obsahu je nejčastěji prezentována HTML kódem.
- **Vizuální prvky** – vztahují se k rozvržení obrazovky webového serveru. Rozvržení je obvykle vytvořeno pomocí technologie HTML, CSS, případně Flash a může obsahovat grafické prvky s dekorační nebo navigační funkcí. Vizuální stránka webového serveru je nejzřetelnějším aspektem web designu, nicméně netvoří základní nebo nejdůležitější součást tohoto oboru.
- **Technologie** – zatímco použití různých základních webových technologií, jako je HTML nebo CSS, jsou přirozenou součástí webu, technologie v tomto kontextu odpovídá rozličným interaktivním prvkům webového serveru, obzvláště těm, které byly vytvořeny programovacími metodami. Mezi takové položky se řadí například klientské skriptovací jazyky, jako JavaScript, případně serverové aplikace, jako servlety Java.
- **Výkon** – rychlost a spolehlivost přenosu dat prostřednictvím sítě Internet nebo vnitřní sítě společnosti jsou spojeny s použitým hardwarem/softwarem a strukturou sítě.
- **Účel** – záměr, s jakým je webový server vytvořen, často spojený s ekonomickými otázkami, je pravděpodobně nejdůležitější součástí web designu. Na záměr, s jakým se webový server tvoří, by se měl brát zřetel při všech rozhodnutích týkajících se ostatních oblastí.

Rozsah, v jakém každý faktor web designu ovlivňuje webový server, se samozřejmě může lišit podle typu vytvářeného serveru. Osobní domovská stránka obvykle nemá ekonomický rozměr nákupního webového serveru. Intranet pro výrobní společnost nemusí mít vizuální prvky veřejného webového serveru propagujícího akční film. Vlastní význam výrazu „web design“ se může poněkud lišit.

1.1 Pyramida web designu

Jedním ze způsobů, jak znázornit všechny součásti web designu, je přirovnání k webové pyramidě (viz Obrázek 1). Obsah se skládá z cihel tvořících pyramidu, přičemž základy pevně spočívají na vizuálních prvních a technologii a jsou značně závislé na finančních otázkách, bez nichž by realizace našeho projektu neměla smysl.

Jako web designéři se své servery pokoušíme plánovat obezřetně, nicméně jejich vytvoření je obtížné. Tvorba webu vyžaduje týmovou práci a úplné porozumění webu jako médiu. I za předpokladu, že jsme odborníci schopní vybudovat překrásný a funkční webový server, mohou uživatelé na vytvořené dílo hledět s rozpaky. Designéři často tráví více času zvažováním vlastních potřeb a představ, místo potřeb uživatelů. Náš web jako pyramida se může stát příliš podobný opravdovým pyramidám – neprostupným hrobkám, které nás přivádějí k otázce, zda uživatelé přicházející k našim pomníkům prostřednictvím sítě WWW mohou vůbec nalézt vchod. Pyramida poskytuje designérům jednoduchý způsob, jak vzít v úvahu všechny vzájemně se ovlivňující aspekty web designu, ale neumožňuje hlubší porozumění médiu zvanému web.



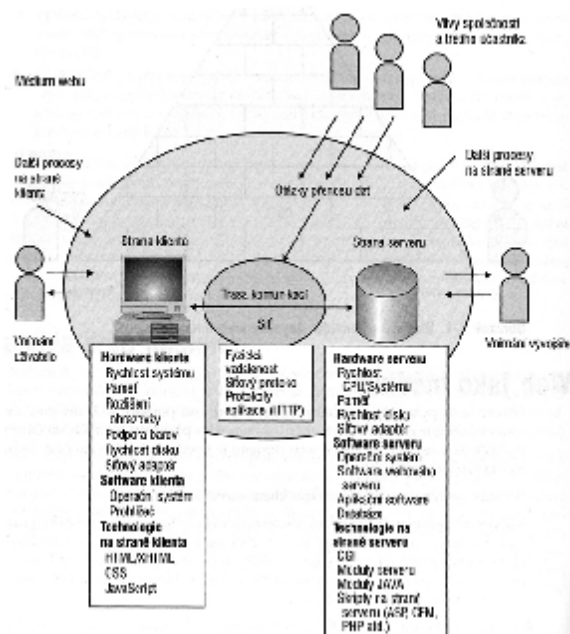
Obrázek 1 Thomas A. Powell - Web design – kompletní průvodce – pyramida web designu

1.2 Web jako médium

Přirovnání k pyramidě je velmi abstraktní možností popsání web designu, avšak je užitečným nástrojem pro znázornění vzájemného působení různých součástí webu. Praktičtější metodou přiblížení web designu je uvedení různých součástí webového média (viz Obrázek 2).

Webové servery jsou systémy typu klient-server a skládají se:

- **Na straně serveru** - z hardwaru a softwaru webového serveru, položky programování a vestavěných technologií. Mezi technologie mohou patřit například jednoduché programy CGI napsané v jazyku PERL nebo složité vícevrstvé aplikace založené ja jazyku Java a mohou obsahovat technologie back-end, jako jsou databázové servery podporující webový server.
- **Na straně klienta** – strana klienta se týká webového prohlížeče a jeho podporovaných technologií, jakou jsou jazyky HTML, CSS a JavaScript a ovládacích prvků ActiveX nebo modulů plug-in Netscape, které jsou používány pro vytvoření prezentace stránky nebo poskytnutí interaktivních funkcí.
- **Sít'** – síť zahrnuje různé položky připojení využité pro přenos obsahu webového serveru k uživateli. Takovými položkami mohou být různé sítě ve veřejné síti Internet nebo privátní připojení uvnitř společnosti – často nazývaná intranet.



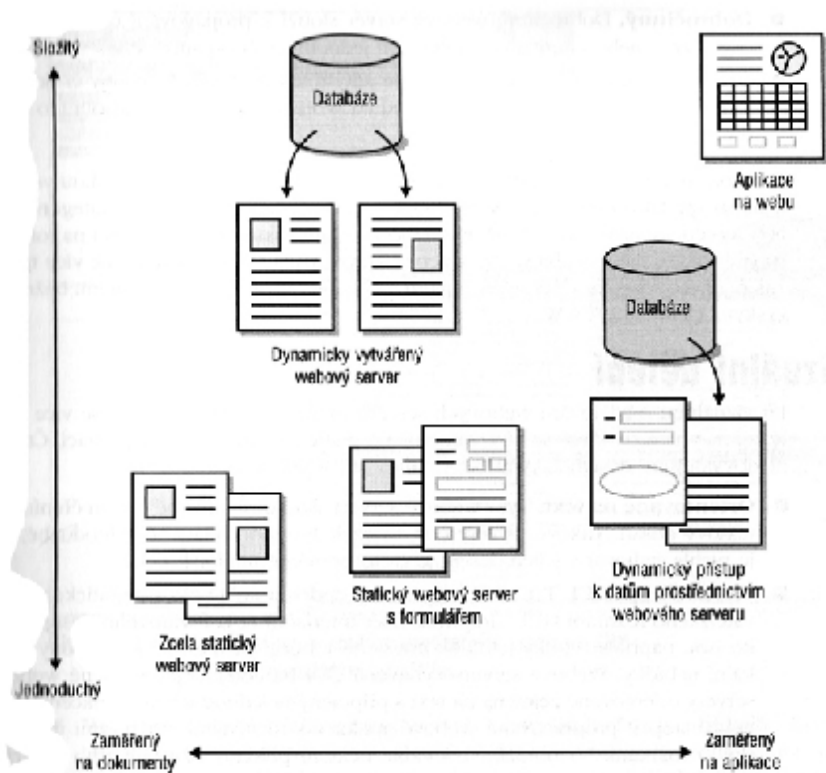
Obrázek 2 Thomas A. Powell - Web design – kompletní průvodce – web jako médium

Webové servery se používají jako prostředek komunikace mezi provozovateli serveru a jeho uživateli a v některých případech pouze mezi uživateli. Provozovatelé webových serverů obvykle stanovují základní pravidla interakce, zatímco uživatelé server navštěvují a pokoušejí se použít zde prezentovaný obsah nebo prostředky. Během komunikace si většina uživatelů při správném fungování webového serveru obvykle existenci webu jako média neuvědomuje. Uživatelé často nerozlišují jednotlivé součásti, jako je síť, jazyk HTML, formátovací sady a JavaScript – dokud vše funguje správně. V případě pomalého serveru nebo takového serveru, který způsobuje vizuální nebo funkční chyby, může uživatel médium zaznamenat, ale stále nerozezná příčinu potíží.

1.3 Typy webových serverů a jejich dělení

1.3.1 Abstraktní dělení

Nejprve uvažme, zda se webový server zaměřuje na informace nebo na úlohy. Někdy můžeme tuto odlišnost popsat jako rozdíl mezi webovým serverem zaměřeným na dokumenty a webovým serverem zaměřeným na aplikace. Webové servery zaměřené na dokumenty nebo informační webové servery poskytují uživatelům informace, ale umožňují velmi omezenou interaktivnost (neumožní uživateli více, než jen prohlížet, vyhledávat či uspořádat uvedené informace). Webové servery, které jsou orientované na úlohy nebo aplikace, umožňují uživateli interakci s informacemi či provedení určité úlohy, jako je například převedení částky z bankovního účtu nebo nákup nového svetru. Smíšené webové servery umožňují od každého trochu; tyto servery se stávají běžnějšími a hranice mezi informačními a aplikačními se tedy vytrácí. Na Obrázku 3 je znázorněna spojitost mezi jednoduchým statickým webovým serverem orientovaným na dokumenty a propracovanými softwarovými aplikacemi.



