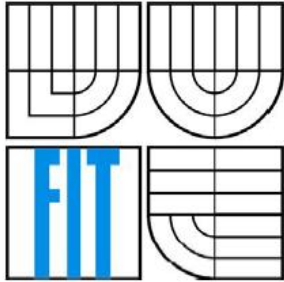




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

WEBOVÝ SYSTÉM PRO PRODEJ FOTOAPARÁTŮ

WEB-BASED SYSTÉM FOR CAMERAS SALE

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

RICHARD PLHAL

VEDOUCÍ PRÁCE
SUPERVISOR

ING. ŠÁRKA KVĚTOŇOVÁ

BRNO 2008

Abstrakt

Moje práce má za cíl vytvoření informačního systému, který bude sloužit k online nákupu přes internet se vším, co tato problematika obsahuje. Dle získaných teoretických i praktických znalostí jsem navrhl online internetový obchod, ve kterém pro demonstraci této technologie budu prodávat digitální fotoaparáty.

System by měl poskytnout běžným a registrovaným uživatelům přehledné rozhraní pro prohlížení, vyhledávání, třídění, srovnávání a objednávání z nabídky zboží. Zcela jiné rozhraní je navrženo pro administrátora, který se stará o správu databáze tzn. přidávání, mazání a úpravu jednotlivých položek. K této problematice jsem použil pro to určené technologie, které jsou například databázový systém MySQL, skriptovací jazyk PHP a jazyk HTML.

Klíčová slova

Internet, WWW, Internetový obchod, E-shop, HTML, PHP, CSS, MySQL, Corel Photoshop, WampServer, Informační systém, Databázový systém, Databáze

Abstract

The main objective of my work is to create an internet information system which will provide online shopping and everything what it needs. Based on theoretic and practical knowledge I designed an online internet shop with digital cameras in which I will demonstrate this technology.

System will allow browsing in catalog to regular and registered users, product search, sorting, comparison and ordering selected products from menu. Complete different interface have database administrator who care for database administration. It means adding, deleting , modifying records. To create this system I will use today's common technologies such as database system MySQL, scripting PHP language or HTML language.

Keywords

Internet, WWW, Internet shop, E-shop, HTML, CSS, MySQL, Corel PHOTO-PAINT, WampServer, Information System, Database System, Database.

Richard Plhal: Webové rozhraní pro prodej fotoaparátů, bakalářská práce, Brno, FIT VUT v Brně, 2008

Webový systém pro prodej fotoaparátů

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením ing. Šárky Květoňové a mého externího zadavatele ing. Tomáše Michala. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Richard Plhal
14.května 2008

Poděkování

Rád bych tímto poděkoval vedoucímu mé bakalářské práce slečně ing. Šárce Květoňové za odborné vedení při vytváření této práce a dále mému externímu zadavateli ing. Tomáši Michalovi.

Obsah

1. Úvod.....	8
2. Informační systém.....	9
2.1 Internet.....	9
2.1.1 WWW.....	9
2.2 Databázový systém.....	11
2.2.1 Databázové modely.....	11
2.2.2 MySQL.....	12
2.3 Skriptovací jazyky.....	13
2.3.1 PHP.....	13
2.4 Ostatní technologie.....	14
2.4.1 Kaskádové styly (CSS).....	14
2.4.2 Corel PHOTO- PAINT.....	16
3. Realizace webového rozhraní.....	16
3.1 Neformální specifikace.....	16
3.2 Analýza.....	16
3.2.1 Use Case Diagram (diagram případů použití).....	17
3.3 Specifikace požadavků.....	18
3.3.1 ER Diagram.....	19
3.3.2 Popis SQL databáze.....	19
3.4 Implementace.....	23
3.4.1 Funkčnost systému.....	24
3.4.2 Platforma pro vývoj a testování.....	30
3.4.3 Použité vývojové nástroje.....	30
4. Závěr.....	31
5. Literatura.....	32

1. Úvod

Trend nakupování přes internet ve všech odvětvích spotřebního sortimentu vzniklo v poměrně nedávné době. Je to velice efektivní řešení, jak zjednodušit nákup nebo jak demonstrovat pokrok nynějších technologií tvorby webových rozhraní. V posledních letech se tento druh nákupu velmi rozšířil a stále se vyvíjí.

Na internetu je dnes možné nakoupit téměř vše od produktů souvisejících s počítačovou technologií, přes veškerou elektroniku, knihy, kosmetiku, oblečení, potraviny až po automobily, software a také informace, které jsou mnohdy daleko důležitější než hmotné produkty.

Největší výhodou online nákupu je především v pohodlnosti nákupu. Uživatel vše vyřídí od monitoru svého počítače a zboží je mu doručeno na uvedenou adresu. Nesmíme však přehlédnout možná ještě daleko podstatnější rozdíl, a to je rozdíl cenový. V mnoha internetových obchodech jsou nabízené produkty až o desítky procent levnější než v klasických kamenných obchodech. Samozřejmě další důležitou výhodou je, že na internetu obchody nezavírají. Nakupující má tudíž možnost si objednat zboží např. ve svátcích či víkendech.

S výhodami tohoto druhu nakupování přichází zákonitě i nevýhody. Mezi ty nejpodstatnější patří nemožnost shlédnutí kupovaného produktu naživo. Jinak řečeno, téměř každý zákazník by si chtěl své zboží „osahat“, popřípadě demonstrovat jeho funkčnost. Každopádně i tento problém je částečně řešen díky odborníkům, kteří prostřednictvím nejrůznějších komunikačních mechanismů poradí s výběrem konkrétního produktu. Pro někoho může být nevýhodou způsob platby přes internet, i když z mého pohledu je to spíše výhoda, a v dnešní době je to už pomalu rozšířenější způsob platby než klasickými bankovkami.

Cílem práce je Vám představit webové rozhraní, které bude obsahovat všechny náležitosti popsané dříve, a dále komunikaci informačních systémů v internetu, technologie a jejich následnou implementaci a následně vlastní realizaci systému. V první části se zabývá specifikací a analýzou požadavků, ve druhé pak návrhem a implementací databáze a samotného webového systému pro prodej fotoaparátu.

Práce je vypracovaná pod záštitou firmy LUCCO a.s. a pravidelně konzultována s externím pracovníkem firmy.

2. Informační systém

Kapitola, ve které se seznámíme s metodami vytváření komunikačních částí informačních systémů v prostředí internetu, abychom pochopili, jak internetový obchod funguje a jak je implementován.

2.1 Internet

Internet je z technického hlediska tvořen desítkami tisíc počítačů, trvale spuštěných a zapojených v sítích. Počítače v síti pracují jednak jako servery, které službu poskytují, a klientské stroje, které služby využívají [1]. Je to síť typu WAN (Wide Area Network) pracující na základě komunikačního protokolu TCP/IP. Internet slouží k přenášení informací a poskytování mnoha služeb, jako jsou elektronická pošta, chat, www stránky, sdílení souborů, vyhledávání a další. Počítačům připojeným k internetu je přidělena IP adresa, která zajišťuje jejich jednoznačnou identifikaci.

2.1.1 WWW

WWW neboli World wide Web je jedna z nejpoužívanějších služeb internetu, která umožňuje přenos hypertextových souborů. Funguje na principu architektury klient/server, což znamená, že na serverech jsou uloženy soubory (dokumenty, stránky) a klienti (ostatní počítače připojené k internetu, jež nejsou servery) těmto serverům posílají požadavky a servery jim odpovídají (posílají například požadované stránky). Je to vlastně obrovská otevřená kniha s milióny stránek, kterou může každý číst, a do které může každý psát. Má obrovskou šířku využití např. noviny, encyklopedie, slovníky, učebnice, hudební přehrávače nebo televize [1].

WWW Prohlížeč

WWW prohlížeč neboli Web Browser je počítačový program, který dekóduje poměrně úsporná data a jednoduché instrukce zaslané serverem, sestaví z nich stránku jako ze stavebních cihlíček a tu zobrazí ve svém hlavním okně. Textové prohlížeče zobrazují stránky jako text. Obvykle velmi jednoduše formátovaný. Grafické prohlížeče umožňují složitější formátování stránky včetně zobrazení obrázků[1]. Pro zobrazení některých zvláštních součástí stránky, jako jsou Flashové animace nebo Javové applety, je třeba prohlížeč doplnit o specializované zásuvné moduly. Mezi nejznámější webové prohlížeče patří grafické Internet Explorer, Mozilla Firefox, Opera.

