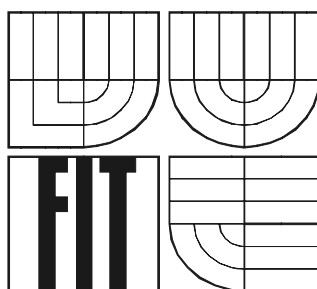


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



Diplomová práce

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Pavla Očenáška.

Další informace jsem čerpal z uvedené literatury.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

Abstrakt

Tento projekt si klade za cíl navrhnout a vytvořit internetový obchod nabízející funkce, které jsou pro tento druh aplikace běžné, nepostradatelné a funkce, které zajistí bezproblémový a bezpečný chod.

Základem návrhu jsou UML modely (ER diagram, USE CASE diagram). Tyto modely slouží pro ucelený návrh schématu nejen databáze, ale i struktury celého obchodu. Obchod bude nabízet standardní administrační funkce, které dovolí administrátorům obchod řídit. Uživateli bude nejen nabízeno zboží, ale bude mít i svůj účet a bude moci sledovat historii svých operací. Systém bude používat jazyky PHP, html, JavaScript, SQL, dále pak technologie MySql databáze, Apache server a Rewrite modul serveru Apache.

Klíčová slova

Obchod, databáze, MySql, php, Apache, rewrite, UML, html, JavaScript.

Abstract

The objective of this project is to design and create internet shop offering functions, which are common and essential for this type of application. It will also offer functions, which will ensure troublefree and safe run. The base of design are UML models (ER diagram, USE CASE diagram). This models will be created for compact design of database, and structure of whole application.

This shop will offer standard administrative functions, which will allow administrators to control the shop. Users will be not only offered by goods, but also will have their own account, where they will be allowed to check the history of their operations. The system will use PHP, html, JavaScript and SQL languages. It will be based on MySql database, Apache server and Rewrite modul of Apache.

Key words

Store, database, MySql, php, Apache, rewrite, UML, html, JavaScript.

Obsah

1	Úvod.....	8
2	Elektronický obchod.....	9
2.1	Definice.....	9
2.1.1	Typy e-obchodu.....	10
2.1.2	Srovnání s klasickým obchodem.....	10
3	Analýza požadavků.....	12
3.1	Neformální specifikace.....	12
3.2	Internet.....	13
3.3	Výběr technologie	14
3.3.1	PHP.....	14
3.3.2	HTML.....	15
3.3.3	SQL.....	15
3.3.4	MySql.....	16
3.3.5	Apache.....	16
3.3.6	Rewrite modul.....	16
3.3.7	JavaScript.....	16
3.3.8	CSS.....	18
3.3.9	UML.....	19
3.4	Požadované funkce.....	20
3.4.1	Funkce směřované na zákazníka.....	20
3.4.2	Funkce směřované na administrátora.....	20
3.5	Bezpečnost.....	21
3.6	Zákonné povinnosti e-obchodu.....	21
3.6.1	Testy e-obchodů.....	22
3.7	Programové prostředky.....	22
4	Návrh systému.....	24
4.1	Use case model	24
4.2	ER model.....	24
4.3	Struktura obchodu.....	24
4.4	Layout.....	25
4.4.1	Rámcový layout.....	25
4.4.2	Tabulkový layout.....	26
4.4.3	Div layout.....	26
4.5	Adresace jednotlivých stránek.....	26
4.6	System autentizace.....	27
4.7	Navigace.....	27
4.8	Kategorie zboží.....	28
4.9	Rozdělení zdrojového kódu do souborů.....	28
5	Implementace.....	30
5.1	Přepřacování ER modelu na schéma databáze.....	30
5.2	Implementace.....	30
5.2.1	Layout.....	31
5.2.2	Bezpečnost PHP.....	31
5.2.3	Struktura stránek.....	31

5.2.4	Kontrola autenticity osoby.....	32
5.2.5	Kontrola vstupů z formulářů.....	33
5.2.6	Nákupní košík.....	33
5.2.7	Kategorie a jejich zpracování do menu.....	34
5.3	Modul rewrite.....	35
5.4	Tvorba náhledů obrázků v PHP.....	37
5.5	Komunikace se zákazníkem.....	38
5.6	Design.....	39
5.7	Normy.....	40
6	Testování a optimalizace webu.....	41
6.1	Validita.....	41
6.2	Funkčnost odkazů.....	41
6.3	Přístupnost.....	42
6.4	Datová velikost.....	44
6.5	Použitelnost.....	45
6.6	Optimalizace pro vyhledávače.....	46
7	Propagace webu.....	48
7.1	Zavedení do vyhledávačů a katalogů.....	48
8	Srovnání s existujícími obchody.....	49
9	Návrh rozšíření.....	50
9.1	Obecné.....	50
9.2	Část administrátora.....	50
9.3	Část uživatele.....	50
10	Závěr.....	52

1 Úvod

Jedním z hlavních důvodů vzniku internetových obchodů je rostoucí množství potencionálních zákazníků, kteří mají připojení k internetu. Na rozdíl od kamenného odchodu může ten virtuální nabízet mnohem více produktů a tyto produkty ani nemusí fyzicky skladovat.

Na internetu je dnes nabízeno prakticky cokoli. Ne každý obchod dokáže zákazníky přilákat, ať už poprvé nebo opakovaně. Doba, kterou má internetová prezentace na to, aby upoutala a udržela pozornost uživatele se pohybuje okolo deseti sekund, což značně ztěžuje situaci a ne každý se s tím vypořádá. Existuje mnoho online obchodů prodávajících nepřeborné množství zboží. Obchody poskytují množství funkcí a služeb na různé úrovni praktičnosti.

Cílem tohoto projektu je navrhnout a zrealizovat obchod, který dovolí zákazníkovi zapomenout, že je ve virtuálním světě a bude nabízet funkce blížíící se světu reálnému. Těmi zásadními funkcemi jsou mimo jiné výběr zboží podle kategorií, vyhledávání dle zadaných parametrů, detailní popis produktu nahrazující fyzický kontakt se zbožím a co nejvíce fotek produktů. Projekt byl započat jako ročníkový projekt. Jeho součástí byl návrh aplikace tj. sběr požadavků, USE CASE a ER modely.