Obrázek 3 Thomas A. Powell - Web design – kompletní průvodce

Další možnosti, jak seskupit webové servery, jsou následující všeobecné kategorie:

- **Informační webové servery** – tyto servery poskytují informace k určitému tématu nebo informace o nějaké organizaci. Jsou nejobvyklejšími webovými servery v síti Internet a často po jisté době přijímají aspekty ostatních kategorií webových serverů.
- **Obchodní webové servery** – tento typ webového serveru může být používán k provedení nějaké transakce nebo úkoly. Do této kategorie spadají webové servery elektronické obchodování.
- **Zábavní webové servery** – tyto servery jsou určeny pro hraní her nebo pro určitou formu zábavné interakce, což může zahrnovat transakční, společenské a informační prvky.
- **Ostatní webové servery** – zde jsou obsaženy umělecké nebo experimentální webové servery, osobní stránky, jako jsou webové blogy, a webové servery, které se neřídí běžnými webovými zásadami nebo mají přesně vymezený ekonomický cíl

Webové servery můžeme také rozdělit podle organizací, které je provozují, případně za ně určitým způsobem platí. V tomto typu kategorizace existuje pět hlavních seskupení:

- **Komerční** – webový server v této skupině je vybudován a provozován organizací nebo jednotlivcem pro komerční účely. Může se jednat o přímé elektronické obchodování nebo nepřímou propagaci nákupu zboží či služeb mimo síť WWW.
 - **Státní** – nadřazeným subjektem tohoto webového serveru je v zásadě státní organizace a účelem tohoto serveru je uspokojení určitých společenských nebo právních potřeb.
 - **Vzdělávací** – nadřazeným subjektem tohoto typu webového serveru je určitá vzdělávací instituce (může být i státní) a účelem serveru je podpora vzdělávání nebo výzkumné cíle.
 - **Dobročinný** – dobročinný webový server slouží k propagování cílů neziskové organizace nebo charitativních činností jednotlivce či organizace.
 - **Osobní** – tento webový server existuje díky rozhodnutí nějaké osoby či skupiny z různých důvodů, většinou jako produkt tvořivosti nebo forma osobního vyjádření
- Kategorizace může být složitá. Například vzdělávací webové servery mohou ve skutečnosti spadat do státní kategorie. Některé webové servery v osobní kategorii mohou teoreticky patřit do dobročinné nebo komerční skupiny.

1.3.2 Vizuální dělení

Při vizuálním rozdělování webových serverů můžeme rozlišit ty, které se více spoléhají na text a ty, které se více zaměřují na grafickou prezentaci či ilustraci. Čtyřmi nejobvyklejšími designérskými odvětvími v síti WWW jsou:

- **Orientované na text** – tyto webové servery jsou designované se zaměřením na textový obsah. Takové servery jsou relativně jednoduché, lze je rychle stahovat a jejich design je často poměrně minimalistický.
- **Styl rozhraní GUI** - tyto webové servery dodržují určité zásady grafického uživatelského rozhraní GUI (Graphical User Interface) ze softwarového designu, jako jsou například řádky nabídek umístěné v horní části obrazovky, ikony a lokální nabídky. Webové servery vybavené GUI mohou být jednoduché webové servery orientované zejména na text s připojenými jednoduchými funkcemi GUI, nebo naopak propracované webové aplikace s rozličnými možnostmi uživatelského rozhraní.

- **Metaforické** – metaforické webové servery si vypůjčují nápady z běžného života. Například webový server o autech může v designu a navigaci použít palubní desku a volant. Metaforický navržený webový server má tendenci být velmi vizuálně založený nebo interaktivní. Některé uživatele toto může odradit a jiné upoutat.
- **Experimentální** – experimentální designy se pokoušejí provádět určité věci odlišně od běžně zavedené praxe. Tvořivost, nepředvídatelnost, inovace a dokonce i nahodilost jsou často použity u webových serverů sledujících experimentální styl designu.

V síti WWW samozřejmě nalzáme kombinace forem nebo potenciální nové kategorizace webových serverů. Jak byste například kategorizovali portálový webový server, poskytující na jediné stránce množství obsahu, volby navigace a také možnosti týkající se společnosti? Tento styl designu je jistě použit u velkého množství webových serverů. Při pohledu na druhy webových serverů, jakou jsou servery pro elektronické obchodování, obzvláště pak webové servery s „katalogy a nákupními košíky“, stejně jako osobní webové servery s deníky online, zvané „Weblogy“ či „blogy“, je možné vidět potenciální vzrůst dalších kategorií designu.^[1]

1.4 Typy databází

Co je to databáze? Jak asi víte, databáze je uspořádaná kolekce dat používaná k modelování nějakého typu organizace nebo organizačního procesu. Opravdu nezáleží na tom, zda ke sběru a ukládání dat používáte papír nebo počítačový systém. Pokud sbíráte a ukládáte data nějakým organizovaným způsobem za určitým účelem, pak máte databázi.

V rámci správy databází se v zásadě používají dva typy databází: operační databáze a analytické databáze. Operační databáze jsou dnes páteří mnoha společností, organizací a institucí po celém světě. Tento typ databáze se používá především ke každodennímu sběru, změnám a správě dat. Ukládaná data jsou dynamického typu, což znamená že se neustále mění a vždy odrážejí aktuální informace. Organizace jako obchody, výrobní společnosti, nemocnice, kliniky a nakladatelství používají operační databáze, protože jejich data jsou v neustálém pohybu.

Oproti tomu analytické databáze ukládají a sledují historická časově závislá data. Analytická databáze je hodnotní při sledování trendů, zobrazování dlouhodobých

statistických dat nebo při vytváření taktických či strategických obchodních projekcí. Ukládaná data jsou statického typu, což znamená, že se tato data nikdy nemění (Nebo se mění jen velmi výjimečně). Informace získávané z analytických databází odrážejí jen určitý okamžik a obvykle nejsou aktuální. Příklady organizací, které používají analytické databáze, jsou chemické laboratoře, geologické společnosti a firmy marketingové analýzy.

1.5 Historie relačního modelu

Existuje několik typů databázových modelů. Používání některých z nich (např. hierarchického a síťového) pomalu klesá, zatímco jiné (např. relační) se stále více rozšiřují. V knihách můžete rovněž narazit na pojednání o „objektovém“ neboli „objektově orientovaném“ modelu. Třebaže tyto největší modely jsou v současné době detailně zkoumány a vyvíjeny, věříme, že budou nejspíše implementovány jako rozšíření relačního modelu. Ve skutečnosti řada činností výboru ANSI, zaměřeného na další standart SQL, směřuje k tomu, aby se do jazyka SQL začlenily i objektově orientované koncepce.

Relační databáze byla poprvé představena v roce 1969 a zřejmě se jedná o dnes nejpoužívanější model v oblasti správy databází. Otcem modelu je dr. Edgar F. Codd, který pracoval na konci 60. let jako výzkumný pracovník IBM. Právě v té době hledal nové způsoby zpracování velkého množství dat. Zásadní nespokojenost s dostupnými databázovými modely a databázovými produkty jej přinutila vymýšlet možnosti aplikování určitých matematických disciplín a struktur za účelem vyřešení zásadních problémů, s nimiž se setkával. Jako matematik věřil, že je možné aplikovat specifická odvětví matematiky k řešení takových problémů, jako je nadbytečnost dat, slabá integrita dat a přílišná závislost struktury databáze na její fyzické implementaci.

Dr. Cood formálně představil svůj nový relační model v zásadní práci nazvané A Relational Model of Data for Larde Shared Databanks (Relační model dat pro velké sdílené databanky) v červnu 1970. Svůj nový model vystavěl na dvou matematických odvětvích – teorii množin a logických predikátech prvního rádu. Samotný název modelu je odvozen z termínu „relace“ neboli vztah, který je součástí teorie množin. Velmi často se chybně uvádí, že relační model se nazývá podle toho, že tabulky v relační databázi lze vzájemně provázat, vytvořit mezi nimi vztahy.

1.6 Anatomie relační databáze

Tabulky

Tabulky jsou základními strukturami v databázi; každá tabulka vždy představuje jediný, specifický subjekt. Logické pořadí záznamů a polí v tabulce nemá vůbec žádný význam. Každá tabulka obsahuje přinejmenším jedno pole – označované jako primární klíč – , které jednoznačně identifikuje jednotlivé záznamy. Právě díky těmto dvěma posledním charakteristikám tabulek existují data v relační databázi nezávisle na způsobu svého fyzického uložení v počítači. To je pro uživatele vynikající, protože při přebírání dat nějakého záznamu nemusejí znát jeho fyzické umístění.

Subjektem reprezentovaným danou tabulkou může být objekt nebo událost. Je-li subjektem objekt, pak tabulka reprezentuje něco hmatatelného, například osobu, místo nebo věc. Bez ohledu na typ má každý objekt charakteristiky, které lze uložit jako data. Tato data je následně možné zpracovávat téměř nekonečným množstvím způsobů. Piloti, výrobky, stroje, studenti, budovy a vybavení, to všechno jsou příklady objektů reprezentovatelných tabulkou. Obrázek 4 ilustruje jeden z nejobvyklejších příkladů tohoto typu tabulek.

Customers (zákazníci)

CustomerID	FirstName	LastName	StreetAddress	City	State	ZipCode
1010	Michael	Davollo	672 Lamont Ave	Houston	TX	77201
1011	Margaret	Peacock	667 Red River Road	Austin	TX	78710
1012	Estela	Purdit	2500 Rosales Lane	Dallas	TX	75260
1013	Merk	Rosales	323 Advocate Lane	El Paso	TX	79915
1014	Consuelo	Maynez	3445 Cheyenne Road	El Paso	TX	79915
1015	Ryan	Ehrlich	455 West Palm Ave	San Antonio	TX	78284

Záznamy (records)

Pole (fields)

Obrázek 4 Myslíme v jazyku SQL – ukázková tabulka

Je-li subjektem tabulky událost, pak daná tabulka reprezentuje něco, k čemu dochází v určitém okamžiku a má charakteristiky, jež si přejete zaznamenat. Tyto charakteristiky lze uložit jako data a následně je zpracovávat jako informace přesně stejným způsobem, jako by tabulka představovala specifický objekt. Příklady zaznamenávaných událostí jsou soudní slyšení, distribuce fondů, výsledky laboratorních testů a geologické výzkumy. Obrázek 5 ukazuje příklad tabulky reprezentující událost, s níž se občas setká každý z nás – kontrola u lékaře.