WWW stránka

Webová stránka je v podstatě internetová prezentace. Může být pouze textová nebo obsahovat nástroje, které jsou pro uživatele internetu daleko přitažlivější než pouhý text (obrázky, animace, zvuky). Potom je možno mluvit o dynamických webových stránkách, které mohou očekávat interakci od uživatele (například formuláře, ankety, internetové obchody).

URL (Universal Resource Locator)

URL adresy jsou adresy, pomocí kterých lze jednoznačně identifikovat konkrétní webové stránky, ale stejně tak i další objekty včetně emailových schránek (adres) konkrétních uživatelů.

Skládá se z označení protokolu použitého pro přístup (http, ftp), za kterým následuje dvojtečka a dvě lomítka, za nimiž je adresa cíle (někdy i úplnou specifikací souboru, popřípadě i části dokumentu) [2].

HTML (Hypertext Markup Language)

HTML je značkovací jazyk, který se používá k tvorbě WWW stránek. Jazyk HTML se skládá z prvků, které definují textový dokument a jeho zobrazení. Tyto prvky nazýváme značky (od toho značkovací jazyk) nebo také tzv. tagy.

HTML umožňuje tvorbu hypertextových odkazů, které jsou schopny navigace v dokumentu nebo odkazují na ostatní WWW stránky.

HTML File – textový soubor obsahující kód jazyka HTML. Obsahuje návod pro sestavení webové stránky, je čten, interpretován a ve výsledné podobě zobrazován prohlížečem v podobě webové stránky [3].

Cookie

Před vznikem souborů cookie neměly servery žádnou možnost jak zjistit, co už vlastně uživatel viděl. Vyplývá to z technologie protokolu HTTP, jenž je technologií bezstavovou. Jednotlivé stránky jsou na sobě nezávislé. Bez absence sledování pohybu uživatele na webovém serveru by neexistoval způsob, jak vytvořit například nákupní košík.

Soubory cookie jsou jedním ze způsobů, jak informace o tomto pohybu ukládat. Bohužel mezi uživateli internetu mají cookies špatnou pověst, neboť si myslí, že umožňují serverům dovědět se o nich příliš mnoho. Opak je však pravdou, protože jejich schopností je pouze ukládat informace pocházející z daného serveru. Cookies neznamenaají žádné nebezpečí pro počítač jako takový. Přesto cookies mohou být nebezpečné pro ochranu soukromí. Navštívený web si totiž může ukládat do cookies jakékoliv informace, které o návštěvníkovi zjistí a může tak postupně zjišťovat zájmy

konkrétního návštěvníka. Které stránky navštěvuje, jaké informace vyhledává, jak často daný web navštěvuje apod.

Díky informacím tohoto typu se může posléze i proti vůli návštěvníka využívat pro cílenou reklamu, statistické vyhodnocování chování návštěvníků, apod.

Architektura klient/server

Síťová architektura, která odděluje klienta (často aplikaci s grafickým uživatelským rozhraním) a server. Jednotlivé instance klientů komunikují se serverem, který obvykle běží na vzdáleném počítači. Klasickou ukázkou může být prohlížení webových stránek, kde webový prohlížeč je klient, který při požadavku uživatele na novou stránku kontaktuje vzdálený server a vyžádá si od něj patřičnou webovou stránku. Programem typu klient je například webový prohlížeč.

2.2 Databázový systém

Pro samotný přístup k datům uloženým v databázi používáme speciální software. Nazývá se anglicky *Database Management System* (DBMS) nebo chcete-li česky *Systém řízení báze dat* (SŘBD). Fyzická struktura uložených údajů potom nemusí být aplikačnímu programu a tím pádem ani uživateli vůbec známa [7].

Hlavním úkolem je poskytnout těmto uživatelům abstraktní pohled na data. Uživatel nepotřebuje vědět, jak jsou data uložena fyzicky v databázi. Jde hlavně o to, jak jsou mu prezentována a jak jsou navzájem propojena. SŘBD se tedy stará o uložení, přístup a o manipulaci s daty. Dále by měl zajišťovat bezpečnost a integritu dat, souběžný přístup (transakční zpracování) a co nejvyšší výkonnost.

2.2.1 Databázové modely

Z hlediska způsobu ukládání dat a vazeb mezi nimi můžeme rozdělit databáze do tří základních typů.

Hierarchický model

Nejstarší z uvedených je hierarchické modelování databází. Toto pojetí pochází z reálného uspořádání světa. Jako příklad si můžeme vzít třeba model organizace moci, rozklad výrobků na součástky, strom adresářů aj.

Pro hierarchické modelování je typická práce se stromy, kdy ve stromu jsou realizovány vztahy 1:N. Příkladem hierarchické databáze může být IMS od firmy IBM [7].

Síťový model

Variací hierarchického modelu je síťový model databáze. V síťovém modelování je možné vyjadřovat vedle vztahů 1:N i vztahy M:N. Fyzická realizace síťového modelu je ale náročná a aktualizace obvykle komplikovaná [7].

Základní výhodou hierarchického a síťového modelu je efektivnost zpracování, tj. rychlost přístupu k datovým záznamům. Na druhé straně mezi nevýhody patří to, že je nesnadné jednou nadefinované stromy a vazby mezi nimi měnit. Nejsou uzpůsobeny pro dotazy.

Relační model

Z teoretického hlediska je nejpropracovanější relační model databáze. Relační model definuje způsob, jakým je možné reprezentovat strukturu dat, způsoby jejich ochrany a operace, které můžeme nad daty provádět. Relační databáze je sestavená z řady tabulek, jejichž sloupce mohou být vázány na sloupce v jiných tabulkách. Takto propojená datová pole jsou na sobě určitým způsobem závislá. Jejich vztahy jsou založeny na klíčových hodnotách uložených v příslušných sloupcích [7].

U relačních databází je základní výhodou relativně snadná modifikace a propojování tabulek a s nimi spojená možnost dotazů. Slabým místem je nízká efektivnost zpracování, což se projevuje v tom, že řada příkazů vyžaduje velké množství přístupů na disk a tím se zpomaluje zpracování.

2.2.2 MySQL

Je to databázový systém, který vyvinula švédská firma MySQL AB. Jeho hlavními autory jsou Michael „Monty“ Widenius a David Axmark. Je považován za úspěšného průkopníka dvojího licencování – je k dispozici jak pod bezplatnou licenci GPL, tak pod komerční placenou licenci.

MySQL je multiplatformní databáze. Komunikace s ní probíhá – jak už název napovídá – pomocí jazyka SQL. Podobně jako u ostatních SQL databází se jedná o dialekt tohoto jazyka s některými rozšířeními.

Pro svou snadnou implementovatelnost (lze jej instalovat na Linux, MS Windows, ale i další operační systémy), výkon a především díky tomu, že se jedná o volně šiřitelný software, má vysoký podíl na v současné době používaných databázích. Velmi oblíbená a často nasazovaná je kombinace MySQL, PHP a Apache jako základní software webového serveru.

MySQL bylo od počátku optimalizováno především na rychlost a to i za cenu některých zjednodušení: má jen jednoduché způsoby zálohování a až donedávna nepodporovalo pohledy,

triggery a uložené procedury. Tyto vlastnosti jsou doplňovány teprve v posledních letech, kdy začaly nejčastějším uživatelům produktu – programátorům webových stránek – již poněkud scházet.

Zabezpečení

System MySQL splňuje i vysoké nároky kladené na zabezpečení dat před zneužitím nepovolanými osobami. Zabezpečení dat v MySQL nemá obdoby. Přístup k datům lze určovat i ze vzdáleného počítače. Takto lze stanovit například to, jaká data jsou určena pro toho kterého uživatele. Databáze lze uzamknout dokonce i prostřednictvím operačního systému, který v tomto případě bude hrát roli strážce. MySQL je v současné době adaptován téměř pro všechny platformy.