2 Elektronický obchod

2.1 Definice

„Pod pojmem elektronický obchod rozumíme podnikání elektronickými prostředky. To zahrnuje obchodování se zbožím hmotným i nehmotným (potraviny, hudební nahrávky, atd.) i službami (informačními, právními, atd.). Zahrnuje všechny kroky od reklamy přes uzavření smlouvy, její plnění, a to včetně poprodejní podpory a služeb. Z právního hlediska jde zásadně o projevy vůle. Právní úkony, směřující k uzavírání smluv, které jsou realizovány pomocí počítačových sítí.

Podle definice OECD zahrnuje elektronický obchod jakékoliv obchodní transakce, které provádějí osoby fyzické i právnické, přičemž tyto transakce jsou založeny na elektronickém zpracování a přenosu dat. Pod pojmem elektronického obchodu OECD tedy spadá i elektronická výměna takových informací, nařízení orgánů, podporující řídicí nebo účastníci se klasických neelektronických obchodů.

Podobně WTO (Světová obchodní organizace) pod e-obchod zahrnuje jak výrobky, které jsou podávány a placeny přes internet, ale doručovány ve hmotné podobě, tak produkty, které jsou přes internet doručovány v podobě digitální. Tyto široké definice elektronického obchodu tedy zahrnují transakce pomocí telefonu, faxu, televize, elektronických plateb a transferu platebních prostředků, které jsou dlouhodobě zavedeny.

Právo EU (směrnice 2000/31 o elektronickém obchodu; ÚL 2000 L 178/8) samotný pojem e-obchodu nedefinuje, ale nahrazuje jej pojmem služby informační společnosti (information society services), na jehož definici odkazuje na čl. 2(a) tzv. transparentní směrnice 98/34: služby informační společnosti jsou definovány pozitivně i negativně.

Pozitivní definice: jde o služby poskytované zpravidla (ale ne nutně) za úplatu, na dálku, prostřednictvím elektronického zařízení pro zpracovávání a ukládání dat a na individuální žádost příjemce služby. Taková služba např. zahrnuje on-line prodej zboží, poskytování vyhledávačů a přístup k datům, přenos informací po síti, push služby (např. on-line noviny předem označené vyhledávačem) poskytování připojení k síti, ukládání informací, poskytování komerčních a propagačních informací e-mailem.

Negativní definice: Z věcné působnosti směrnice (a tedy zejména z povinnosti členských států podporovat rozvoj e-aktivit podle směrnice 2000/31) jsou však vyloučeny pojmově: prodej zboží či poskytování služeb off-line (např. hlasové telefonní služby, právní či lékařské poradenství anebo přímý marketing po telefonu či faxem), televizní a rozhlasové vysílání.“ [13]

2.1.1 Typy e-obchodu

Pojem e-obchodu [13] zahrnuje poskytování zboží a služeb elektronickou formou prostřednictvím sítě dalším společností (tzv. Business to Business . B2B), anebo také konečným zákazníkům (tzv. Business to Customer . B2C). V praxi můžeme vidět nejméně tři kategorizace e-obchodu:

Podle účastníků transakcí

- B2B (průmyslové), obchodování mezi obchodníky navzájem. Jsou nejstarší a z finančního hlediska nejvýznamnější formou obchodu.
- B2C (spotřebitelské), typické smlouvy spotřebitelské s koncovými zákazníky. Jsou z pohledu laické veřejnosti asi nejrozšířenější formou e-obchodu.
- C2B obchody, kdy zákazník oslovuje podnikatele, např. kdy definuje zboží a maximální cenu a vyzývá obchodníky k podání nabídek na uzavření smlouvy. Nejméně rozšířené.
- C2C obchod mezi dvěma neobchodníky/nepodnikateli, tj. mezi spotřebiteli navzájem, nejtípcičtější jsou e-aukce.
- B2A, C2A (business/customer to authority) - správní.

Podle otevřenosti použitého média

- Uzavřené transakce: jde o e-obchod po uzavřených sítích mezi omezeným okruhem partnerů (kluby, firemní sítě, platformy B2B).
- Otevřené transakce: e-obchod mezi otevřeným počtem účastníků, kam patří e-obchod elektronický obchod prostřednictvím internetu, m-obchod elektronický obchod prostřednictvím mobilních telefonů.

Podle způsobu plnění

- Přímé e-obchody, které jsou plně on-line: objednávka, placení a zároveň dodávka nehmotných statků (software, informace, mediální produkty) se uskutečňuje výhradně prostřednictvím elektronických prostředků v reálném čase.
- Nepřímé e-obchody: on-line probíhá pouze objednávka či uzavření smlouvy, nejvýše ještě platba, ale plnění smlouvy ze strany prodávajícího, tj. dodávka zboží se děje tradičními prostředky, toto je případ tohoto obchodu.

2.1.2 Srovnání s klasickým obchodem

Prostor: pro klasický obchod je potřeba budova či místnost, kde by byl provozován. Za tento prostor se platí. Obchod by měl být na frekventovaném místě a právě takové prostory jsou dost drahé. Vše co je třeba pro virtuální obchod je sklad zboží, klidně někde v ústraní a kancelář, ze které bude

provozován, rovněž mimo centra měst..

Zboží: klasický obchod prodá jen zboží, které má vystavené. Málokdy se stane, že zákazník chce něco objednat a je ochoten na to čekat. U internetového obchodu se nemusí nabízet pouze zboží, které je momentálně na skladu. Ve většině případů lze produkty doobjednat u dovozce, nebo přímo u výrobce. Jen u některých produktů se stane, že vše se musí objednat před danou sezónou a není šance v průběhu roku doobjednat chybějící kusy. Toto platí zejména u zimního oblečení.

Cena: cena produktů u kamenného obchodu je v drtivé většině zákonitě vyšší. Do ceny se promítnou náklady za výše uvedené prostory, prodavače, atd. Obchod internetový si může dovolit cenu o tyto položky snížit. Zákazník za to ovšem platí pouze omezeným množstvím možností jak produkt vybrat. Chybí fyzický kontakt. Cena poštovního je poměrně nízká, takže stále vítězí obchod virtuální.

Personál: klasický obchod si vyžaduje větší nároky na osoby, jenž jej budou udržovat v chodu. Počítá že mezi ně manažer případně manažeri a obsluha. Obchod internetový nepotřebuje prodavače. Stačí pokud se o něj starají dvě a více osob, v závislosti na velikosti.

Zákazníci: obchod klasický se co se týče zákazníků omezuje na určitou oblast. Málokdy do obchodu zavítá někdo zdaleka. Internet je celosvětová síť. Dalo by se říct, že zde může nakupovat kdokoli a kdekoli. V praxi to samozřejmě neplatí. Málokdo si asi koupí nějaký produkt nabízený obchodem z jiného státu, speciálně pokud se jedná o sportovní oblast. Výjimkou jsou speciální produkty, které nejsou běžně k dostání na domácí scéně. Obchod na internetu může navštívit například tisíc lidí denně. Poměr lidí, kteří si něco koupí a těch, co se jen podívají je samozřejmě menší.