Patient Visit (kontroly pacientů)

Patient ID	Visit Date	Visit Time	Physician	Blood Pressure	Temperature
92001	1998-05-01	10:30	Hernandez	120 / 80	98.8
97002	1998-05-01	13:00	Piercy	112 / 74	97.5
99014	1998-05-02	09:30	Rolson	120 / 80	98.8
96105	1998-05-02	11:00	Hernandez	160 / 90	99.1
96203	1998-05-02	14:00	Hernandez	110 / 75	99.3
98003	1998-05-02	09:30	Rolson	120 / 80	98.6

Obrázek 5 Myslíme v jazyku SQL –tabulka reprezentující událost

Pole

Pole je nejmenší strukturou v databázi a představuje určitou charakteristiku subjektu té tabulky, do níž patří. Pole jsou struktury, jež se skutečně používají k ukládání dat. Data v těchto polích pak lze přebírat a prezentovat jako informace v téměř libovolné představitelné konfiguraci. Pamatujte, že kvalita informací, získaných z dat, přímo závisí na době, kterou jste věnovali zajištění strukturální integrity a datové integrity samotných polí. Význam polí nelze nikdy podcenit.

Každé pole v dobře navržené databázi obsahuje jednu a pouze jednu hodnotu a jeho název identifikuje typ obsažené hodnoty. Zadání dat takového pole je pak velmi přirozené. Spatříte-li pole s názvy FirstName (křestní jméno), LastName (příjmení), City (město), State (stát) a ZipCode (poštovní kód), pak přesně víte, jaký typ hodnoty do každého z polí patří. Rovněž pak bude velmi snadné řadit data podle státu nebo vyhledat všechny lidi, kteří se jmenují Hernandez.

Záznamy

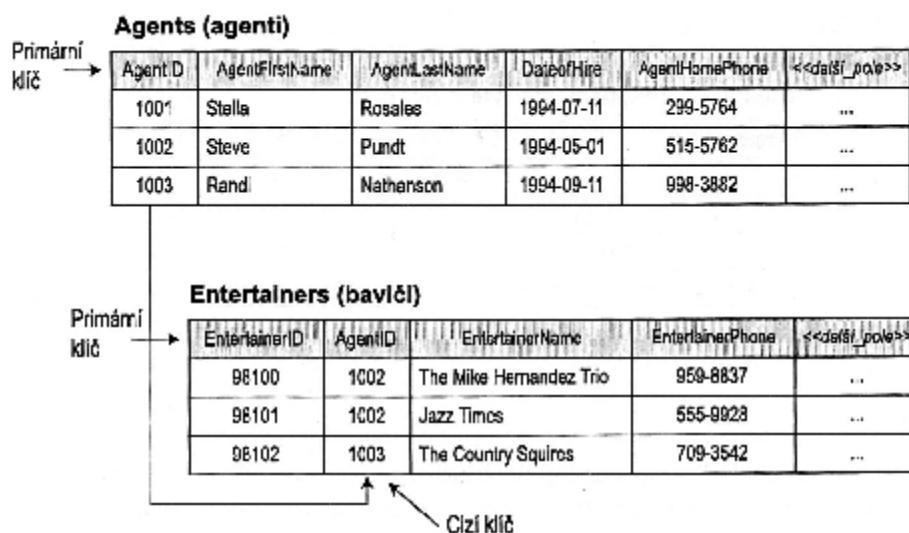
Záznam představuje jedinečnou instanci subjektu tabulky. Je tvořen celou množinou polí v tabulce bez ohledu na to, zda všechna tato pole obsahují zadané hodnoty. Každý záznam je v rámci celé databáze identifikován jednoznačnou hodnotou v poli primárního klíče daného záznamu, což je dáno způsobem definování tabulky. Například na Obrázku 4 představuje každý záznam v tabulce jedinečného zákazníka a pole CustomerID (identifikátor zákazníka) identifikuje příslušného zákazníka v celé databázi. Následně pak každý záznam zahrnuje všechna pole v tabulce a jednotlivá pole popisují nějaký aspekt zákazníka reprezentovaného tímto záznamem. Záznamy jsou klíčovým faktorem k pochopení vztahů mezi tabulkami, protože potřebujete vědět, jak záznam v jedné tabulce souvisí s jinými záznamy v jiné tabulce.

Klíče

Klíče jsou speciální pole, která hrají v tabulce velmi specifické role. Typ klíče určuje jeho smysl v dané tabulce. Třeba, že může každá tabulka obsahovat několik typů klíčů. Ve svém pojednání se omezíme na dva nejdůležitější: primární klíč a cizí klíč.

Primární klíč je skupina polí, která jednoznačně identifikují každý záznam v tabulce (je-li primární klíč tvořen dvěma či více poli, pak se označuje za složený neboli kompozitní primární klíč). Primární klíč je ze dvou důvodů ze všech nejdůležitější: jeho hodnota identifikuje specifický záznam v rámci celé databáze a jeho pole identifikuje pole v rámci celé databáze. Primární klíče rovněž zajišťují integritu na úrovni tabulky a pomáhají při vytváření vztahů z jinými tabulkami. Primární klíč by měla mít každá tabulka ve vaší databázi.

Pole AgentID (identifikátor agenta podle Obrázku 6 je dobrým příkladem primárního klíče, protože jednoznačně identifikuje všechny agenty v tabulce Agents). Pomáhá také zajistit integritu na úrovni tabulky, jelikož neumožní zařazení duplicitních záznamů. Používá se také k vytvoření vztahů mezi tabulkou Agents a dalšími tabulkami v databázi jako je např. tabulka Entertainers (Baviči) rovněž uvedena v příkladu.

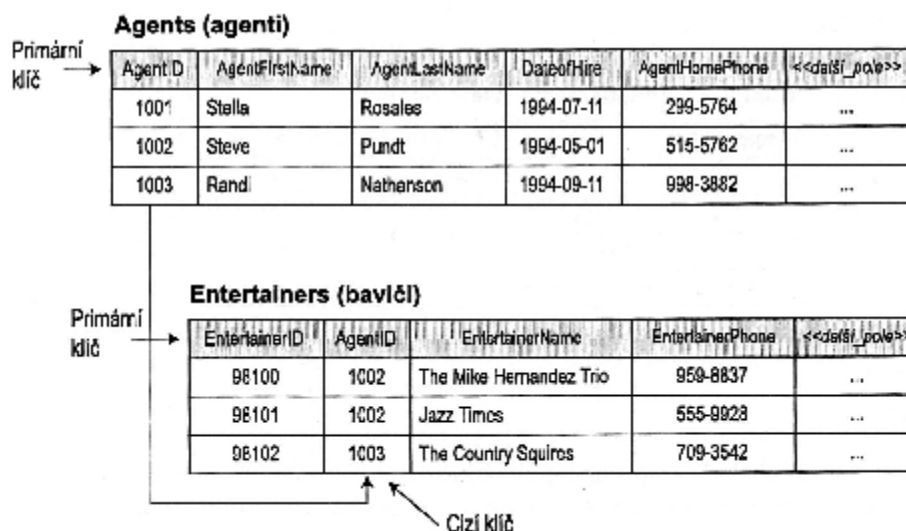


Obrázek 6 Myslíme v jazyku SQL – příklad primárního klíče

Když určíte, že spolu nějaká dvojice tabulek souvisí, pak obvykle vytvoříte příslušný vztah tím způsobem, že vezmete kopii primárního klíče z první tabulky a vložíte do druhé tabulky kde se stane cizí klíčem (termín cizí klíč je odvozen ze skutečnosti, že druhá tabulka

má již svůj vlastní primární klíč a že vámi přidávaný primární klíč první tabulky je pro tu druhou cizí).

Obrázek 7 je dobrým příkladem cizího klíče, zde je AgentID primárním klíčem v tabulce Agents a cizím klíčem v tabulce Entertainers jak je zřejmé, tabulka Entertainers již má svůj primární klíč – EntertainerID (Identifikátor baviče). V tomto vztahu je AgentID polem, které stanovuje spojení mezi tabulkami agentů a bavičů.



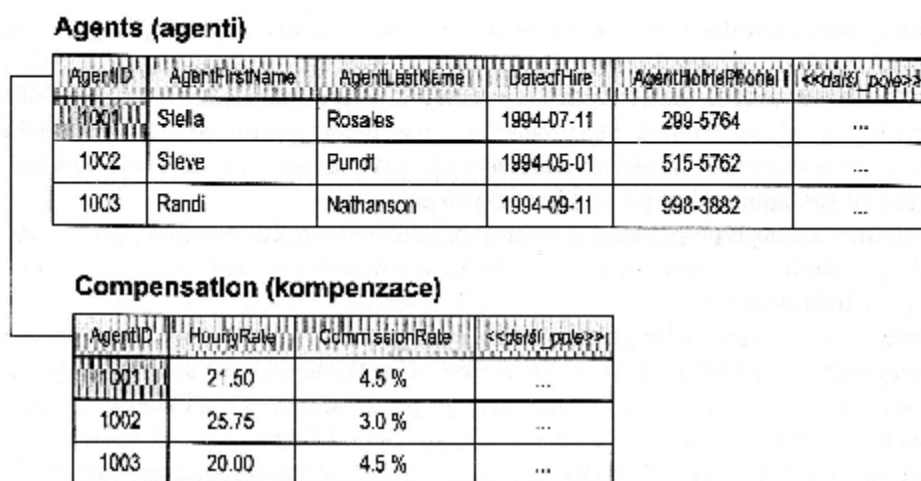
Obrázek 7 Myslíme v jazyku SQL – příklad cizího klíče

Cizí klíče jsou důležité nejen z toho zřejmého důvodu, že pomáhají vytvořit vztahy mezi dvojicemi tabulek, ale i proto, že pomáhají zajištění integrity na úrovni vztahů. To znamená, že záznamy v obou tabulkách budou vždy řádně souviset, jelikož hodnoty cizího klíče musí vycházet z hodnot primárního klíče, na který se odkazují. Cizí klíče rovněž pomáhají v odstranění obávaných „opuštěných záznamů“, jimi jsou v klasickém příkladu záznamy objednávek bez přiřazených klientů. Když nevíte, kdo objednávku uskutečnil, nemůžete ji zpracovat a samozřejmě na ni nemůžete vystavit fakturu.