2.3 Skriptovací jazyky

Skriptovací jazyky jsou programovací jazyky, které jsou učeny pro psaní skriptů. Patří sem zejména Perl, Python, PHP. Skriptovací jazyky jistě neprodukují superrychlé programy, ale zde nejde o efektivitu kódu, jako spíše o efektivitu programování.

2.3.1 PHP

Od roku 1994 je PHP jedním z nejpoužívanějších způsobů tvorby dynamicky generovaných WWW stránek. Jeho tvůrce (Rasmus Lerdorf) jej vytvořil pro svou osobní potřebu přepsáním z Perlu do jazyka C. Sada skriptů byla vydána ještě v téže roce pod názvem *Personal Home Page Tools*, zkráceně PHP. Později s rozšířením možností tohoto jazyka se ujal název „PHP: Hypertext Preprocessor“.

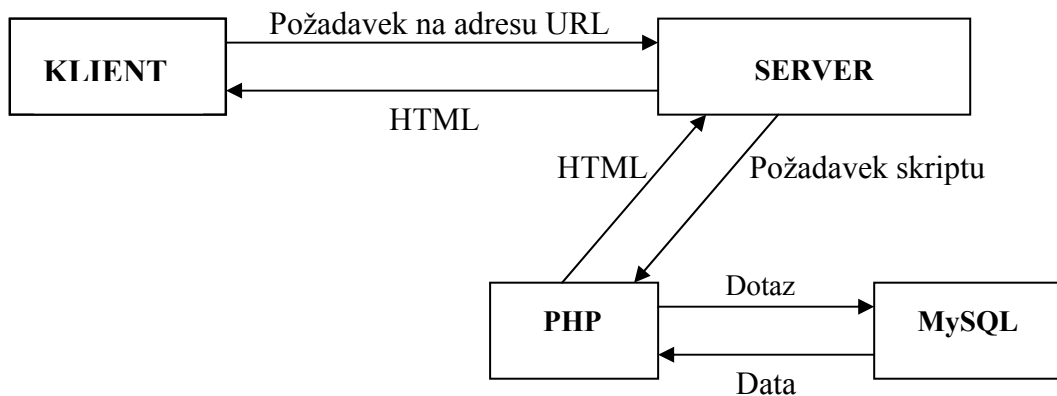
Vývoj PHP

Skriptovací programovací jazyk určený především pro programování dynamických internetových stránek. Nejčastěji se začleňuje přímo do struktury jazyka HTML, XHTML či WML, což je velmi výhodné pro tvorbu webových aplikací. PHP lze ovšem také použít i k tvorbě konzolových a desktopových aplikací. PHP skripty jsou prováděny na straně serveru. K uživateli je přenášen až výsledek jejich činnosti. Syntaxe jazyka kombinuje hned několik programovacích jazyků (Perl, C, Pascal a Java). PHP je nezávislý na platformě, skripty fungují bez úprav na mnoha různých operačních systémech. Obsahuje rozsáhlé knihovny funkcí pro zpracování textu, grafiky, práci se soubory, přístup k většině databázových serverů (mj. MySQL, ODBC, Oracle), podporu celé řady internetových protokolů (HTTP, SMTP, SNMP, FTP...)

PHP se stalo velmi oblíbeným především díky jednoduchosti použití a tomu, že kombinuje vlastnosti více programovacích jazyků a nechává tak vývojáři částečnou svobodu v syntaxi.

V kombinaci s databázovým serverem (především s MySQL nebo PostgreSQL) a webovým serverem Apache je často využíván k tvorbě webových aplikací. Díky velmi častému nasazení na serverech se vžila zkratka LAMP – tedy spojení Linux, Apache, MySQL a PHP nebo Perl.

S verzí PHP 5 se výrazně zlepšil přístup k objektově orientovanému programování podobný Javě.



Obr. 1: Komunikace dynamické webové aplikace využívající jazyk PHP a databázi MySQL

2.4 Ostatní technologie

Označují další použité technologie při programování internetového obchodu jako je Kaskádový styl CSS a Corel PHOTO-PAINT.

2.4.1 Kaskádové styly (CSS)

V počátcích webové technologie se objevila celá řada tagů jazyka HTML. Různé prohlížeče podporovaly své vlastní specifické tagy a žádný z nich neuměl všechny. Tyto specifické tagy se obvykle uváděly místo obecně podporovaných tagů. Tím nejvýznamnějším problémem, který vznikl, bylo to, že se v každém prohlížeči stránka zobrazovala jinak [3].

Proto konsorcium World Wide Web Consortium (W3C) vyvinulo kolem roku 1997 standard Cascading Style Sheets (CSS), který má tyto problémy eliminovat a vede k jednoduchému modelu, který odděluje styl stránky od jeho obsahu.

Co je to CSS?

CSS neboli *Cascading Style Sheets*. Je to jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML.

Jazyk byl navržen standardizační organizací W3C. Byly vydány zatím dvě verze specifikace CSS1 a CSS2 (plus CSS 2.1), pracuje se na verzi CSS3.

Hlavním smyslem je umožnit návrhářům oddělit vzhled dokumentu od jeho struktury a obsahu. Původně to měl umožnit už jazyk HTML, ale v důsledku nedostatečných standardů a konkurenčního boje výrobců prohlížečů se vyvinul jinak. Starší verze HTML obsahují celou řadu elementů, které nepopisují obsah a strukturu dokumentu, ale i způsob jeho zobrazení. Z hlediska zpracování dokumentů a vyhledávání informací není takový vývoj žádoucí [8].

I když prohlížeč kaskádové styly nepodporuje, například prohlížeče v mobilních zařízeních, text (a tedy to nejdůležitější) se zobrazí správně.

Existují tři způsoby vložení kaskádových stylů do stránky:

- *Přímo v atributu style HTML značky:*

Např. `<p style font-size: 12px; font-weight: bold>Formátovaný text</p>`

Formátovaný text bude tučný a bude mít velikost 12 pixelů.

- *Pomocí stylopisu v hlavičce stránky:*

Např. `<style type="text/css" media="screen">
body { background-color: black; }</style>`

Pozadí stránky bude mít černou barvu.

- *Nebo pomocí externího stylopisu, uloženém v externím souboru:*

Např. v soubor `styly.css` může obsahovat takto definovaný styl

```
.textOdkaz {  
    font-size: 12px;  
    font-weight: bold;  
}
```

Tento styl se jmenuje `textOdkaz` a použije se takto:

`<div class="textOdkaz">Formátovaný text</div>`

Písmo bude tučné a jeho velikost je nastavena na 12 pixelů.

Používání kaskádových stylů ve srovnání se samotným HTML v praxi přináší výhody:

- rozsáhlejší možnosti,
- oddělení struktury a styly,
- dynamická práce se styly,
- konzistentní styl,
- formátování XML dokumentů,
- větší kompatibilita alternativních webových prohlížečů.

2.4.2 Corel PHOTO- PAINT

Aplikace pro tvorbu ilustrací a grafický design stránky podporuje mnoho výkonných nástrojů včetně zpracování textů.

Nabízí kompletní sadu digitálních kreslicích nástrojů, které jsou rychlé, výkonné a snadno použitelné. Obsahuje impozantní počet profesionálních funkcí - od nástrojů pro precizní editaci až po umělecké štětce pro originální a tvůrčí malbu. Návrháři Web stránek mohou jednoduše navrhovat a publikovat pomocí pokročilých nástrojů pro Web design.

3. Realizace webového rozhraní

Popisu postupu při vytváření internetového obchodu tzn. detailní rozebrání specifikace, analýzy, návrhu a implementace.