Sortiment: klasický obchod se většinou specializuje na poměrně úzkou oblast zboží. Nemá na to prostory a je to nevýhodné prodávat příliš různorodé produkty. Zákazník si může sice koupit více typů zboží ovšem je ochuzen o pestrý výběr. Na internetu není problém provozovat obchody podobné obchodním centrům. Zákazník tak není ochuzen o možnost výběru a vše je na malém prostoru jeho obrazovky, nemusí dlouze procházet nákupní centra.

3 Analýza požadavků

Pro správný návrh systému je podrobná a úplná analýza požadavků nezbytná. Tato část je prvním krokem k úspěšnému vytvoření obchodního systému. Zahrnuje v sobě sběr co nejvíce dostupných informací. Z těchto informací si pak tvůrce musí vybrat to, co se mu hodí pro korektní návrh systému. Informací je všude kolem mnoho, jen je třeba je najít, projít a zpracovat. Jako zdroje jsem použil internet, literaturu týkající se daného problému a konzultace s případnou osobou, která bude obchod provozovat.

3.1 Neformální specifikace

Cílem tohoto projektu je vytvořit internetový obchod. Tento obchod bude administrátorovi poskytovat naprostou svobodu tvorby obsahu, tj. bude moci přidávat a odebírat zboží, bude definovat akce, novinky a tipy, bude spravovat objednávky a bude přidávat aktuální informace, ať už ty týkající se přímo obchodu, nebo jakékoli informace, které donutí zákazníka, aby se vrátil. Systém bude dovolovat běžnému uživateli objednat zboží, bude moci zboží vyhledávat a bude mít svůj účet, kde bude kontrolovat stav svých objednávek. Uživatel se bude muset nejprve zaregistrovat do systému, než si bude moci objednat nějaké zboží.

Nefunkční požadavky

- Databázový systémem bude MySql.
- Systém bude provozován na serveru Apache.
- Obchod bude moci prohlížet kdokoli na běžném internetovém prohlížeči (IE, Opera, NS, Mozilla, atd.).
- Náhodně generovaná hesla budou délky 8-10 znaků. Tato délka zajistí dostatečnou bezpečnost a přesto si je zákazník bude moci snadno zapamatovat. Uživatel si následně bude moci heslo změnit podle sebe.
- Administrátor by měl pro zadávání popisu produktu znát základy jazyka html. V popisu produktu může graficky zvýraznit a rozložit popisované parametry.
- Systém bude provozován na komerčním web hostingu, tudíž zálohu databází bude provádět poskytovatel. Běžná je denní záloha.
- Na hardware si systém neklade žádné speciální požadavky.

Funkční požadavky – zákazník + administrátor

- Registrace do systému.

- Změna osobních údajů uložených v systému.
- Kontrola stavu objednávek.
- Přihlášení do systému.
- Vyhledávání produktů podle daných parametrů.
- Přidání komentáře k produktu.
- Přidání produktu do košíku.
- Smazání a upravit položky košíku.
- Objednání zboží.
- Přidání příspěvek do fóra.

Funkční požadavky – administrátor

- Správa produktů.
- Správa tipů, novinek, akcí.
- Správa aktualit.
- Editace velikostí, barev, značek.
- Správa objednávek.
- Správa osob registrovaných do systému.
- Editace kategorií.
- Administrátor bude přistupovat do souboru s konfigurací, aby mohl měnit parametry, které nejsou uloženy v databázi.

3.2 Internet

Internet je rozsáhlý zdroj informací. Prvním krokem, který jsem učinil je zkoumání existujících obchodů. U nich jsem zjišťoval způsob neverbální komunikace mezi stránkou a zákazníkem. To znamená způsob jakým je zboží nabízeno a jak obchod přesvědčí, že je tento produkt ten pravý, že právě tento si musí koupit. Produkty musí být děleny do kategorií, které mohou být zanořeny ovšem maximální hloubka by měla být dva až tři stupně. Při větší hloubce zákazník ztrácí přehled o tom, jak je zboží děleno. Pro obchod se sportovním zbožím, navíc konkrétně s lyžařským, pro který je tento obchod zamýšlen, bude potřeba maximálně dvou stupňů. Přesto jsem se rozhodl zanoření udělat libovolné.

Detail produktu musí obsahovat co nejvíce dostupných informací o produktu. Zákazník nemá možnost si výrobek prohlédnout, a tak si alespoň může o něm přečíst. Každý produkt by měl mít alespoň jednu fotku. Kupovat něco u čeho není fotka je skoro jako kupovat zajíce v pytli. Zákazník tak raději nakoupí u konkurenčního obchodu, nebo v tom lepším případě si musí informace opatřit jinde a

pokud máme výhodnější cenu, tak možná nakonec výrobek koupí u nás. Produkty by měly být předhazovány zákazníkovi skoro na každé stránce. Proto je potřeba vytvořit systém novinek, tipů, akcí, nejprodávanějších produktů, posledně koupených, nejzobrazovanějších, atd. Z ostatních obchodů je možné vyčíst, jaké informace je potřeba mít k produktu uloženy v databázi. Stejně tak lze učinit u osoby, novinek, akcí, atd. I struktura stránek lze do určité míry naplánovat s pomocí jiných obchodů. Pokud zrovna nepoužívají rewrite modul, či jinou metodu jak adresu učinit přehlednější.

3.3 Výběr technologie

V této části budou zmíněny použité technologie a jejich hlavních rysy. Celý obchod by jistě šel postavit za pomoci Java server pages (JSP). Tato technologie se na tento typ aplikace příliš nepoužívá. Obchod manipuluje se základními funkcemi a možnosti jazyka Java by ani zdaleka nevyužil. Jazyk ASP je asi o něco rozsáhlejší než jazyk PHP a pro podobný typ aplikace je hojně využíván a byl by jistě vhodným řešením.