1.7 Vztahy

Jeden k jednomu

Mezi záznamy dvojice tabulek je vztah jednoho k jednomu, když jediný záznam v první tabulce souvisí jen s jedním záznamem v druhé tabulce a jediný význam v druhé tabulce souvisí pouze s jedním záznamem v první tabulce. V tomto typu vztahu se jedna z tabulek označuje za primární a druhá za sekundární tabulku. Vztah se vytvoří tak, že se vezme primární klíč primární tabulky a vloží se do sekundární tabulky, kde se stane cizím klíčem. To je speciální typ vztahu, protože v mnoha případech slouží cizí klíč také jako primární klíč sekundární tabulky.



Obrázek 8 Myslíme v jazyku SQL – vztah jeden k jednomu

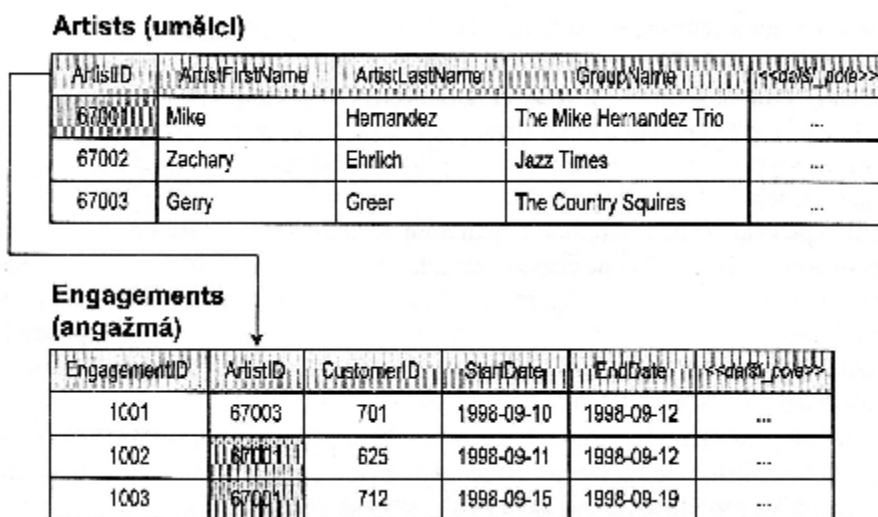
Příklad typického vztahu jednoho k jednomu najdete na Obrázku 8, kde je tabulka Agents primární a tabulka Compensation sekundární. Vztah mezi těmito tabulkami spočívá v tom, že jeden záznam v tabulce Agents lze vztáhnout pouze k jednomu záznamu v tabulce Compensation a že jeden záznam v tabulce Compensation lze vztáhnout pouze k jednomu záznamu v tabulce Agents.

Jeden k více

Když mají záznamy ve dvojici tabulek vztah jednoho k více, pak jeden záznam v první tabulce může mít souvislost s mnoha záznamy v druhé tabulce, ale jeden záznam v druhé tabulce může mít souvislost pouze s jedním záznamem v první tabulce. Tento vztah se vytvoří přenesením primárního klíče tabulky na straně „jeden“ a jeho vložením do tabulky na straně „více“, kde se stává cizím klíčem.

Typický příklad vztahu jednoho k více je uveden na Obrázku 9. V tomto příkladu lze jediný záznam v tabulce Artists vztáhnout k mnoha záznamům v tabulce Engagements, ale

jeden záznam v tabulce Engagements lze vždy vztáhnout jen k jednomu záznamu v tabulce Artists. Jak asi už tušíte, ArtistID je cizí klíč v tabulce Engagements.



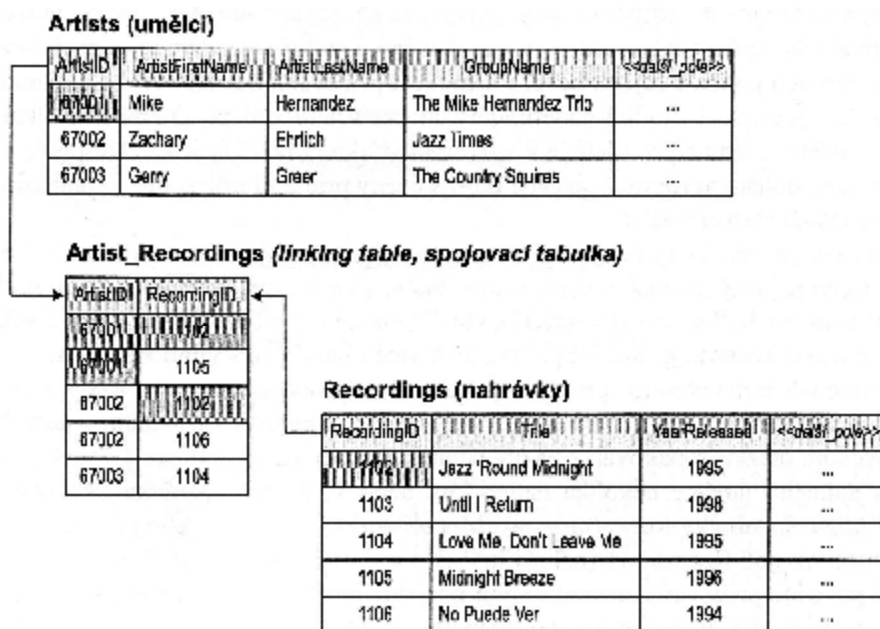
Obrázek 9 Myslíme v jazyku SQL – vztah jeden k více

Více k více

Záznamy dvojice tabulek se vyznačují vztahem více k více, jakmile může být jeden záznam v první tabulce vztažen k více záznamům v druhé tabulce a jeden záznam v druhé tabulce může být vztažen k více záznamům v první tabulce. Aby byl tento vztah ustanoven správně, musíte vytvořit strukturu označovanou za spojovací tabulku. Taková tabulka nabízí jednoduchý způsob přiřazení záznamů z jedné tabulky k záznamům v druhé tabulce, přičemž zároveň pomůže zajistit bezproblémovost přidávání, odstraňování a změnu souvisejících dat. Spojovací tabulku nadefinujete tak, že vezmete kopii primárních klíčů obou tabulek daného vztahu a použijete je k vytvoření struktury nové tabulky. Tato pole vlastně slouží dvěma odlišným rolím: Společně tvoří složený primární klíč spojovací tabulky a samostatně slouží jako cizí klíče. Vytvořením a používáním spojovací tabulky dokážete správně řešit vztahy více k více. Obrázek 10 ukazuje příslušné řešení v praxi.

Spojovací tabulka na Obrázku 10 byla vytvořena tak, že jsme vzali pole ArtistID z tabulky Artists a pole RecordingId z tabulky Recordings a použili je jako základ pro novou tabulku. Stejně jako jiné tabulky v databázi má i nová spojovací tabulka svůj název – Artist_Recordings (nahrávky umělců). Skutečná výhoda spojovací tabulky spočívá v tom, že vám umožňuje asociovat libovolný počet záznamů obou tabulek ve vztahu. Jak ukazuje

náš, příklad můžete nyní snadno přiřadit k danému umělci libovolný počet nahrávek nebo specifické nahrávce libovolný počet umělců. [2]



Obrázek 10 Myslíme v jazyku SQL – vztah více k více

1.8 PHP

1.8.1 Historie PHP

Počátky PHP se datují rokem 1995, kdy jeden nezávislý dodavatel vyvíjející software jménem Rasmus Lerdorf vyvinul jistý skript Perl/CGI, který mu umožňoval zjistit, kolik návštěvníků četlo jeho online resumé. Jeho skript dělal dvě věci: protokoloval informace a návštěvníkovi a zobrazoval na webové stránce počet návštěvníků. Protože web, jak ho známe dnes, byl tehdy ještě v plenkách, neexistovaly nástroje, jako je tento, takže vyvolal záplavu e-mailů se žádostmi o Leodorfovy skripty. Leodorf proto začal dávat k dispozici svou sadu nástrojů, a opatřil ji titulem Personál Home Page (PHP).

Halasné dožadování se sady nástrojů PHP přimělo Leodorfa, aby začal vyvíjet dodatky k PHP. Jeden z nich měl převádět data zadaná do formuláře HTML do symbolických proměnných, aby je uživatelé mohli exportovat jiné systémy. Aby toho dosáhl, pokračoval ve vývoji kódy C a Perl opustil. Průběžné přidávání dodatků do sady nástrojů PHP kulminovalo v listopadu 1997 vydáním PHP 2.0, neboli Personál Home Page Form Interpreter (PHP-FI). Důsledkem trvale rostoucí popularity PHP bylo, že verze 2.0 byla

provázena řadou všelijakých zobecnění a zdokonalení od programátorů po celém světě. Nové vydání PHP bylo neobyčejně populární a ke klíčovému týmu vývojářů se brzy připojil i Leodorf. Ten udržel původní myšlenku začleňování kódu přímo do HTML a přepsal „analyzující motor“, neboli parsing engine, čímž přišel na svět PHP 3. Podle vydání verze 3.0 z června 1998 využívalo už PHP přes 50 000 uživatelů, aby obohatili své webové stránky.

Po dobu dalších dvou let pokračoval vývoj hektickým tempem. Byly přidány stovky funkcí a počet uživatelů rostl mílovými kroky. Na začátku roku 1999 oznámila společnost Netcraft, že střízlivý odhad uživatelské základny přesahuje 1 000 000, takže se PHP stal jeden z nejpůvodnějších skriptovacích jazyků na celém světě. Jeho popularita překročila i ta největší očekávání vývojářů. Brzy bylo zřejmé, že uživatelé hodlají jazykem PHP vybavovat daleko větší paletu aplikací, než se původně předpokládalo. Iniciativu převzali dva klíčoví vývojáři, kteří byli v čele procesu, při němž se kompletně znovu promýšlelo, jak doposud PHP fungoval, což kulminovalo tím, že přepsali parser PHP a dali mu jméno Zend scripting engine. Výsledky této práce jsme viděli ve vydání PHP 4.^[3]

Proč používat PHP?

Jazyk PHP je jedním z nejlepších kompromisů mezi tím, jak snadno jej můžou používat začátečníci a jak širokou škálu možností využití nabízí zkušeným programátorům. K přednostem technologie PHP patří výkon, stabilita, přenositelnost, těsná integrace s většinou databázových systémů a téměř neomezené možnosti rozšiřování – to vše prakticky zdarma (PHP je dodáván s veřejným zdrojovým kódem).