3.1 Neformální specifikace

Po konzultaci s externím pracovníkem firmy, který mi nastínil základní a hlavní kritéria požadující firmy od internetového obchodu s fotoaparáty, jsem se zabýval těmito požadavky:

- jednoduchý a efektivní design
- rozlišit registrované a neregistrované uživatele
- statistiky prodejnosti výrobků za jednotlivý rok
- možnost různých výběrů a řazení fotoaparátů podle kritérií
- sledování stavu objednávek
- srovnání produktů

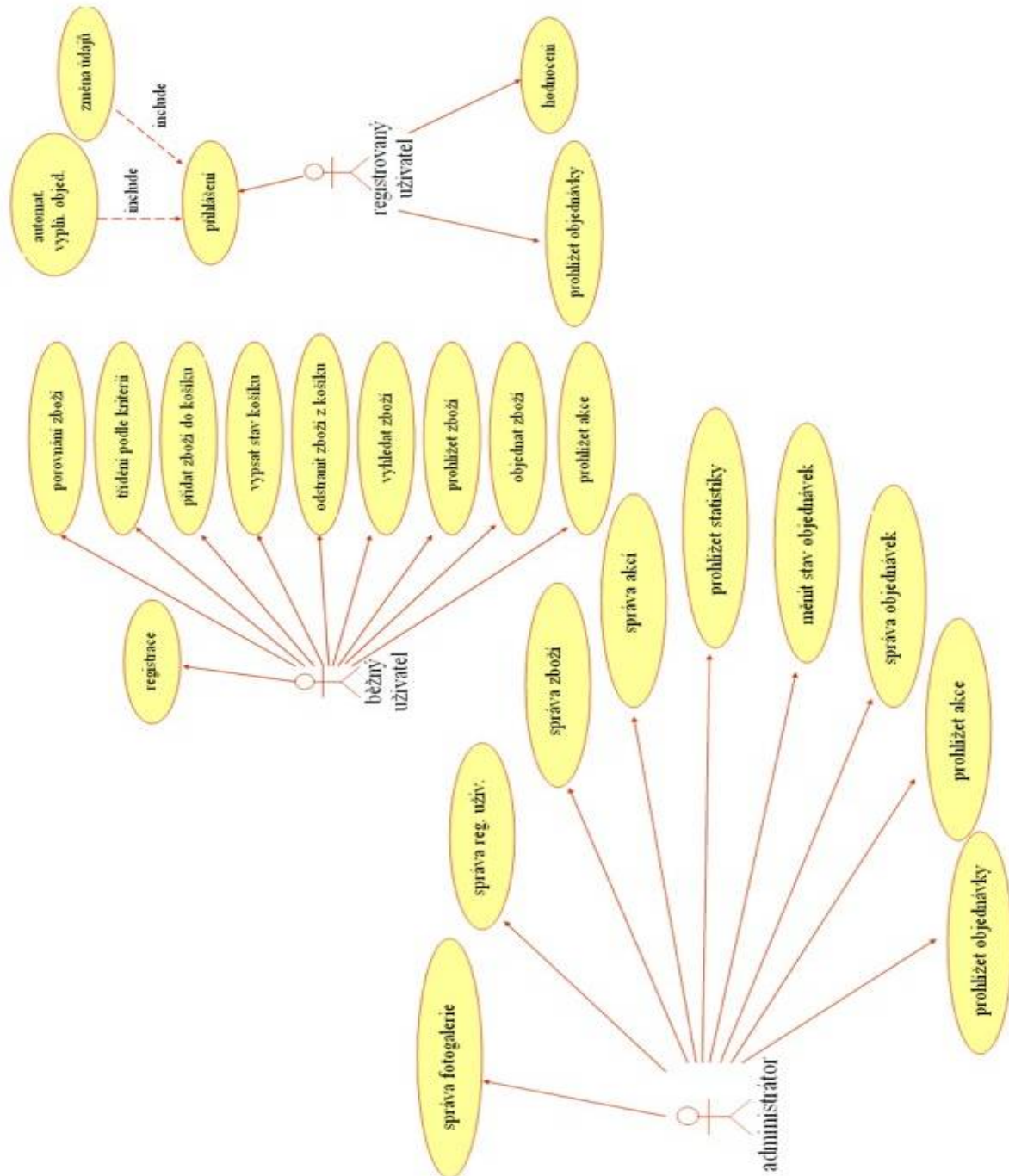
3.2 Analýza

V první řadě si musíme vytvořit datovou základnu, se kterou bude pracovat naše aplikace. Vzhledem k tomu, že při tvorbě internetového obchodu použiji skriptovací jazyk PHP, jsem jako systém pro správu relační databáze zvolil systém MySQL. Jako server k interpretaci jazyka PHP použiji WampServer. Všechny tyto technologie jsou dostupné zdarma.

K vytvoření návrhu jsem použil modelovací techniky jazyka UML, konkrétně Use Case Diagram a ER Diagram.

3.2.1 Use Case Diagram (diagram případů použití)

Use Case Diagram modeluje hlavní funkční požadavky na systém. Případy užití neboli Use Case jsou psány z pohledu zákazníka a podávají první představu o rozsahu projektu pomocí Use Case diagramu.



Obr. 2: Use Case Diagram (diagram případů použití)

3.3 Specifikace požadavků

Internetový obchod by měl splňovat kromě funkčního hlediska i designové a přehledné rozhraní pro jakéhokoliv uživatele.

Existují rozdílná rozhraní (přístupy) do internetového obchodu. V každém takovém rozhraní má konkrétní uživatel jiná práva. Je mnoho důvodů, proč tomu tak je. Jedním z hlavních je zabezpečení dat v databázi, které by měl spravovat administrátor, a registrovaní i neregistrovaní uživatelé by měli mít k nim pouze omezený přístup. Tudíž jsem rozlišil tři různé možnosti přístupu:

Neregistrovaný uživatel

Základní rozhraní pro všechny návštěvníky e-shopu. Uživatel má možnost prohlížet si přehledně nabídku zboží, vyhledávat podle názvu produktu, třídit dle různých kritérií, vybírat podle zadaných parametrů, přidávat zboží do košíku, upravovat zboží v košíku a vyprázdnit košík.

Vybrané zboží si pak může objednat, přičemž bude požádán o osobní informace (jméno, adresa, telefon, e-mail ...) a informace pro zpracování objednávky (způsob platby a doručení zboží).

Registrovaný uživatel

Výhodou registrovaného uživatele je, že oproti neregistrovanému uživateli může po přihlášení do systému sledovat všechny své nevyřízené objednávky. Další výhodou je automatické doplňování osobních údajů během objednávání vybraného zboží, což výrazně usnadní nákup z hlediska času.

Každý uživatel má možnost stát se registrovaným, a to díky jednoduchému vyplnění formuláře s osobními údaji a odeslání požadavku.

Administrátor

Speciální neveřejná část internetového obchodu, kde administrátor spravuje databázi s veškerými údaji. Přidává do ní položky (např. zboží, akce), mění a odstraňuje tyto položky, kontroluje objednávky apod.