3.3.1 PHP

Zkratka PHP [1] je odvozena od akronymu. PHP: Hypertext Preprocessor. Jedná se o open-source technologii, což znamená, že je za dodržení určitých podmínek volně šiřitelná jak v binární tak ve zdrojové podobě. PHP je tzv. server-side jazyk. To znamená, že na rozdíl například od JavaScriptu je kód PHP skriptu zpracován serverem a uživateli přijde hotová html stránka. PHP zdrojový kód je v podstatě html kód obsahující jazyk PHP v speciálních značkách (`<?php zdrojový kód PHP?>`). Podstatnou výhodou je, že osoba prohlížející stránky nevidí zdrojový kód, vidí pouze hotový výsledek. Tato vlastnost ovšem není postačující ochranou zdrojových kódů. Existují nástroje pro stažení všech souborů vyskytující se na daném serveru.

PHP je jazyk poměrně jednoduchý. Začínající programátor se v něm rychle zorientuje. Na druhou stranu poskytuje pokročilé funkce, které z něj dělají univerzální webový jazyk. PHP rovněž nabízí možnost OOP, tedy objektově orientovaného programování. Tato verze není ještě zcela standardizována. PHP podporuje komunikaci s dalšími serverovými technologiemi jako LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (pro Windows), atd. Rovněž podporuje práci se sockety, dokáže zpracovat XML.

Hlavními oblastmi použití jsou

- Skriptování na straně serveru. Asi nejčastější užití. Je k tomu potřeba server, php parser, což je skript, který zpracuje kód jazyka a vrátí výstup. Dále je potřeba prohlížeč, k prohlížení výsledného kódu, nejčastěji html.
- Jazyk se dá užít i bez účasti serveru a to pro skriptování v příkazové řádce. K tomu je pouze

potřeba php parser.

- Ne příliš časté užití je psaní klientských aplikací. Existuje rozšíření PHP-GTK, které možno volně stáhnout z adresy <http://gtk.php.net>.

Přenositelnost

PHP je podporováno většinou operačních systémů jako jsou Linux, Unix like systémy, HP-UX, Solaris a OpenBSD), Microsoft Windows, Mac OS X, RISC OS, atd. PHP je dnes již podporováno mnoha serverovými systémy jako Apache, Microsoft Internet Information Server (IIS), Personal Web Server, Netscape a iPlanet servers, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, atd.

Podpora

Nemalou výhodou je bohatá škála databázových serverů podporovaných jazykem PHP. Jsou jimi Adabas D, Ingres, Oracle (OCI7 and OCI8), dBase, InterBase, Ovrimos, Empress, FrontBase, PostgreSQL, FilePro (read-only), mSQL, Solid, Hyperwave, Direct MS-SQL, Sybase, IBM DB2, MySQL, Velocis, Informix, ODBC, Unix dbm.

3.3.2 HTML

Zkratka HTML znamená HyperText Markup Language. Jedná se o jednoduchý značkovací jazyk používaný pro tvorbu webových stránek. Řídící tagy (značky) jsou uzavřeny do lomených závorek. Je to konečný výstup složitějších jazyků jako PHP a ASP.

Jednoduchý příklad

```
<HTML>
<head>
  <title>Název dokumentu</title>
</head>
  <body>
    Obsah dokumentu.
  </body>
</HTML>
```

3.3.3 SQL

SQL je zkratka pro Structured Query Language, tedy strukturovaný dotazovací jazyk. Vznikl v rámci výzkumného projektu IBM. Užívá se především ve spojení s dolováním dat z databáze a k instrukcím pro zpracování dat.

3.3.4 MySql

MySql [2] je systém pro správu relační databáze. Data v takové databázi jsou uspořádána do tabulek, které jsou propojeny vazbami. Je to opět open-source systém. Kdokoli si jej může stáhnout z internetu. MySql je distribuováno pod GPL, tedy General Public Licence. To znamená, že kdokoli jej může užívat a modifikovat. Modifikovanou verzi pak musí poskytnout i se zdrojovým kódem. Pokud někdo potřebuje užít systém bez licence GPL musí si zakoupit licenci komerční. MySql je velice rychlý, spolehlivý a snadný na použití. V době vzniku byl nejrychlejším dostupným systémem. Jedná se o klient/server systém. Pracuje s vlákny. Je podporován mnoha aplikacemi.

3.3.5 Apache

Apache je jednoduchý a výkonný server. Apache vznikl z prvního www serveru nazvaného NCSA. Do tohoto serveru začali nadšenci doplňovat záplaty, kvůli výkonu a bezpečnosti. Z tohoto systému vznikl postupem času nový www server. Apache je dostupný zdarma. Na jeho vývoji se může kdokoli podílet. V současné době jej používá 50% serverů. Pro Apache je dostupné množství modulů. Je možno na něm pracovat s CGI skripty.

3.3.6 Rewrite modul

Pro server Apache existuje mnoho doplňujících modulů. Jedním z nich je modul rewrite. Modul rewrite nabízí mnoho užitečných funkcí. Lze použít v následujících případech

- Přesměrování uživatele na jinou stránku.
- Přepsání špatně čitelných, nepřehledných, nepraktických adres na přehledné adresy.
- Skrytí technologie stránek, tedy koncovek .html .php, atd.
- Odstranění parametrů stránek, vyhledávače je neradi akceptují.
- Modul číhá na serveru a jakmile mu sedí adresa na kterou má podmínku tak příkaz přepsání uplatní.

Jednoduchý příklad

```
RewriteEngine On
RewriteRule ^([^/]+)/strana-([0-9]+)/$
/kosik/?kategorie=$1&strana=$2 [L,QSA]
```


3.3.7 JavaScript

JavaScript [4] je programovací jazyk, který dovoluje vytvoření dynamických html stránek. Existují dva základní druhy, Client-side JavaScript a Server-side JavaScript. Oba mají společné jádro a každý z nich obsahuje navíc knihovny a předdefinované objekty.

Client-side verze je obsažena přímo v HTML souboru a je interpretována prohlížečem. Server posílá klientovi (prohlížeči) kompletní zdrojový kód. Prohlížeč tento kód čte od začátku do konce a zpracovává jej. Výsledky zobrazuje přímo na obrazovku. Užití tohoto typu je na kontrolu vstupů formulářů, reakce na vstupy uživatele, kliky, pohyb myši atd.

Jednoduchý příklad

```
<HEAD>
  <TITLE>Titulek</TITLE>
  <SCRIPT>
    function alert(msg) {
      alert(msg);
    }
  </SCRIPT>
</HEAD>
<BODY>
<a href="javascript:alert('ahoj')">Klikni</a>
</BODY>
```

Server-side skripty jsou kompilovány na serveru dříve, než jsou poslány uživateli. Tento je rovněž obsažen v HTML stránce. Užívá se pro připojení k relační databázi, sdílení informací, přístup k systému souborů klienta. Kompilovaný Server-side JavaScript může obsahovat Client-side verzi. Kód je přeložen do bitekódu a jsou interpretovány serverem s runtime enginem.