Jak PHP Funguje

PHP je jazykem určeným pro skriptování na straně serveru – tzn. kód, který je napsán v jazyce PHP je uložen na počítači nabízejícím webové stránky návštěvníkům (lze je prohlížet prostřednictvím webových prohlížečů).

Navštíví-li uživatel webovou prezentaci napsanou v jazyce PHP, načte server kód PHP a zpracuje ho podle zadaných parametrů. Server pak data odešle webovému prohlížeči, který pak s daty pracuje jako se standardními stránkami HTML. Tím se liší dynamická prezentace od statické, v níž na požádání server pouze odešle data HTML do webového prohlížeče – na straně serveru nedochází k žádnému zpracování.

1.9 MySQL

MySQL je nejoblíbenější databázový systém s veřejným zdrojovým kódem. Je velice vyrovnaným konkurentem drahých systémů jako jsou Oracle, nebo Microsoft SQL Server. MySQL byla vyvinuta a je udržovaná švédskou společností MySQL AB.

Jde o systém správy databází určený pro relační databáze. Relační databáze je kolekcí vzájemně provázaných dat – používá k ukládání informací více tabulek, čímž je informace dělená na nejmenší součásti.

Software MySQL se skládá z několika částí:

- server MySQL (démon mysqld) – spouští a udržuje databáze
- klient MySQL (program mysql) – nabízí rozhraní pro správu serveru
- další nástroje k údržbě a jiným účelům

MySQL může obsahovat na 60 000 tabulek s více než 5 miliardami řádků (cca 4 GB).

Jak funguje propojení PHP+MySQL?

Začleněním databáze do webové aplikace lze určitá data generovaná PHP načítat z databáze MySQL. Obsah stránek přestává být statický (napevno zapsaný) a stává se flexibilním. To je pak klíčem k dynamickým webovým prezentacím. Oficiální webovou aplikaci lze najít na stránkách <http://www.mysql.com>.

Potřebné softwarové vybavení

Abych mohli vyzkoušet práci se síťovými verzemi databázových systémů bez ohledu na používaný operační systém, budeme k tomu potřebovat:

1. software webového serveru (např. Apache, Xitami, IIS)
2. PHP
3. MySQL
4. Webový prohlížeč – (Internet Explorer, Netscape Navigátor, Mozilla, Opera ...)
5. textový editor, nebo integrované vývojové prostředí

Je dobré, že vše se může pořídit prakticky zadarmo – Apache, PHP i MySQL jsou produkty s veřejně přístupným kódem – open source. Máme-li počítač, stačí pouze stáhnout nezbytný software z internetu a nainstalovat jej dle předpisu. Počítač pak může sloužit jako klient a jednak jako server. Je možné si předplatit službu webhosting (využívání prostoru na

serveru poskytovatele). Při výběru poskytovatele je důležité, zda lze na daném serveru spouštět aplikace PHP a využívat databáze MySQL. ^[4]

II. PRAKTICKÁ ČÁST

2 ANALÝZA SOUČASNÉHO STAVU SYSTÉMU

Firma Jinlong s.r.o. založila svůj webový server jako čistě informační. V té době se vývojář zaměřil spíše na grafické provedení webového serveru a vlastní funkčnost a administraci nedomyslel. Jelikož je celý webový server založen na technologii Flash je v praxi velice obtížný a pracný jeho update. Pro samotného vývojáře znamená každá změna obsahu webového serveru spuštění programu Flash, předělání potřebných údajů pomocí složitých grafických operací, jeho uložení a opětovné nahrání na server.

V dnešní době se situace poněkud změnila a firma Jinlong s.r.o. se rozhodla, že chce svůj webový server více využívat a pravidelně updatovat jeho obsah. Nastal velký problém, jelikož původní designér a programátor stránek nezanechal potřebné soubory, které se používají při úpravě.

První problém může nastat pokud uživatel nemá nainstalovaný Flash Player, který se používá při zobrazování Flash stránek na webovém serveru. Jakmile se něco takového stane uživateli se stránky nezobrazí a uvidí pouze černou obrazovku. Tento problém je docela závažný hlavně kvůli nedostatečným znalostem uživatelů, které nenapadne případné nainstalování již zmíněného Flash Playeru

Pokud má uživatel vše potřebné na prohlížení webového serveru, dostane se k hlavní stránce která je zobrazena na Obrázku 11.

Webový server je rozdělen do pěti částí:

- restaurace
- bar
- fitness
- bowling
- squash

Každá sekce má v hlavním menu svoje logo. Po kliknutí na příslušné logo vyjede podmenu v kterém si uživatel vybírá další specifikace pro danou sekci.



Obrázek 11 Úvodní stránka Jinlong Sport Clubu

2.1 Jednotlivé sekce webového serveru

2.1.1 Restaurace

Sekce restaurace obsahuje následující prvky:

- **Informace** – základní informace o restauraci
- **Nealkoholické nápoje** – seznam nealkoholických nápojů
- **Alkoholické nápoje** – seznam alkoholických nápojů
- **Jídelní lístek** – nabídka jídel, salátů a dalších pokrmů
- **Foto galerie** – fotky restaurace a interiéru

V malém rozboru probereme jednotlivé sekce a ujasníme si jejich nedostatky. První stránka, která se zobrazí po kliknutí na logo restaurace je čistě informační. Pokud se zaměříme na její obsah je dostatečně obsáhlý a obsahuje všechno potřebné. Problém je ovšem v organizaci stránky, která nám neumožní zobrazení celého textu. Uživatel musí zbytečně klikat na posunutí stránky k dosažení dalších informací, které by měli být patrné na první pohled. Další zásadní věc je barevné oddělení textu. Pokud se v textu objevují věci, které by měly být patrné ihned, měli bychom je uživateli nabídnout tím, že jim dáme jinou barvu.

Sekce nealkoholické nápoje, alkoholické nápoje a jídelní lístek jsou na první pohled velice nepřehledné. Jelikož je seznam položek dlouhý uživatel musí stále klikat k jeho posunutí a tím se stává velice nepřehledným. Chybí také filtrování, které by daný problém efektivně vyřešilo. Uživatel by si vybral například z nabídky jídelního lístku kuřecí pokrmy a po jednom kliknutí by se zobrazil obsah požadovaného dotazu.

Foto galerie je řešena pomocí programu Zoner Photo Studio 9, a proto je její update velice složitý. Pokud chceme vložit další fotky musíme si nechat galerii znovu vygenerovat a nahrát na server. Zásadní nedostatek galerie je, že fotky neobsahují žádné informace a uživateli není ihned jasné co na fotce vlastně je. Tento problém by vyřešila jednoduchá databáze fotek, do které bychom mohli bez problémů fotky přidávat a ke každé fotografii napsat i krátký komentář.

2.1.2 Bar

Sekce bar obsahuje následující prvky:

- **Informace** – základní informace o baru
- **Akce** – seznam akcí, které v baru probíhají
- **Foto galerie** – fotky baru
- **Party list** – měsíční přehled akcí v baru
- **Party fotky** – fotografie z diskoték, rockoték, karnevalů atd.

V sekci bar bych se zmínil hlavně o prvku party list a party fotky, které jsou vyřešeny špatným způsobem. Ostatní odkazy a jejich obsah jsou stejné jako v předchozím případě a proto je zbytečné analyzovat opětovný problém.

Při kliknutí na odkaz party list jsme přeměrování na stránku vygenerovanou programem FreeWebTemplates.com. Na první pohled nás zastraší příšerné grafické provedení, které je velice nepřehledné a tabulka která by měla znázorňovat seznam plánovaných akcí na daný měsíc se v něm poněkud ztrácí. Na Obrázku 12 můžeme vidět provedení stránky. Zásadní problém je v nepřehlednosti seznamu a jeho neúplnosti. Chybí informace o vstupném a jednotlivé sloupce nejsou zřetelně označeny. Potřebné by bylo také filtrování jednotlivých záznamů v tabulce. Uživatel by si mohl např. zobrazit party na kterých hraje pouze DJ PeeJay.

Program Duben 2008		
11.4.2008	DJ Bouy	clubový večer a klubová hudba
12.4.2008	DJ Petr Ja. uť	Club music
18.4.2008	DJ PeeJay	Fireparty live
19.4.2008	DJ Smith - DJ Wano - DJ Altra	Pop vs. Hip Hop vs. R&B
25.4.2008	DJ Denge	Mixed bag! Anything
26.4.2008	DJ Jabko	New metal, Rock, Pop, metal

Copyright 2008. All rights reserved. Design developed by FreeWebTemplates.com. Free web design, web templates, web layouts, and website resources.

Obrázek 12 Party list

Galerie s fotkami je řešena opět programem Zoner Photo Studio 9. V tomto případě se ale nejedná o závažnou věc, jelikož se fotky do galerií nepřidávají a proto stačí galerii vygenerovat pouze jednou a nahrát na server. Jediný problém je, že se data nedají filtrovat a při tak velkém množství dat je to zásadní.

2.1.3 Fitness, squash, bowling

Sekce obsahují následující prvky:

- **Informace** – základní informace o možnostech zábavy
- **Ceník** – ceník zmíněných služeb
- **Fotogalerie** – galerie fotek

V následujících sekcích se pokračuje stejným způsobem jako v předchozích. Špatné barevné oddělení důležitých částí obsahu, strohé a základní informace. Galerie fotografií je řešena přímo ve Flashi a proto je její úroveň větší než galerie v Zoner Photo Studiu.

Celý webový server je špatně rozvržený. Po grafické stránce není špatný, ale funkčně se nedostáváme ani na základní funkce optimalizovaného a funkčního webu. V novém návrhu se zaměříme zejména na celkovou funkčnost webového serveru, jeho optimalizaci pro webové vyhledávače díky kterým se webová prezentace dostane na přední příčky ve vyhledávačích. Celkově se budeme snažit udělat web přehlednějším pro uživatele, výkonnějším a lépe ovladatelným.

3 NÁVRH NOVÉHO WEBOVÉHO SERVERU

Jedním z nejdůležitějších faktorů, při tvorbě kvalitní webové prezentace je její prvotní analýza. V takové analýze se zaměříme na cíle, které by měl web splnit, nalezneme vhodná klíčová slova k optimalizacím pro webové vyhledávače, navrhne strukturu databází, vytvoříme grafickou podobu.