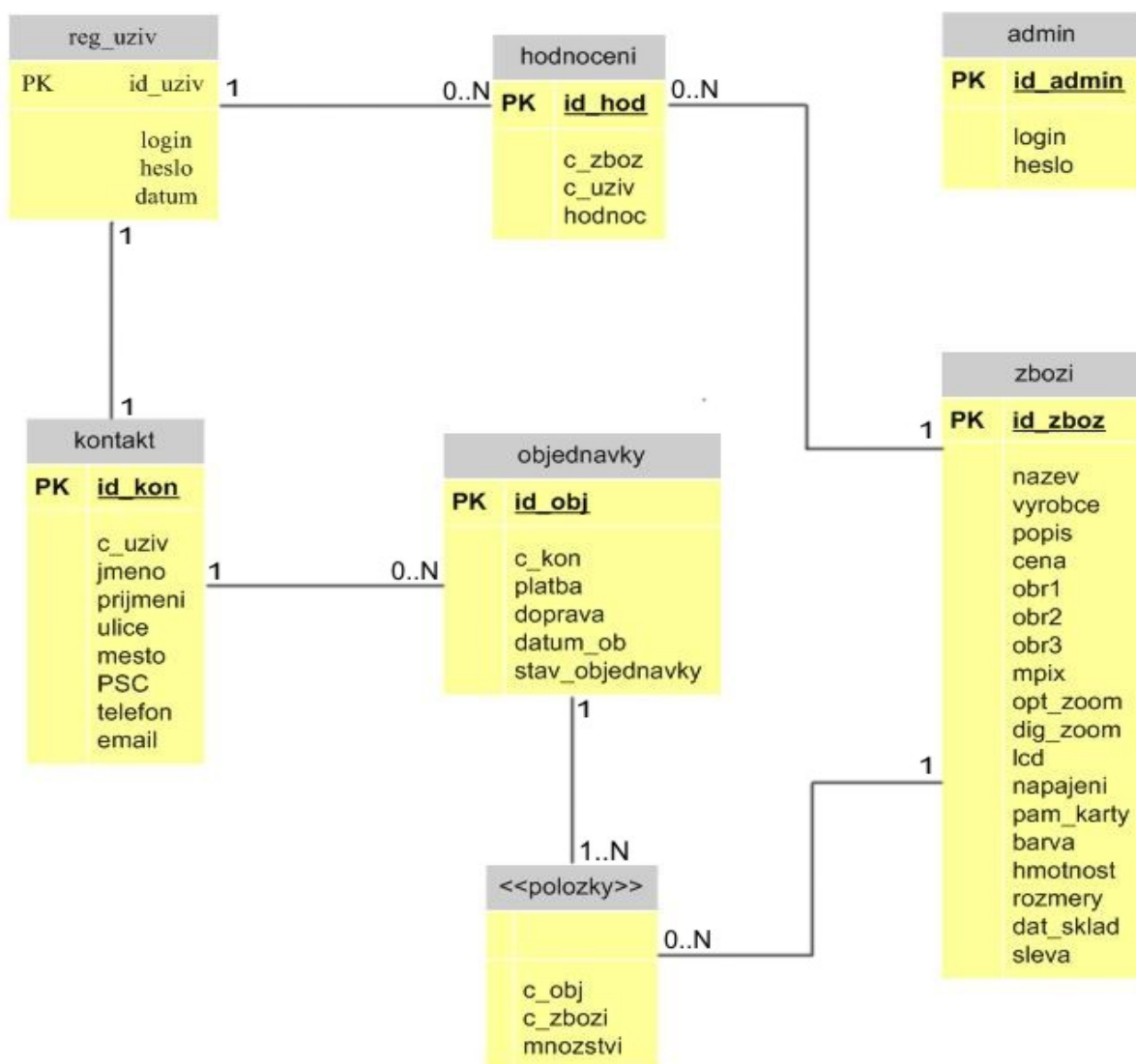
Vstup do této části je podmíněn přihlášením do systému jako administrátor.

3.4 Návrh

Databáze byla postavena na relačním modelu. Jak už jsem výše popisoval, základním prvkem tohoto modelu jsou tabulky. Snažil jsem se počet tabulek optimalizovat z důvodu efektivnosti uložení dat. Návrh jsem provedl pomocí ER Diagramu.

3.3.1 ER Diagram

ER Diagram je grafickým vyjádřením entit a vztahu mezi nimi. Jako entitu považujeme objekt, či událost, o které chceme ukládat informace v databázi. Vlastnosti každé entity nazýváme atributy. Fyzickou reprezentací entity v databázi je relační tabulka.



Obr. 3:ER Diagram (diagram případů použití)

3.3.2 Popis SQL databáze

Databáze obsahuje sedm tabulek, v nichž jsou uložena data, potřebná k provozu mnou implementovaného internetového obchodu.

Tabulka „Admin“

Tabulka není nijak vázána na ostatní tabulky v databázi, slouží pouze k identifikaci administrátora během přihlašování do systému.

Obsahuje tyto sloupce (atributy v ER Diagramu):

- **id_admina:** Primární klíč sloužící k jednoznačné identifikaci administrátora. Je typu unsigned int(5).
- **login:** Přihlašovací jméno administrátora. Je typu varchar(30)
- **heslo:** Heslo náležící přihlašovacímu jménu administrátora. Je typu varchar(32). Ukládá se do databáze pomocí hashovací funkce MD5, která je schopna jednosměrného šifrování.

Tabulka „zbozi“

Zde jsou uloženy informace o zboží v internetovém obchodě.

Obsahuje tyto sloupce (atributy v ER Diagramu):

- **id_zbozi:** primární klíč, jednoznačně identifikuje konkrétní produkt. Je typu unsigned int(5).
- **nazev:** Název nebo typ produktu (např. IXUS i5). Je typu varchar(30).
- **vyrobce:** Název konkrétního výrobce produktu (např. Canon). Je typu varchar(30).
- **popis:** Textový popis zboží. Je typu text. Sloupec má atribut NULL (nemusí být definován).
- **cena:** Cena produktu uvedena vč. DPH. Je typu decimal(7,2).
- **obr1:** Adresářová cesta na disku k uloženému souboru GIF (obrázek produktu). Sloupec má atribut NULL (nemusí být definován). Je typu varchar(50).
- **obr2:** Adresářová cesta na disku k dalšímu uloženému souboru GIF (obrázek produktu). Sloupec má atribut NULL (nemusí být definován). Je typu varchar(50).
- **obr3:** Adresářová cesta na disku k dalšímu uloženému souboru GIF (obrázek produktu). Sloupec má atribut NULL (nemusí být definován). Je typu varchar(50).
- **mpix:** Číselné vyjádření hodnoty MegaPixelů ve fotoaparátu. Sloupec má atribut NULL (nemusí být definován). Je typu decimal(2,1).
- **opt_zoom:** Číselné vyjádření hodnoty násobku optického zoomu fotoaparátu. Sloupec má atribut NULL (nemusí být definován). Je typu int(3).
- **dig_zoom:** Číselné vyjádření hodnoty násobku digitálního zoomu fotoaparátu. Sloupec má atribut NULL (nemusí být definován). Je typu int(3).
- **lcd:** Informace o typu LCD displeje. Sloupec má atribut NULL (nemusí být definován). Je typu varchar(20).

- **napajeni:** Informace o způsobu napájení fotoaparátu. Sloupec má atribut NULL (nemusí být definován). Je typu varchar(50).
- **pam_karty:** Informace o druhu paměťové karty, která slouží pro dany typ fotoaparátu. Sloupec má atribut NULL (nemusí být definován). Je typu varchar(50).
- **barva:** Druh barvy fotoaparátu. Sloupec má atribut NULL (nemusí být definován). Je typu varchar(20).
- **hmotnost:** Číselná hodnota hmotnosti v gramech. Sloupec má atribut NULL (nemusí být definován). Je typu int(3).
- **rozmery:** Rozměr fotoaparátu ve tvaru (š x v x h). Sloupec má atribut NULL (nemusí být definován). Je typu varchar(50).
- **dat_sklad:** Datum uskladnění daného zboží a použije se pro pozdější zvýraznění novinek. Je typu varchar(50).
- **sleva:** Sleva na konkrétní produkt vyjádřená v procentech. Sloupec má atribut NULL (nemusí být definován). Je typu int(3).

Tabulka „Reg_uziv“

V tabulce „Reg_uziv“ jsou uloženy informace o registrovaných uživateli. Na ni se váže tabulka „Kontakt“, která o nich ukládá ostatní osobní informace. Bude popsána níže.

Obsahuje tyto sloupce (atributy v ER Diagramu):

- **id_uziv:** Primární klíč, jednoznačná identifikace registrovaného uživatele. Je typu unsigned int(10).
- **login:** Přihlašovací jméno registrovaného uživatele. Je typu varchar(30).
- **heslo:** Heslo náležící přihlašovacímu jménu registrovaného uživatele. Je typu varchar(32). Ukládá se do databáze pomocí hashovací funkce MD5, která je schopna jednosměrného šifrování.
- **datum:** Datum registrace tohoto uživatele tzn. kdy byl uložen do databáze. Je typu date.

Tabulka „Kontakt“

Zde jsou uloženy veškeré informace o zákazníkovi, který si objednal jakékoli zboží z databáze. Tabulka je spojena s tabulkou Reg_uziv pomocí sloupce c_uziv, který odpovídá sloupci id_uziv v tabulce Reg_uziv. Toto spojení (vztah), je ohodnoceno tzv. „multiplicitou 1 ku 1“. Znamená to, že registrovaný uživatel může mít v databázi uložen pouze jeden kontakt a jeden konkrétní kontakt patří pouze jednomu uživateli.