Jednoduchý příklad (hangman.js) [16]

```
function Substitute( guess, word, answer) {
  var result = "";
  var len = word.length;
  var pos = 0;
  while( pos < len ) {
    var word_char = word.substring( pos, pos + 1 );
    var answer_char = answer.substring( pos, pos + 1 );
    if ( word_char == guess ) result = result + guess;
  }
}
```

```

        else result = result + answer_char;
        pos = pos + 1;
    }
    return result;
}

```

Jednoduchý příklad - pokračování (hangman.htm) [16]

```

<HTML> <HEAD> <TITLE> Hangman </TITLE></HEAD>
<BODY> </H1> Hangman </H1>
<SERVER>
    if (client.gameno == null) {
        client.gameno = 1
        client.newgame = "true"
    }
</SERVER>
You have used the following letters so far:
<SERVER>write(client.used)</SERVER>
<FORM METHOD="post" ACTION="hangman.htm">
<P>
What is your guess?
<INPUT TYPE="text" NAME="guess" SIZE="1">
...
</BODY></HTML>

```

3.3.8 CSS

CSS neboli Cascading Style Sheets [9] se živají pro grafické zpracování html dokumentu. Jsou nadstavbou jazyka HTML. Tyto styly umožňují každému prvku stránky html přiřadit vlastnosti definující jeho grafický vzhled. Tento vzhled závisí na druhu a verzi prohlížeče či jiného zařízení, který je pro prohlížení dokumentu použit. Styly také umožňují určit vlastnosti pro třídy a identifikátory, jejichž názvy odpovídají názvům tříd v CSS definici stylu.

CSS styly mohou být součástí dokumentu jehož vlastnosti definují. Lze tak učinit přímo v html tagu nebo styl definovat a pojmenovat na začátku dokumentu a dále použít jen jeho název. Tento způsob je výhodný v případě, že potřebujeme určit jen několik vlastností. V opačném případě by se dokument s CSS definicemi stal nepřehledným.

Příklad

```
<p style="color:red; border:1px solid black;">text odstavec</p>
```

Dalším způsobem je psát definice do samostatného souboru a tento pomocí tagu link připojit k html souboru. Tento způsob zajistí přehlednost a škálovatelnost dokumentu. Při tomto způsobu je možno definovat mnoho prvků a do výsledného souboru zahrnout několik style sheetů.

Příklad definice stylu v externím souboru

```
.název_stylu{ width: 50ppx; height:600ppx;}  
Import stylu: <LINK REL="stylesheet" type="text/css"  
href="/styl.css">  
Použití stylu: <div class"název_stylu">text formátovaného  
textu</div>
```

Pokud pracujeme s CSS musíme počítat s tím, že mohou být různými prohlížeči interpretovány různě.

3.3.9 UML

UML [5] je zkratka Unified Modeling Language. Je to vizuální jazyk pro modelování systémů za užití diagramů a textu. Každé slovo názvu definuje oblast použití.

Unified

Object Management Group (OMG) a Rational Software Corporation spolu přinesly jednotný jazyk standardizovaný pro jednoznačné použití.

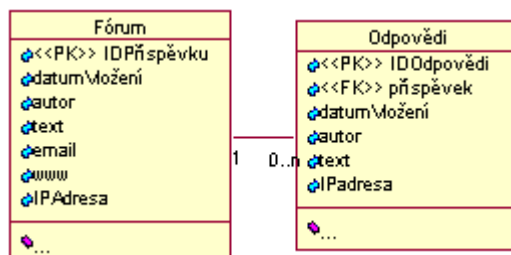
Model

Model značí reprezentaci subjektu. Je užita abstrakce subjektu, která umožňuje jednoznačnou komunikaci v týmu.

Language

UML definuje jazyk jímž mezi sebou komunikují členové týmu při tvorbě návrhu programového díla.

Jednoduchý příklad:



Obrázek 1.

3.4 Požadované funkce

Před samotným psaním zdrojového kódu je třeba si vytyčit rozsah funkcí. Internetový obchod lze postavit minimalistický a přesto bude prosperovat. Lze postavit rozsáhlou aplikaci. Tento obchod bude něco mezi.

3.4.1 Funkce směřované na zákazníka

Nebudou zde uvedeny samozřejmé funkce jako je prohlížení katalogu, výběru kategorie, přidávání produktů do košíku, vyhledávání, atd. Zákazník bude mít vlastní účet, kde budou jeho osobní údaje. Z toho vyplývá, že objednat zboží bude moci jen ten, kdo bude nejprve registrovaný. Existují obchody, kde jsou osobní údaje vloženy až při samotné objednávce. Tyto údaje pak putují s objednávkou a při dalším objednání zboží je třeba je znovu vyplnit. Zákazník bude mít tedy účet s jeho detaily. Tyto údaje bude moci modifikovat. Při objednání zboží budou do údajů objednávky automaticky nabídnuty již známé detaily. Objednání zboží se bude provádět v několika krocích. Zákazník bude moci nahlédnout do svých předešlých objednávek a do jejich obsahu, tzn. do počtu kusů jejich množství a cen. Zákazník bude moci komentovat výrobky a přispívat do fóra.

3.4.2 Funkce směřované na administrátora

Administrátor je osoba/osoby, které budou obchod řídit. Administrátor bude zajišťovat aktuálnost obsahu. Bude měnit produkty, jejich kategorie, značky. Bude Přidávat aktuality, měnit produkty zobrazované jako tipy, akce a novinky. Bude si zobrazovat objednávky a zařizovat odeslání požadovaného zboží a posléze měnit stav objednávek. Administrátor bude moci mazat příspěvky z komentářů a z fóra. To hlavně v případě výskytu nevhodných výrazů a reakcí.

3.5 Bezpečnost

Existují tři hlavní oblasti, kterým je nutno věnovat pozornost v oblasti zabezpečení.