Jelikož se jedná o větší informační systém bude kladen důraz na jeho snadné ovládání, přehlednost, dobře orientovaný a kvalitní obsah a také na optimalizaci.

3.1 Grafický návrh

První věc při vytváření webové prezentace je grafická podoba. Programátor by měl mít určitou představu o rozvržení webu (jak rozmístit odkazy, jaké zvolit barvy atd.).

Při rozhodování jsem bral v potaz dva zásadní faktory. Umístění menu tak aby bylo přehledné a také aby obsahovalo vše potřebné. Ve webdesignu se zásadně používá menu v levé části (odkazy pod sebou) a nebo těsně pod hlavním bannerem (odkazy vedle sebe). Pro rozsáhlejší web je velice vhodné volit variantu menu v levé části a to hlavně z důvodu velkého množství odkazů.

Celkový design webové prezentace bude řešen pomocí Css a jednoho banneru, který bude umístěn v horní části stránky. Při vytváření grafické podoby banneru byly zohledněny tato faktory: barevné provedení, obsah a velikost.

Barevné provedení: jelikož má firma Jinlong s.r.o. ve svých barvách modrou a žlutou volil jsem barevné provedení následující. Název firmy Jinlong žlutou barvou, sport club modrou barvou. Zbývající text bude také žlutou barvou.

Obsah: obsah je velice důležitý jelikož značí uživateli co na stránkách může najít a na co jsou zaměřené. V případě stránek Jinlong club jsou to zajiště služby fitness, bowling, squash a bar proto jsem v grafice použil čtyři základní fotky, které jsem rozvrhnul podle důležitosti a ke každé jsem žlutě napsal o kterou část jde. Vzniknul efektivní banner, který určitě zaujme a uživatel ihned ví jak jsou stránky orientované.

Velikost: velikost je zvolena 980x280 px. Hlavní důvod je aby se stránky dobře zobrazovali na monitorech v rozlišení 1024x768.

Hlavní část stránky jsem v css kódu zapsal následovně:

```
.banner{  
  
    width:980px;                //šířka 980px  
  
    height:280px;              //výška 280px  
  
    background-image: url("banner.jpg"); //url obrázku  
  
    background-repeat: no-repeat; //neopakovat obrázek  
  
    border-left-style:solid;     //rámeček levá strana  
  
    border-right-style:solid;    //rámeček pravá strana  
  
    border-top-style:solid;      //rámeček horní strana  
  
    border-color:#FFFFFF;       //barva rámečku bílá  
  
    border-width:1px;           //tloušťka rámečku 1px  
  
}
```

V druhé části jsem se zaměřil na pozici menu a celkové main stránky. Ihned pod hlavním bannerem začíná main stránka, která má následující css prvky:

```
.main{  
  
    background-color:#000000;    //barva pozadí černá  
  
    text-align:left;             //zarovnání textu vlevo  
  
    font-family:Verdana;        //font Verdana  
  
    font-size:12px;             //velikost písma 12px  
  
    color:#FFFFFF;              //barva písma bílá  
  
    height:560px;               //výška 560px  
  
    width:980px;                //šířka stejná jako je banner 980px  
  
    border-left-style:solid;     //rámeček levá strana  
  
    border-right-style:solid;    //rámeček pravá strana  
  
    border-color:#FFFFFF;       //barva rámečku bílá  
  
    border-width:1px;           //tloušťka rámečku 1px
```

```
}
```

Hlavní main stránka se dále člení na další podsekce a těmi jsou main_left. Tato část bude obsahovat menu stránky. V našem případě se jedná o jednu z nejdůležitějších částí webové prezentace.

Zápis css kódu je následující:

```
.main_left{  
  
    background-color:#000000;           //barva pozadí černá  
  
    font-family:Verdana;                //font Verdana  
  
    font-size:12px;                     //velikost písma 12px  
  
    color:#FFFFFF;                       //barva bílá  
  
    height:560px;                        //výška 560px  
  
    width:150px;                          //šířka 150px nám postačí na všechny odkazy  
  
    float:left;                           //zarovnání bloku v levé části main  
  
    margin-right:15px;                    //odsazení od pravého ohraje 15px  
  
    border-right-style:solid;             //rámeček na pravé straně  
  
    border-color:#FFFFFF;                 //barva rámečku bílá  
  
    border-width:1px;                     //šířka rámečku 1px  
  
}
```

Jelikož bude menu obsáhlé musí se zajistit určitá efektivita odkazů tzn. musí být patrné co je odkaz a co ne. Hlavní sekce jsem nazval sport bar, restaurace, bowling, squash, fitness a solárium. Pro tyto názvy jsem volil vlastní nadpis H1 bílé barvy velikosti 16px, který efektivně oddělí normální text od odkazů. Pro odkazy jsem volil žlutou barvu. Každý odkaz jsem ještě oddělil bílou čarou pro větší přehlednost.

Css kód pro menu:

```
.menu{  
  
    background-color:#000000;  
  
    border-bottom-style:solid;
```

```
border-color:#FFFFFF;  
  
border-width:1px;  
  
}
```

Css kód pro odkazy:

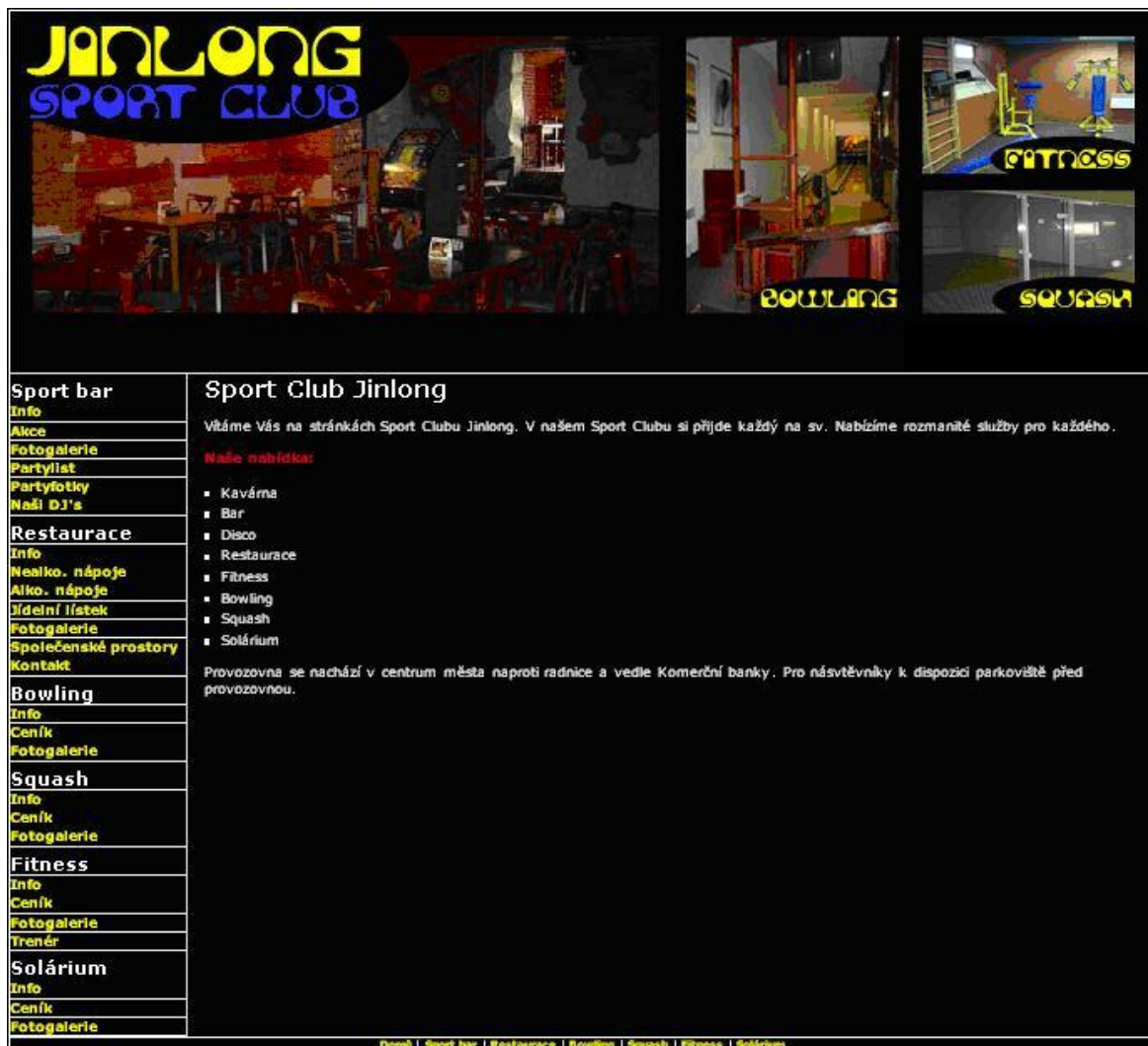
```
A:link {text-decoration:none; color:#FFFC00; font-weight:bold;}  
  
//žádné podtržení, žlutá barva písma, tučně  
  
A:visited {text-decoration:none; color:#FFFC00; font-weight:bold;}  
  
// žádné podtržení, žlutá barva písma, tučně  
  
A:hover { text-decoration:none; color:#3530F9; font-weight:bold;}  
  
//žádné podtržení, modrá žlutá barva písma, tučně
```

Poslední věc, kterou jsem při vytváření grafiky udělal je patička stránky, která bude obsahovat odkazy na jednotlivé sekce. Hlavní důvod, proč jsem umístil odkazy i do spodní lišty je určitá optimalizace webu a také větší přehlednost.

Css kód pro patičku:

```
.main_bottle{  
  
background-color:#000000;           //barva pozadí černá  
  
font-family:Verdana;               //font Verdana  
  
font-size:8px;                     //velikost písma 8px  
  
color:#FFFFFF;                     //barva písma bílá  
  
text-align:center;                 //zarovnání na střed  
  
height:10px;                       //výška 10px  
  
width:980px;                       //šířka 980px  
  
border-style:solid;                //celý rámeček  
  
border-color:#FFFFFF;              //barva rámečku bílá  
  
border-width:1px;                  //tloušťka rámečku 1px  
  
}
```

Při zapsání kódu do stránek a naplnění menu dostáváme celou grafickou podobu stránek, kterou budeme dále používat pro další podstránky. Ukázka grafické podoby je na Obrázku 13.