Pokud se uživatel registruje, jeho osobní údaje jsou uloženy právě zde. Je to z toho důvodu, abych nemusel tvořit tři tabulky pro registrované uživatele, neregistrované uživatele a jejich kontakt (adresa, telefon ...).

Obsahuje tyto sloupce (atributy v ER Diagramu):

- **id_kon:** Primární klíč, jednoznačná identifikace konkrétního kontaktu. Je typu unsigned int(5).
- **c_uziv:** Pomocí tohoto sloupce je propojena tabulka „Kontakt“ s tabulkou „Reg_uziv“ (sloupec id_uziv). Je typu unsigned int(10). Má atribut NULL, což znamená, že tento sloupec nemusí být definován (v případě uložení kontaktu neregistrovaného zákazníka).
- **jmeno:** Jméno uživatele. Je typu varchar(30).
- **prijmeni:** Celé příjmení uživatele. Je typu varchar(30).
- **ulice:** Úplný název ulice a číslo popisné. Je typu varchar(50).
- **mesto:** Název města. Je typu varchar(30).
- **PSC:** Poštovní směrovací číslo. Je typu unsigned int(5).
- **telefon:** Osobní telefon uživatele. Je typu bigint(20).
- **email:** Adresa elektronické pošty. Je typu varchar(50).

Tabulka „objednavky“

V této tabulce jsou uloženy informace o jednotlivých objednávkách zboží. Je vázána na tabulku „Kontakt“ pomocí sloupce c_kon, který odpovídá sloupci id_kon v tabulce „Kontakt“. Tento vztah je ohodnocen multiplicitou 1 ku N (na jeden kontakt je možné mít libovolné množství objednávek a jedna konkrétní objednávka se váže právě k jednomu kontaktu).

Obsahuje tyto sloupce (atributy v ER Diagramu):

- **id_obj:** Primární klíč, sloužící k jednoznačné identifikaci konkrétní objednávky. Je typu unsigned int(5).
- **c_kon:** Pomocí tohoto sloupce je propojena tabulka „Objednavka“ s tabulkou „Kontakt“ (sloupec id_kon). Je typu unsigned int(5).
- **platba:** Způsob platby za objednané zboží. Je typu varchar(30).
- **doprava:** Způsob dopravy zboží uživateli. Je typu varchar(40).
- **datum_ob:** Datum objednání zboží. Je typu date.
- **stav_objednavky:** V jakém stavu rozpracovanosti se objednávka nachází. Je typu varchar(255). Má atribut NULL (v případě prvotního uložení objednávky). Zákazníkovi se zobrazí implicitní hodnota „nezpracováno.“

Tabulka „Polozky“

Tato tabulka je spojena vztahem agregace s tabulkou „Objednavka“ pomocí sloupce `c_obj` (odpovídá sloupci `id_obj` v tabulce „Objednavka“) a s tabulkou „Zbozi“ pomocí sloupce `c_zbozi` (odpovídá sloupci `id_zbozi` v tabulce „Zbozi“). Agregace je silnější formou vztahu asociace, jenž je použit v ostatních případech. Tabulka „Polozky“ by neměla smysl bez tabulek, na které se váže.

Dále jsem ji modeloval jako slabou entitní množinu a to z důvodu, že když zanikne entita „Objednavka“ musí zpravidla zaniknout i entita „Polozky“. Stejně je to se vztahem k tabulce „Zbozi“.

Objednávka může obsahovat jednu a více položek, avšak konkrétní řádek v tabulce „Polozky“ náleží pouze jedné objednávce. Vztah mezi tabulkou „Zbozi“ vyjadřuje skutečnost, že konkrétní položka obsahuje pouze jeden produkt a konkrétní produkt může být obsažen v libovolném počtu položek.

Obsahuje tyto sloupce (atributy v ER Diagramu):

- **c_obj:** Spojení s konkrétní identifikující entitou „objednavka“. Je typu `unsigned int(5)`.
- **c_zbozi:** Spojení s konkrétní identifikující entitou „zbozi“. Je typu `unsigned int(5)`.
- **mnozstvi:** Množství objednané položky z tabulky „zbozi“. Je typu `unsigned int(10)`.

Tabulka „hodnoceni“

Poslední tabulka obsahuje informace o hodnocení jednotlivých produktů různými uživateli. Samozřejmě uživatel může a nemusí hodnotit daný produkt, tudíž produkty můžou i nemusí mít hodnocení. Tuto tabulku jsem implementoval jako samostatnou z důvodu přehledného zjištění počtu hodnotících lidí a dále zjištění jestli daný uživatel už nehodnotil tento produkt dříve.

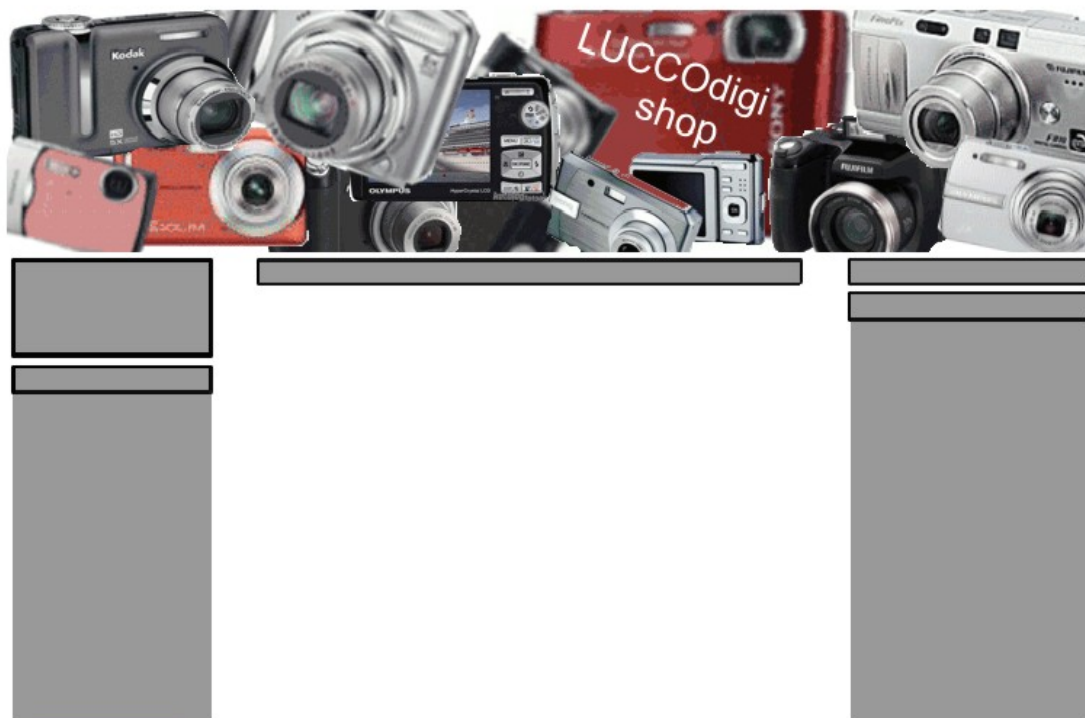
Obsahuje tyto sloupce (atributy v ER Diagramu):

- **id_hod:** Primární klíč, jednoznačná identifikace konkrétního hodnocení. Je typu `unsigned int(5)`.
- **c_uziv:** Spojení s konkrétní identifikující entitou „reg_uziv“. Je typu `unsigned int(10)`.
- **c_zbozi:** Spojení s konkrétní identifikující entitou „zbozi“. Je typu `unsigned int(5)`.
- **hodnoc:** Obsahuje číslo od 1 do 5 pro výpočet průměrného hodnocení. Sloupec má atribut `NULL` (nemusí být definován). Je typu `int(1)`.

3.4 Implementace

Na základě analýzy jsem se rozhodl při implementaci pro volně šiřitelnou verzi PHP a MySQL serveru a to WampServer2.0.

Pro tvorbu grafického rozhraní jsem použil program Corel PHOTO-PAINT, kde jsem navrhl úvodní logo obrázek. Rozprostření ovládacích prvků a celkový vzhled mého internetového obchodu jsem převedl do tabulek ve formátu HTML. Tím sem si vytvořil graficky podklad pro tvorbu mého rozhraní.