- 1) Uložení dat na serveru. Je nutno nastavit vhodná oprávnění pro databázi MySQL. Obchod bude provozován na komerčním serveru, čili toto bude zajišťovat jeho administrátor. Veškerá hesla budou uložena jako hash, tj. Funkce md5() v PHP. Hesla takto uložená i kdyby někdo z databáze získal, tak z nich původní heslo nedostane a nebude se tak moci přihlásit. Heslo a jméno pro přístup k databázi budou uloženy v kořenovém adresáři serveru, kde nemá přístup nikdo jiný než ten kdo zná heslo na ftp účet, všechny ostatní stránky z www serveru obchodu lze pomocí příslušných aplikací stáhnout.
- 2) Je třeba chránit citlivé informace v části obchodu, kde provádí úkony administrátor nebo přihlášený uživatel. Je potřeba zabránit přístupu neoprávněným uživatelům. Typy uživatelů jsou tři a to nepřihlášený uživatel, přihlášený uživatel a administrátor. Jejich oprávnění se liší. Na začátku každé stránky bude kontrola zda je v sessions definován příslušný stupeň oprávnění, tedy hodnota 0,1 nebo 2 . Pokud ne, je další zobrazování ukončeno například funkcí die(), nebo exit().
- 3) Třetí oblast je zabezpečení dat během transakce. Toto nastává v okamžiku kdy se zákazník dostává k procesu ověřování, tedy předávání hesel, informací o dodávce. Je třeba použít zabezpečených transakcí. K tomu lze použít protokol SSL (Secure sockets layer), ke kterému je nutno připojit platný certifikát serveru. Předpona webové adresy se pak z prostého http://, změní na https://. Internetový obchod nemanipuluje s příliš citlivými daty. Pokud se někomu podaří vstoupit do cizího účtu tak maximálně objedná zboží pro někoho jiného, nebo mu změní údaje. Zákazník neplatí nikdy nic předem. Proto bezpečný přenos dat přes zabezpečenou síťovou vrstvu nebyl využit.

3.6 Zákonné povinnosti e-obchodu

Informační povinnosti vyplývající z § 53 Občanského zákoníku:

- Je nutné vystavit obchodní jméno a adresu provozovatele obchodu a identifikační číslo provozovatele.
- Povinné informování o nákladech na dodání zboží.
- Nezbytné poučení spotřebitele na právo odstoupení od smlouvy před jejím uzavřením a po dodávce objednaného zboží.
- Vzhledem k faktu, že tento projekt nemá zatím reálné užití nejsou tyto informace do stránek zahrnuty, nebo jsou uvedeny fiktivní hodnoty.

3.6.1 Testy e-obchodů

Existuje několik testů, zaměřených na e-obchody:

- <http://www.finexpert.cz/Zakony/AR.asp?ARI=3136>
- <http://www.computerworld.cz/cw.nsf/ID/916273DDF26660FBC125708F004A09B7>
- <http://interval.cz/clanek.asp?article=1100>
- <http://www.spotřebitele.cz/priloha/testinternet2004.pdf>

Na posledně jmenované adrese byly testovány následující položky:

- 1) Před odesláním objednávky měl spotřebitel možnost se na stránkách seznámit s plným zněním adresy (sídla u obchodních společností či bydliště u fyzických osob).
- 2) Před odesláním objednávky měl spotřebitel možnost se na stránkách seznámit s IČO provozovatele elektronického obchodu.
- 3) Před odesláním objednávky byl spotřebitel poučen o možnosti odstoupení od smlouvy bez stanovování omezujících podmínek nad rámec zákona.
- 4) V obchodních podmínkách nebyla uvedena u spotřebního zboží záruční doba kratší než 24 měsíců.

Shrnutí výsledků testu

Při prohlídkách internetových obchodů bylo zjištěno [13], že zdaleka ne všechny plní povinnosti dané zákonem o informování spotřebitelů.

- 3 % nezveřejňují svou adresu.
- 30 % nezveřejňuje své identifikační číslo.
- 42 % nepoučuje své zákazníky o jejich právu na odstoupení od kupní smlouvy.
- 12 % klame spotřebitele ohledně délky záruční doby.
- Všechna čtyři kritéria splnilo pouze 44 % testovaných obchodů.

Toto jsou základní povinnosti, které musí provozovatel obchodu splnit. Tento projekt není zatím reálně fungujícím obchodem, proto tyto parametry jsou buďto nahrazeny ukázkovými, nebo jsou vynechány vůbec.

3.7 Programové prostředky

Konceptuální návrh

Modely návrhu jsem vytvořil v prostředí Rational Rose Enterprise Edition, Release Version 2002.05.02. Je to aplikace sloužící k návrhům, nejen aplikací, za pomoci grafického jazyka UML.

Zdrojový kód

Zdrojový kód byl psán v prostředí HomeSite 5.5. Je do prostředí specializované pro zápis html kódu. Prostředí rovněž rozlišuje strukturu mnoha dalších jazyků, v mém případě JavaScript a PHP. Podobně jako například .NET nabízí HomeSite aktuální možnosti tagů, což v případě HTML značně urychluje psaní kódu.

Web server

Jako web server byl užit Apache 1.3.23. Je dostupný zdarma, jeho instalace je snadná a hlavně nabízí modul rewrite pro přepisování adres.

Databázový server

Pro databázi byl použit MySql-3.23.38 .Je to server dostupný zdarma, bez problémů spolupracuje s Apachem.

PHP

Pod server Apache byl nainstalován jazyk PHP verzi 4.0.

Prohlížeče

Korektní zobrazení všech elementů bylo zkontrolováno a odladěno v prohlížečích Mozilla Firefox 1.5.0.8, Internet explorer 6.0, Netscape 7.1 a Opera 7.11.

4 Návrh systému

Předchozí kapitola se zabývá sběrem všech dostupných informací za účelem rychlého a korektního návrhu systému. Tato kapitola se tedy bude zabírat praktickým postupem při návrhu konkrétního systému a všemi prostředky, které s tím souvisí.

4.1 Use case model

Use case diagram [17] slouží k zachycení aktérů a funkcí systému. Používá se jako základní náhled na systém. Lze z něj zjistit vstupy a výstupy systému. Výsledný model je dostupný v příloze A.

Use case diagram tvoří

- Případy použití (Use Cases) – definují funkce systému jako jsou reakce na poruchu, zpracování dat, atd.
- Aktéři (Actors) – jsou to osoby, které pracují se systémem.
- Vztahy (Relationships) – definují vztahy mezi jednotlivými případy použití.
- Asociace (Associations) – vazby mezi aktérem a případy použití.

4.2 ER model

„ER model je založen na chápání světa jako množiny základních objektů - entit (Entity) a vztahů (Relationship) mezi nimi. Popisuje data "v klidu", neukazuje, jaké operace s daty budou probíhat. Někdy se označuje také jako ERA – třetím základním prvkem modelu jsou atributy (Attributes).“ [6]

Tento model slouží pro vizualizaci struktury databáze. V příloze B je konečná verze ER modelu. V něm každý atribut třídy představuje jednu položku tabulky v databázi. Při vývoji ER a USE CASE modelů je třeba promyslet všechny detaily aplikace. Jakákoli chyba v nich představuje dlouhé zdržení při samotné implementaci.