Obrázek 13 Grafická podoba stránky

Složení menu:

- **Sport bar** – Info, Akce, Fotogalerie, Partylist, Partyfotky, Naši DJ's
- **Restaurace** – Info, Nealko nápoje, Alko nápoje, Jídelní lístek, Fotogalerie, Společenské prostory, Kontakt
- **Bowling** – Info, Ceník, Fotogalerie
- **Squash** – Info, Ceník, Fotogalerie
- **Fitness** – Info, Ceník, Fotogalerie, Trenér
- **Solárium** – Info, Ceník, Fotogalerie

3.2 Struktura a návrh databází

Jelikož se jedná o poměrně velký webový projekt je zapotřebí, aby se nejdůležitější obsah dal editovat a efektivně přidávat nebo ubírat data. Při navrhování potřebných databází si vždy kladu otázku co v databázi bude uloženo. Když se bude jednat o fotogalerii určité bude databáze obsahovat informace o fotce. To v předchozím případě většinou chybělo a uživatel mohl tápat co vlastně na obrázku je. Pokud bude databáze zaměřená spíše na informace musíme přesně vědět jaké informace budeme potřebovat a jaké elementy tabulky v databázi vytvořit. Název databáze jsem volil jinlong_club v níž se dále budou nacházet tabulky pro jednotlivé využití.

Vytvoření databáze v MySql console:

Využíváme příkazu `create database nazev-database;`. V naše případě napíšeme příkaz `create database jinlong_club;`. Středník je nezbytný bez něj by se příkaz neprovedl. Po vytvoření musíme databázi aktivovat pomocí příkazu `use jinlong_club.`

3.2.1 Sekce Sport Bar

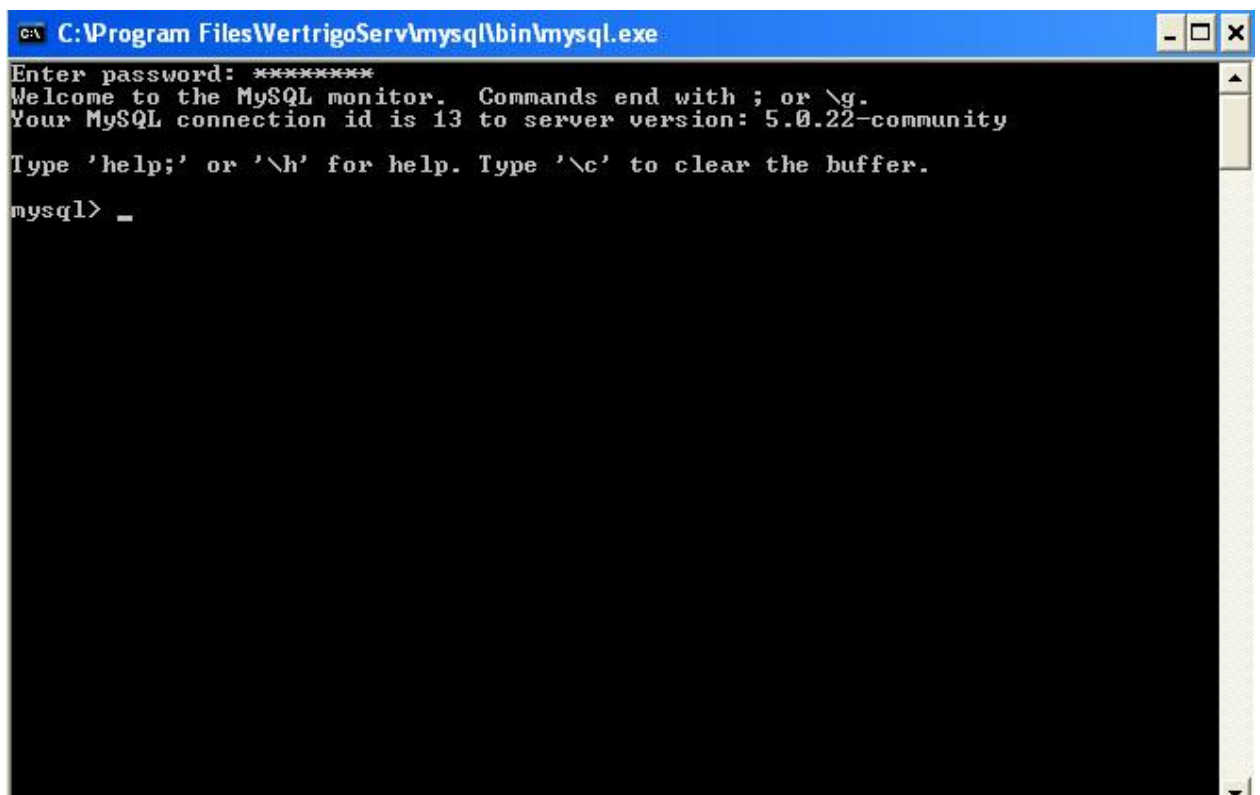
1. Tabulky v sekci – Sport bar

Sekce sport bar je složená z šesti odkazů: info, akce, fotogalerie, Partylist, Partyfotky, Naši DJ's. Ze zmíněných šesti odkazů budou čtyři využívat databázi MySql. První záložka info databázi nebude využívat. Hlavní důvod je, že obsah odkazu bude statický a nebude se měnit, jelikož se bude jednat o základní a neměnné informace. Uživatel v nich může najít nabídku služeb a také otevírací dobu. Stejně platí i pro záložku akce, v které návštěvník nalezne informace o možnostech využití prostor sport baru jinlong.

První tabulka, která bude vytvořena ponese název fotogalerie. Tabulka bude využívána pro všechny fotografie, které budou zobrazovat prostory v jednotlivých sekcích. Hlavní důvod, proč udělat pouze jednu tabulku pro všechny sekce je jednoduchý. MySQL umožňuje velké množství řádků v tabulce a jelikož můžeme využít efektivního třídění dat z jedné tabulky vybereme podle svého. Všechny úkony, které s databází provádíme můžeme dělat také v PhpMyAdmin, ale pro naučení a vysvětlení budeme používat MySQL consoli (na Obrázku 14 můžeme vidět prostředí PhpMyAdmin a na Obrázku 15. prostředí MySQL consoli).



Obrázek 14 Prostředí PhpMyAdmin



Obrázek 15 Prostředí MySQL console

Tabulka fotogalerie bude obsahovat následující názvy sloupců:

- **id_fotky** – první a nejdůležitější položka tabulky id_fotky bude také primární klíč tabulky. Pro efektivní přidávání využijeme funkce auto increment, která nám bude generovat číselné označení
- **popis_fotky** – základní popis fotografie
- **kategorie** – sloupec kategorie je také velice důležitý a to hlavně z důvodu dalšího třídění dat z tabulky pro zvolené kategorie
- **url_fotky** – název fotografie

Tabulku vytvoříme pomocí následujícího příkazu:

```
Create table fotogalerie(
```

```
    id_fotky int auto_increment primary key,    //primární klíč, auto increment
```

```
    popis_fotky varchar(100),                //pole znaků o velikosti 100
```

```
    kategorie varchar(100),                  //pole znaků o velikosti 100
```

```
    url varchar(100)                          //pole znaků o velikosti 100
```

```
);
```

Nesmíme zapomínat, že na každém řádku musí být čárka a v posledním přidaném sloupci závorka a středník. Čárka se již nepíše.

Tabulka party list bude obsahovat tyto sloupce:

- **id_party** – v tomto případě nebudeme využívat položky auto increment, ale administrátor bude ID čísla zadávat ručně.
- **id_dj** – identifikační číslo příslušného DJe, který bude na party hrát
- **datum_akce** – datum kdy se akce uskuteční
- **dj** – jméno DJe, který bude hrát
- **informace** – základní informace o akci (jméno skupiny, hudebná styp atd.)
- **vstup** – informace o vstupném

Tabulku vytvoříme pomocí následujících příkazů:

```
Create table party_list(
```

```

id_party int primary key,      //int a primární klíč
id_dj int,                     //int
datum_akce date,              //speciální proměnná date se používá pro datum
dj varchar(20),                //pole znaků o velikosti 20
informace varchar(256),        //pole znaků o velikosti 256
vstup varchar(30)              //pole znaků o velikosti 30
);

```

Tabulku party list bude velice často editována administrátorem nebo jinou pověřenou osobou a proto bude v administračním prostředí možnost jednoduché práce s tabulkou jako je mazání údajů, editace atd.

Tabulka party fotky bude obsahovat tyto sloupce:

- **id_fotky** – identifikátor jednotlivých fotek. Budeme využívat možnosti auto increment
- **popis_fotky** – jednoduchý popis fotografie
- **nazev_akce** – příslušný název pořádané akce
- **datum_akce** – datum, kdy se akce pořádala
- **url_fotky** – název fotografie

Tabulka party fotky bude velice obsáhla jelikož bude obsahovat všechny fotografie. Způsob řešení je v našem případě nejvhodnější, jelikož je zbytečně složité dělat pro každou novou akci speciální databázi, která by obsahovala jen určité fotografie. Pomocí výběrů se efektivně dostaneme k požadovaným datům.

Tabulku vytvoříme pomocí následujících příkazů:

Create table party_fotky (

```

id_fotky int primary key auto increment,      //int primární klíč a auto increment
popis_fotky varchar(256),                     //pole znaků o velikosti 256
nazev_akce varchar(160),                      //pole znaků o velikosti 160
datum_akce date,                              //speciální proměnná date

```

```
url_fotky varchar(100) //pole znaků o velikosti 100  
);
```

Tabulka dj bude obsahovat následující sloupce:

- **id_dj** – identifikátor pro každého DJe
- **jmeno** – příslušné jméno
- **fotka** – url fotografie
- **dance** – informace o zaměření DJe (pole bude obsahovat pouze ano-ne)
- **techno** – informace o zaměření DJe (pole bude obsahovat pouze ano-ne)
- **rock** – informace o zaměření DJe (pole bude obsahovat pouze ano-ne)
- **rnb** – informace o zaměření DJe (pole bude obsahovat pouze ano-ne)
- **hip_hop** – informace o zaměření DJe (pole bude obsahovat pouze ano-ne)
- **oldies** - informace o zaměření DJe (pole bude obsahovat pouze ano-ne)

Tabulka dj je udělaná čistě pro přehled a prezentuje nabídku DJ's v klubu. Databáze byla použita z důvodu rychlé editace a přidání dalšího umělce.