Obr.4:Grafický návrh vytvořený programem Corel PHOTO-PAINT

3.4.1 Funkčnost systému

Celý systém je uložen v adresáři „shop“. Ve stejném adresáři jsou obsaženy všechny skripty tvořící a obsluhující rozhraní pro registrované i neregistrované uživatele. Tento adresář obsahuje další dva podadresáře („/obrazy“ a „/images“), ve kterých jsou uloženy všechny soubory související s grafickým uživatelským rozhráním. Jsou to soubory typu „GIF“ a „JPG“.

V podadresáři „admin“ jsou uloženy skripty, které tvoří a obsluhují rozhraní pro administrátora.

Věřejná část systému – popis skriptů

Každému uživateli, ať už registrovanému či nikoli, se vygeneruje hned na začátku session s názvem „kosik“. To se děje pomocí funkce „session_start()“, která vygeneruje unikátní identifikační číslo pro

okno prohlížeče (respektive uživatele), aby byla zajištěna separace jednotlivých nakupujících. Toto se děje z důvodu, kdyby ve stejný čas bylo v e-shopu více uživatelů najednou, vkládali by si zboží navzájem do jednoho košíku. Tuto skutečnost existence sessions vylučuje. Navíc je to velmi vhodný způsob pro systém, jak si zapamatovat, co už uživatel vykonal.

Index.php: Veškeré zobrazení internetového obchodu. Zobrazí se jako první při vstupu na stránky. Za pomoci ostatních skriptů generuje všechny ovládací prvky a obsluhuje jejich funkčnost.

Navigace v menu je ovládána pomocí HTML příkazu:

```
<a href="index.php?zobraz=3" class="odkaz">Fujifilm</a>
```

Příkaz pošle po kliknutí na vybranou značku zboží sám sobě hodnotu proměnné „zobraz“ a podle ní se zobrazí požadované zboží.



Obr.5:Navigace pomocí menu

Obsluha zobrazení se děje na základě hodnoty „zobraz“ pomocí PHP příkazu „switch“.

```
$vyber = $_GET['zobraz'];  
switch ($vyber)  
{  
  
    case "3":  
        $znacka='Fujifilm';  
        include "./vypis_zbozi.php"  
        break;  
}
```

Podle hodnoty proměnné „zobraz“ (respektive „vyber“) se provede příkaz „include“, který připojí požadovaný skript. Tohle se provádí i pro ostatní části levého navigačního menu, kde se kromě rozdělení podle značek, náchází také rozdělení podle počtu MegaPixelů. Tento výběr sem začlenil do hlavního navigačního pole, protože si myslím, že tohle je pro většinu nakupujících jedno z hlavních kritérií při koupi fotoaparátu.

Vypis_zbozi.php: Přehledné vypsání požadovaného druhu zboží podle výše uvedené hodnoty proměnné „zobraz“. Pokud je počet zboží v databázi vysoký, dokáže výpis rozdělit na několik stránek. Výpis fotoaparátu také můžeme třídit podle pěti různých kritérií (cena, název, rozlišení, paměťové karty, zoom) s tím, že kritérium „název“ se nastaví implicitně. Toto dělení se dále může nastavit jako

vzestupné nebo sestupné. Pro vkládání do košíků je zde vytvořen jednoduchý formulář, jenž je ovládán skriptem „kos_pridat.php“.



Obr.6:Formulář pro vkládání do košíku

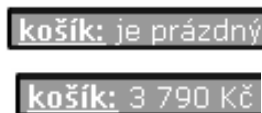
HTML příkaz sloužící k vytvoření formuláře, ve kterém je uveden skript, který pak získaná data ošetří.

```
<form action='kos_pridat.php' method='post'></form>
```

Kos_pridat.php: Ošetření dat z formuláře pro vkládání do košíku. Data jsou poslána metodou „POST“, což jsem specifikoval v atributu „method“. Je to jedna ze dvou metod (druhá je „GET“), které jsou povoleny protokolem HTTP. „POST“ je daleko bezpečnější, protože přenášená data jsou neviditelná. Metoda „GET“ oproti tomu posílá data cílové stránce jako posloupnost parametrů připojených za adresu URL a tak má každý možnost si je přečíst.

Každý session „kosik“ je vlastně pole, do kterého jsou vkládány hodnoty id_zbozi z tabulky „Zbozi“ pro identifikaci konkrétního produktu a počet objednávaných kusů. Až budeme potřebovat zjistit, co už má uživatel v košíku, jednoduše jeho obsah vypíšeme.

Stav_kosiku.php: Pomocí tohoto skriptu se mění ukazatel stavu košíku. Pokud je v něm už nějaké zboží, zobrazuje jeho celkovou cenu, kterou později vypíše podrobně.



Obr.7:Ukazatel stavu košíku

Vypis_kosik.php: Kliknutím na tlačítko ukazatele košíku, které můžete vidět na předchozím obrázku, se použije skript Vypis_kosik.php. Přehledně vypíše celý obsah vašeho košíku a vypočítá celkovou cenu vaší objednávky. Zobrazí se zde i formulář pro možnou změnu obsahu košíku.

Název	Cena	Množství	Celková cena
Fujifilm FinePix Z10fd	3 185 Kč	<input type="text" value="1"/>	3 790 Kč <input type="button" value="x"/>
Celkem: 3 790,15 Kč vč. DPH			
<input type="button" value="aktualizovat"/>		<input type="button" value="vyprázdnit košík"/>	<input type="button" value="jít k pokladně"/>

Obr.8:Obsah košíku

Kliknutím na tlačítko „vyprázdnit košík“, se provede PHP příkaz,

```
unset ($_SESSION['kosik'][$klik]);
```

který vyprázdní session „kosik“.

Pokladna.php: Pokud už má uživatel vybráno zboží, které si chce objednat, a následně ho vložil do košíku, potom klikne na tlačítko „jít k pokladně“. Dojde k přepnutí řízení pomocí funkce „header()“ do skriptu Pokladna.php. Skript vypíše opět formulář, do kterého uživatel vypíše své osobní údaje, aby mu mohlo být zboží zasláno na určenou adresu. Musí zadat také způsob platby a dopravy.

Skript pro kontrolu a ošetření povinných údajů využívá jednoduchou funkci jazyka PHP:

```
if (empty($_POST['jmeno']))
```

Vyplněná data je však potřeba zkontrolovat. To je možné provést regulárním výrazem, který jazyk PHP podporuje. Pomocí této technologie definujeme vzor, podle kterého může řetězec nabývat svou hodnotu. Například kontrolu správně zadaného emailu:

```
if (!ereg ( "^[[:alnum:]]+[a-z0-9_.-]*@[a-z0-9.-]+\.[a-z]{2,4}$" ,  
trim($_POST['email'])))
```

Zápis tohoto regulárního výrazu znamená, že na začátku řetězce může být jakýkoli alfanumerický znak, následuje libovolný počet malých písmen, číslic, podtržítok, teček nebo pomlček, dále musí být zavináč, potom následuje jeden nebo více znaků z množiny uvedené v závorce, pak povinná tečka, za ní očekáváme dva až čtyři znaky malých písmen. Pokud řetězec zadaný uživatelem nesplní tyto podmínky, vrací funkce hodnotu FALSE.

Funkce „trim()“ odstraní netisknutelné znaky ze začátku a konce řetězce.

Prihlaseni.php, Registrace.php: Skripty se starají o přihlašování a registraci do systému. Pokud je uživatel zaregistrovaný, nemusí údaje, zadávané během objednávky vyplňovat znovu. Stačí se přihlásit a o doplnění údajů se postará skript. Po přihlášení se totiž vygeneruje session „VaseID“, která je využita pro kontrolu identifikace a následném doplnění údajů.

Uzivatel.php: Pomocí session „VaseID“ se ve skriptu kontroluje, zda je uživatel přihlášen. V případě, že ano, tak vypíše jeho jméno do ukazatele pro identifikaci.