4.3 Struktura obchodu

Samotná struktura obchodu se dá dost dobře stanovit z ER modelu. Je jasné, že skoro každá tabulka bude reprezentována jako adresář či podadresář v kořenovém adresáři webu. Základní snahou bylo dosáhnout co nejpodrobnější rozdělení pomocí adresářů tak, aby se rewrite modul použil jen tam, kde to jinak rozdělit nelze, tj. u dynamicky generovaných stránek s parametry. Všechny statické stránky, jako např. úvodní stránka nebo aktuality, jsou v adresářovém stromu umístěny tak, aby výsledná adresa jasně říkala, kde se návštěvník nachází. Příkladem jsou akce, které jsou v adresáři /produkty/akce/. Z této adresy je jasné, že jsou zobrazeny všechny produkty v akci. Každá tabulka která obsahuje akce jako přidat, smazat či upravit obsahuje u dané entity adresáře přidat, smazat,

upravit a proto se příslušná akce provede v logické adrese např. u již zmíněných akcí je to `/produkty/akce/pridat/`.

Podrobná adresářová struktura je důležitá i pro skrytí technologie, tedy odkazuje se na adresář, `/produkty/akce/`, nikoli na konkrétní soubor `/produkty/akce/index.php`. Jedná se o skrytí nedokonalé, jelikož pokud uživatel konkrétní soubor `index.php` doplní do adresy pak je použitá technologie prozrazena. Tomuto by šlo zabránit použitím `rewrite` modulu a adresy jako `/produkty/akce/` by byly přepsány třeba na `/produkty/akce.html`. V praxi se vlastně při volání adresy `/produkty/akce.html` podstrčí obsah stránky `/produkty/akce/`.

Adresy generované dynamicky není možné stanovit napevno do adresářů. Například adresa produktu je složena z jeho kategorie a názvu. Zde je nutné použít buďto parametrů za adresou jako `/produkty/akce/?akce=5`, nebo adresu přepsat na „hezčí“ verzi `/produkty/akce-5/`. Navigace po takto rozděleném webu je pak přehlednější a pro vyhledávače srozumitelnější a přijatelnější.

4.4 Layout

Layout je rozvržení stránky na jednotlivé bloky. Uživatel vnímá tyto bloky jako logické celky, které mají rozdílný účel. A proto je třeba pozičně a graficky rozčlenit. Existuje několik způsobů jak toho dosáhnout. Ideální metodu určit nelze. Dnes je nejpoužívanější `div` layout.

4.4.1 Rámcový layout

Dnes se od jeho použití ustupuje. Celá stránka je rozdělena do rámců. Některé rámce zůstávají statické, např. menu, v jiných se mění obsah.

Výhody

- x Menu je na místě, nemusíme rolovat v případě dlouhé stránky.
- x Uživatel může velikost rámců měnit podle potřeby.

Nevýhody

- x Problémy s tiskem. Přestože lze nastavit který rám chceme tisknout, mnoho uživatelů toho není schopna.
- x V adresním řádku se zobrazuje adresa menu a ne adresa konkrétní stránky. Nelze tak vytvořit odkaz na konkrétní stránku, nebo ji uložit do oblíbených položek.
- x Stránky v rámci nejsou indexovány některými vyhledávači.
- x Vyhledávače odkazují přímo na konkrétní stránku a ne na celou rámcovou strukturu.
- x V menu nelze zvýraznit aktivní položky, jelikož se obsah daného rámce zůstává stále stejný.
- x Některá mobilní zařízení rámce nepodporují.

4.4.2 Tabulkový layout

Jedná se o metodu, která je dnes spíše na ústupu. Celá stránka je jednou velkou tabulkou.

Výhody

- x Tabulka je starý koncept, proto je správně napsaná tabulka korektně zobrazována většinou prohlížečů.
- x Tabulky nabízejí bohaté grafické možnosti. Některé věci se dají vytvořit snadněji než v CSS.

Nevýhody

- x Závislost vzhledu na kódu. Tabulka říká co který element znamená. Dnes se dává důraz na to, aby značky definovaly pouze styl jakým se bude element vykreslovat a jeho vlastnosti jsou definovány nejlépe v externím souboru.
- x Stránka s tabulkou se zobrazí jakmile se načte její kompletní obsah.
- x Kód tabulky je neústupný. Tabulka má jasné rozdělení do sloupců a řádků. Někdy se tabulkový kód označuje jak vlakový, viz. „TR TD TD TD TD TR TD TD TD“.

4.4.3 Div layout

Celá stránka je rozdělena na elementy div a těm je definován CSS styl, buďto pomocí id nebo class.

Výhody

- x Je na rozdíl od tabulek oddělen obsah od vzhledu. Vzhled je definován v externím souboru s CSS stylem.
- x Předchozí výhoda má za následek větší přehlednost kódu.
- x Stránka se zobrazuje postupně jak je načítána, nečeká se až se načte celá.
- x Lze definovat jiné vzhledy pro různá zařízení.

Nevýhody

- x Prohlížeče zobrazují různě. Jednotlivé vlastnosti CSS stylu jsou podporovány různě.
- x CSS chybí, oproti tabulkám, některé vlastnosti.
- x CSS kód je poměrně nepřehledný. Z html kódu vzhled nepoznáme, tam je definovaný jen styl. Při pohledu na definici tohoto CSS stylu se jen stěží odhaduje, jak bude výsledek vypadat

4.5 Adresace jednotlivých stránek

Pokud máme rozděleny adresáře, pak víme, které adresy nelze stanovit napevno a tudíž je musíme

přepsat. V těchto případech je nutné předem stanovit parametry nutné pro zobrazení požadovaného obsahu jednotlivých stránek.

Konkrétně pokud uživatel v levém menu klikne na kategorii, čeká, že se mu zobrazí její obsah. Pro výběr produktů je tedy potřeba název nebo id kategorie, kterou chceme zobrazit. Výsledek pak vypadá takto: /produkty/bundy/strana-1/. Je zde jasné kde se uživatel nachází, v kategorii bundy. Produktů bude pravděpodobně mnoho a není možné je zobrazit na jednu stránku. Proto další parametr strana. Když uživatel listuje, tak se číslo strany mění a on stále ví, kde se nachází. Při užití vyhledávání se za adresu přidají další parametry, které se na začátku pro lepší přehlednost nezobrazí.