Sekce sport bar je asi nejrozsáhlejší část celého systému a její aktualizace bude velice častá, proto bylo zvoleno větší množství velice výhodných databází, které administrátorům ušetří spoustu práce díky jednoduché editaci v prostředí pro správu jednotlivých tabulek.

Uživatelé si také přijdou na své jelikož budou moci pomocí jednoduchých výběrů zobrazit potřebná vyfiltrovaná data. Výběr dat podle určitého filtru je velice důležitý u rozsáhlejších databází, hlavně kvůli uživatelům.

3.2.2 Sekce restaurace

Sekce restaurace bude také poměrně obsáhlá a bude obsahovat čtyři databáze. Hlavní důvod proč se bude pracovat se čtyřmi databázemi je jasný. Jelikož se bude jednat o větší množství dat, které se budou často upgradovat a mazat je nutno aby tyto údaje byly snadno editovatelné. Tabulky v sekci restaurace budou věnované hlavně jídelnímu lístku a nápojovému lístku a proto se budou jmenovat následovně: nealko_napojje, alko_napojje, jidelni_listek a jako čtvrtá tabulka bude databáze fotogalerie, která jak jsem se již jednou zmínil slouží pro všechny kategorie.

Tabulka nealko_napojje bude obsahovat následující sloupce:

- **id_napojje** – identifikátor pro každý nápoj
- **mnozstvi** – bude určovat množství nápoje v jednotkách
- **nazev_napojje** – název nápoje popř. jeho stručný popis
- **cena** – cena pro každý nápoj
- **kategorie** – kategorie nápojů

Tabulka nealko_napojje bude obsahovat poměrně velké množství dat, které se budou často aktualizovat a to hlavně z důvodu změny cen, proto musí být tabulka flexibilní a snadno editovatelná v administrátorském prostředí, které bude nabízet veškeré změny a mazání údajů podle přání administrátora či jiného uživatele k těmto úkonům pověřeným. Stejně funkce se budou dále uplatňovat v tabulce alko_napojje a jidelni_listek.

Všechny tabulky budou mít jednoduché filtry, které zajistí uživateli bezproblémové vyfiltrování údajů podle jeho potřeb.

Tabulka alko_napojje bude obsahovat následující sloupce:

- **id_napojje** – identifikátor pro každý nápoj
- **mnozstvi** – bude určovat množství nápoje v jednotkách
- **nazev_napojje** – název nápoje popř. jeho stručný popis
- **cena** – cena pro každý nápoj
- **kategorie** – kategorie nápojů

Tabulku vytvoříme pomocí následujících příkazů:

```
Create table alko_napojje(                                     //popř. nealko_napojje
    id_napojje int primary key auto increment,                //int primární klíč a auto increment
    mnozstvi varchar(100),                                    //pole znaků o velikosti 100
    nazev_napojje varchar(260),                               //pole znaků o velikosti 260
    cena int,                                                 //proměnná int
    kategorie varchar(100)                                    //pole znaků o velikosti 100
);
```

Tabulka `jidelni_listek` bude nepatrně pozměněná a to hlavně v přidání jednoho sloupce složení jídla, který bude specifikovat ingredience a složení pokrmu. Položka je nezbytná pro orientaci zákazníka, který bude ihned vědět jakých ingrediencí bylo použito.

3.2.3 Sekce bowling, squash, fitness a solárium

V následujících sekcích se bude objevovat jen jedna tabulka fotogalerie, která je pro všechny sekce stejná a bude obsahovat fotografie jednotlivých částí centra. Kód pro vytvoření tabulky jsme si již ukázali a proto není nutné se k problému více vracet. Ostatní odkazy budou čistě statické a neměnné.

ZÁVĚR

Úkolem této bakalářské práce bylo úkolem analyzovat stávající webový server a následné vytvoření nového informačního serveru pro firmu Jinlong s.r.o. pomocí www stránek a databázového systému MySQL.

Grafická podoba nových webových stránek byla vytvořena pomocí programu Adobe Photoshop a CSS. Jelikož firma požadovala efektivní webový server bylo zapotřebí některá data vložit do databáze. Pro náš případ bylo nejlepší využít databázi MySQL, která skvěle spolupracuje s programovacím jazykem PHP.

Samotná praktická část bakalářská práce je složena ze dvou částí. V první se analyzuje současný webový server firmy Jinlong s.r.o. a v druhém je návrh a řešení nového lepšího systému. Předchozí verze informačního systému nespĺňovala požadavky, které firma očekávala a proto se musel nahradit novým systémem. Hlavní nedostatky byly patrné na první pohled, složité menu, které nutilo uživatele stále klikat než se k požadované informaci dostali, neúplné a špatné informace o službách, špatná administrace a velice nepřehledné rozhraní pro nápojový a jídelní lístek. Dále chybělo určité filtrování dat podle různých parametrů, které uživatel vybral.

Nový systém byl vytvořen podle pravidel webdesignu a nabízí uživateli efektivní hledání všech potřebných informací. Hlavní menu je přehledné a uživatel ihned vidí co vše na stránkách může naléznout. Jídelní lístek a nápojový lístek je efektivně vložený do databáze MySQL a proto si může uživatel zobrazit jen jídla, na které má chuť nebo které ho lákají. Všechny potřebné informace jsou viditelné a odlišené různými barvami textu popřípadě odrážkami. Největší výhodou je administrační prostředí v kterém může administrátor jednoduše měnit údaje v tabulkách, přidávat či mazat data. Nový webový server je na první pohled efektivnější a praktičtější. Bohužel jeho aplikace na internet není možná jelikož firma Jinlong s.r.o. prozatím nezajistila potřebný webhosting s podporou PHP a MySQL.

ZÁVĚR V ANGLIČTINĚ

The task of the bachelor work was to analyze the current web server and to create sequentially a new information server for the firm Jinlong s.r.o. by means of the www sites and the database system MySQL.

The graphic form of the new web sites was established by means of the programme Adobe Photoshop and CSS. Because the firm required an effective web server, it was need to input some data into the database. For our case it was the best to use the database MySQL, which cooperates greatly with the programming language PHP.

The separate practical segment of the bachelor work is composited of two parts. The first one analyzes the contemporary web server of the firm Jinlong s.r.o. and the second one includes the concept and the solution of the new better system. The previous version of the information system didn't fulfill requirements, which the firm expected, and that's why it had to be replaced by a new system. The main lacks were noticeable at first sight, a complicated menu, which pressed the user to click still to come to the demanded information, incompleted and bad information about services, bad administration and very unnoticed interface for the list of drinks and the menu card. Further it was missing a certain data filtration by various parameters, which the user chose.

The new system was created under the rules of the webdesign and offers the user an effective searching of all needed information. The main menu is transparent and the user immediately sees all, what he can find on the pages. The menu card and the list of drinks are effectively input into the database MySQL, and that's why the user can display only the dishes, which he fancies or which attract him. All needed information is visible and differentiated with various colours of the text, eventually with the bullets. The biggest advantage is the administration environment, in which the administrator can simply change data in the tables, add and delete data. The new web server is at first sight more effective and practical. Unfortunately its application for internet is not possible, because the firm Jinlong s.r.o. hasn't provided the needed webhosting with the support of PHP and MySQL for now.

SEZNAM POUŽITÉ LITERATURY

- [1] Thomas A. Powell, Web Design – kompletní průvodce. Computer Press, 2004, ISBN: 80-722-6949-6.
- [2] Michael J. Hernandez, John L. Viescas, Myslíme v jazyku SQL – tvorba dotazů. Grada, 2004, ISBN: 80-247-0899-X.
- [3] W. Jason Gilmore, Velká kniha PHP5 a MySQL – kompendium znalostí pro začátečníky i profesionály. Zoner Press, 2005, ISBN: 80-86815-20-X.
- [4] Prokopová, Z., Databázové systémy MySQL+PHP. FAI UTB Zlín, 2006, Vysokoškolská skripta. ISBN: 80-7318-486-9.
- [5] Krčmář Jakub, Adobe Photoshop – praktický webdesign. Grada, ISBN: 80-247-1423-X.
- [6] Staníček, P, CSS – hotová řešení. Computer Press, 2006, ISBN: 80-251-1031-1.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

CSS Cascading Style Sheets

HTML Hypertext Markup Language

www World Wide Web

PHP Hypertext Preprocessor

SEZNAM OBRÁZKŮ

Obrázek 1 Thomas A. Powell - Web design – kompletní průvodce – pyramida web designu.....	11
Obrázek 2 Thomas A. Powell - Web design – kompletní průvodce – web jako médium	12
Obrázek 3 Thomas A. Powell - Web design – kompletní průvodce.....	14
Obrázek 4 Myslíme v jazyku SQL – ukázková tabulka.....	18
Obrázek 5 Myslíme v jazyku SQL –tabulka reprezentující událost.....	19
Obrázek 6 Myslíme v jazyku SQL – příklad primárního klíče	20
Obrázek 7 Myslíme v jazyku SQL – příklad cizího klíče.....	21
Obrázek 8 Myslíme v jazyku SQL – vztah jeden k jednomu	22
Obrázek 9 Myslíme v jazyku SQL – vztah jeden k více	23
Obrázek 10 Myslíme v jazyku SQL – vztah více k více	24
Obrázek 11 Úvodní stránka Jinlong Sport Clubu.....	30
Obrázek 12 Party list	32
Obrázek 13 Grafická podoba stránky	38
Obrázek 14 Prostředí PhpMyAdmin	40
Obrázek 15 Prostředí MySQL console.....	40

SEZNAM TABULEK

SEZNAM PŘÍLOH

PI Dokumentační CD, které obsahuje elektronickou verzi bakalářské práce a kompletní webový server včetně všech skriptů, obrázků, databází a všech potřebných věcí.

PŘÍLOHA P I: NÁZEV PŘÍLOHY