Obr.9:Ukazatel pro identifikace uživatele

Vypis_akce.php: Vypsání zboží ve slevě, pokud takové existuje. Vypisuje se maximálně tři zlevněné produkty, a to v náhodném pořadí a náhodně vybrané z databáze. To umožňuje funkce jazyka SQL „rand()“.



Obr.10: Výpis zlevněného zboží

Hledat.php: Skript umožňuje jednoduché vyhledávání v databázi podle názvu produktu. Realizoval jsem to pomocí příkazu jazyka SQL „LIKE“. Tento příkaz funguje tak, že hledaný řetězec porovnává z řetězci uloženými v databázi. Proto je potřeba z hledaného řetězce odstranit všechny mezery. To zajistím funkcí jazyka PHP:

```
$hlSlovo = str_replace(" ", "", "$slovo");
```

Tato funkce najde všechny mezery v řetězci „slovo“ a nahradí je prázdným znakem. Pak lze spolehlivě porovnat, zda se hledaný řetězec nachází v databázi nebo je podřetězcem některého z řetězců v databázi uložených.

Gify.css: Soubor definic kaskádových stylů. Připojí se na začátku skriptu index.php takto:

```
<link href="gify.css" rel="stylesheet" type="text/css">
```

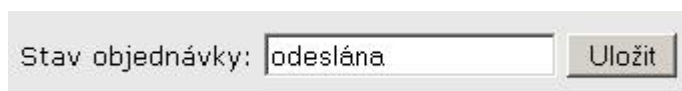
Ostatní skripty se nijak zvlášť už neodlišují, proto je pouze vypíšu bez komentářů: **detailZb.php**, **nakRad.php**, **objednavky.php**, **srovnaniProduktu.php**.

Neveřejná část systému

Přihlášení je podmíněno zadáním loginu a hesla administrátora, jenž jsou porovnány v tabulce „Admin“.

Implementace skriptů se od veřejné části nijak zvlášť neliší. Administrátor tu má možnost zjistit, kteří uživatelé jsou zaregistrovaní, a má možnost je zrušit. Dále má možnost pracovat se zbožím tzn. přidávat, editovat a mazat. V neposlední řadě pro administrátora je práce s akcemi na zboží. Má možnost je přidávat, editovat a rušit. Dále práce s objednávkami tzn. výpis obsahu objednávky, kde

nastavuje stav rozpracovanosti dané objednávky pro registrované zákazníky, kteří ji mají možnost vidět po svém přihlášení a kliknutí na svůj přihlašovací login.



Obr.12:Nastavení stavu objednávky

Nakonec sleduje statistiky celkového obratu firmy a výpis třech nejprodávanějších fotoaparátů za daný rok. Funkce má hlavně informativní účel, aby prodejce měl možnost např. vytvářet cenové akce na nejprodávanější zboží.

Rok	Obrat	3 nejprodávanější
2008	15 794 Kč	Fujifilm FinePix Z10fd - 2x Olympus FE-280 - 1x Fujifilm FinePix A920 - 1x

Obr.13:Výpis statistik

SQL dotazy

V PHP skriptech potřebujeme často přistupovat k databázi. Provádění SQL dotazů umožňuje příkaz „mysql_query()“. Načtení výsledku konkrétního dotazu provedeme příkazem „mysql_fetch_array()“. Pro zjištění počtu záznamů (řádků), vzniklých SQL dotazováním můžeme použít příkaz „mysql_num_rows()“.

Příklady použití

```
$result = mysql_query ("SELECT * FROM zboží WHERE id_zbozi='$id'");  
$pocet = mysql_num_rows($result);  
while ($radek = @mysql_fetch_array ($result)) {  
    $nazev = $radek['nazev'];  
}
```

V proměnné „result“ bude uložen výsledek SQL dotazu. S tím ovšem ještě nemůžeme pracovat. Musíme proto do proměnné „radek“ načíst pomocí zmiňované funkce pole záznamů, odpovídající jednomu řádku z tabulky. Přístup k těmto záznamům demonstruje přiřazení názvu z databáze do proměnné „nazev“.

3.4.2 Platforma pro vývoj a testování

System jsem testoval ve třech různých prohlížečích. Jsou to Microsoft Internet Exploreru 7.0, Mozilla Firefox 2.0, Opera 9.27.

Testovací platforma:

- operační systém Windows XP
- WWW server WampServer 2.0
- databázový server MySQL 5.0.45
- skriptovací jazyk PHP 5.2.5

3.4.3 Použité vývojové nástroje

- Corel PHOTO-PAINT
- PSPad 4.5.2
- Microsoft Word
- Adobe Acrobat Reader 6.0 CE Professional

V kapitole realizace webového rozhraní jsme se dozvěděli postup, který jsem dodržoval při řešení mého internetového obchodu a to přes specifikaci požadavků, návrh a implementaci.

4. Závěr

Cílem této bakalářské práce bylo zhodnotit metody vytváření informačních systémů v prostředí internetu, seznámit se podrobněji s problematikou internetových obchodů a na základě získaných znalostí vytvořit informační systém s použitím skriptovacího jazyka PHP.

Kromě tohoto jazyka jsem použil k implementaci i značkovací jazyk HTML a databázový server MySQL, který postačuje potřebám ukládání dat z internetového obchodu, v němž prodávám digitální fotoaparáty. Důraz při tvorbě systému jsem kladl na funkčnost, přehledné uživatelské rozhraní a jednoduchou obsluhu, aby byl nákup v e-shopu, co nejpříjemnější pro uživatele. Existují také jakási nepsaná pravidla pro rozmístění a vzhled ovládacích prvků, kterých jsem se snažil držet. Uživatel tak nemusí zdlouhavě zkoumat systém ovládání a může se věnovat samotnému nákupu.

Aplikace umožňuje jak běžným, tak registrovaným uživatelům pohodlně listovat v nabídce zboží, vyhledávat konkrétní produkt podle názvu a následně jej objednat. Pro pohodlné objednávání jsem vytvořil interaktivní nákupní košík, jehož obsah lze libovolně a jednoduše měnit. Pokud se uživatel zaregistruje do databáze, bude nákup ještě snadnější, neboť nebude muset neustále zadávat osobní údaje, nutné k objednání zboží. Pro administrátora systému bylo vytvořeno rozhraní pro správu, které umožňuje přidávání, změnu a mazání položek v databázi.

Při tvorbě této práce jsem si značně rozšířil nejen znalosti skriptovacích jazyků (především jazyka PHP), databázové komunikace, ale také znalosti z oblasti tvorby grafického uživatelského rozhraní. Zdokonalil jsem se velmi i v oblasti celkového procesu vývoje dynamických internetových aplikací.

Možnosti dalšího pokračování vidím v rozšíření funkcí mého internetového obchodu pro uživatele, a to dle potřeby zadavatelské firmy a nových trendů v informační a internetové technologii.

5. Literatura

- [1] Kosek, J. HTML Tvorba dokonalých WWW stránek. 1. vydání. Praha: Grada 1998, ISBN 80716960880
- [2] Peterka, J. et al. Se z@vináčem na internet. 1. vydání. Praha: Academia 1999, ISBN 8020007873
- [3] Hlavenka, J. et al. Vytváříme WWW stránky a spravujeme moderní sítě. 2. vydání. Brno: Computer Press 1998, ISBN 80-7226-080-4
- [4] Bráza, J. PHP 4. Praha: Grada 2002, ISBN 8024704420
- [5] Řepa, V. Analýza a návrh informačních systémů. 1. vydání. Praha: Ekopress 1999, ISBN 80-86119-13-0
- [6] Mach, J. PHP pro úplné začátečníky. 2. vydání. Brno: Computers Press 2003, ISBN 80-7226-834-1
- [7] Pokorný, M. Vyvíjíme databázový a informační systém IV [online]. Vystaveno 26. 05. 2004. Dostupné z: <<http://www.dbsvet.cz/view.php?cisloclanku=2004052601>>.
- [8] Dressler, V. Stavíme si vlastní prezentaci pomocí PHP, MySQL a CSS [online]. Vystaveno 25. 07. 2007. Dostupné z: <<http://www.owebu.cz/obecne/vypis.php?clanek=1136>>.