4.6 Systém autentizace

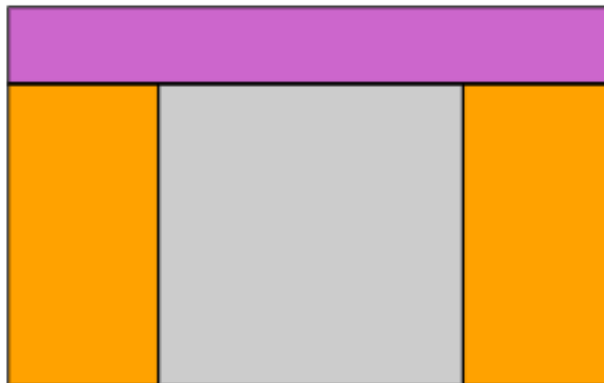
Pro autentizaci lze použít několik způsobů.

- HTTP autentizace: pro autorizaci jsou definovány serverové proměnné `$_SERVER['PHP_AUTH_USER']` a `$_SERVER['PHP_AUTH_PW']`. Při přístupu na soukromou stránku zjistíme, zda jsou tyto proměnné nastaveny a pokud ne zavolá se přihlašovací formulář ve kterém je uživatel vyplní.
- Užití sessions: vyplnění jména a hesla probíhá na zvláštní stránce, kterou sami vytvoříme. Pokud přihlášená proběhne úspěšně, pak se do sessions uloží jednoznačný identifikátor (session identifier) popřípadě i jméno. Odhlášení pak probíhá explicitně pomocí `session_destroy()`.
- Cookies: pro uložení jednoznačného identifikátoru lze užít též cookies, nebo databázi.
- Pro tento systém byl zvolen kombinovaný systém. Uživatel je jednoznačně identifikován svým session id. Uživatel ve vstupním formuláři uživatel vyplní email a heslo. V databázi se zkontroluje existence této dvojice. Pokud jsou vyplněny korektní údaje pak se do databáze uloží záznam do tabulky online a v session je nastaven stupeň uživatele (0 – nepřihlášený, 1 – přihlášený, 2 – administrátor). V tabulce online je uloženo session id, id uživatele, IP adresa a čas vstupu. Čas vstupu je při každém obnovení stránky aktualizován. Oproti předchozím systémům můžeme z tabulky v databázi zjistit počet uživatelů, kteří jsou online. I uživatelé, kteří nevyplnili přihlašovací údaje mají záznam v tabulce online a tím rozdílem, že id osoby je 0. Při každém přístupu do této tabulky jsou nejprve smazáni uživatelé, kteří neprovedli žádnou operaci po požadovaný čas.

4.7 Navigace

Navigace je kámen úrazu mnoha webů. Web může mít sebelepší a zajímavější obsah, ale pokud budou stránky nepřehledné a uživatel se v nich bude ztrácet, pak se asi už nevrátí. Snažil jsem se udělat strukturu logickou a předvídatelnou, což navigaci usnadní. Stránka je rozdělena do klasického konceptu, kde v levé části je menu, to zůstane viditelné na všech stránkách. V horní části je hlavička,

obrázek na pozadí a řádek zobrazující obsah košíku a údaje o přihlášeném uživateli. Ve středu je obsah konkrétní stránky a v pravé části doplňující informace. Pravá část je u některých stránek vypuštěna z důvodu většího nároku na prostor. Obrázek 2. zobrazuje celou situaci.



Obrázek 2.

Levé menu zobrazuje základní sekce obchodu. Toto menu se mění podle toho zda je uživatel přihlášen, či ne. Střed stránky zobrazuje obsah aktuální sekce. Na začátku střední částí se zobrazí nadpis, který říká kde právě jsme. Pod nadpisem jsou zobrazeny odkazy na nadřazené sekce pokud nějaké jsou. Někde bylo zapotřebí při návratu zpět odeslat parametry přes metodu POST, pak bylo použito jako odkaz zpět tlačítko.

4.8 Kategorie zboží

Kategorie zboží tvoří základ levého menu a zároveň celého katalogu. Každý produkt bude patřit do jedné kategorie. Každá kategorie může mít jednu nadřazenou kategorii. Tato struktura, průchodem, vytvoří levé menu. Adresa kategorie bude složena z jejího názvu a z názvů nadřazených kategorií. U těchto názvů budou nahrazeny české znaky a interpunkce znaky '_' a '-'. Adresa konkrétního produktu bude obsahovat celý název kategorie plus název produktu s jeho identifikačním číslem.

4.9 Rozdělení zdrojového kódu do souborů

Zdrojový kód každé stránky by měl být přehledný a ne příliš dlouhý. Existuje možnost, kdy celá stránka je v jednom souboru a případné operace se definují parametry v adrese. Tento způsob se neuzívá. Výsledek je nesrozumitelný a těžko udržovatelný.

Další možností je rozdělit strukturu do mnoha souborů a ty zahrnovat do jednoho hlavního a opět operace provádět pomocí parametrů v adrese stránky. Tento způsob se užívá. Při dodržení pravidel se dá udržet přehledným a udržovatelným.

Třetí a zdaleka ne poslední způsob je uložení každé stránky zvlášť. Oddělí se ty části, které jsou společné všem. V tomto konkrétním případě se jedná o záhlaví s definicí typu dokumentu, levé menu, pravý obdélník a zápatí, kde se uzavírají html tagy. Tyto společné části se uloží do samostatných souborů a jsou vkládány do každé stránky. Obsah konkrétní stránky pak obsahuje pouze střední část. Zbytek je vkládán pomocí php funkcí `include` nebo `require`. Tento způsob jsem zvolil v tomto projektu.

Výsledná stránka pak vypadá v zásadě takto:

```
<?php include 'horní část'?>
<?php include 'levá část'?>
    //střed s obsahem konkrétní stránky
<?php include 'pravá část'?>
<?php include 'spodní část'?>
```

Tento způsob dovoluje přiřadit každé stránce titulek. V souboru, kde je definována značka `title` se její hodnota bude zobrazovat jako proměnná a tato bude definována v každém konkrétním souboru.

5 Implementace

Pokud jsme provedli všechny v předchozí kapitole zmíněné operace, můžeme přistoupit k samotné implementaci. Je podstatné, aby byl návrh promyšlený do detailů. Pokud tomu tak není, pak je implementace zdlouhavá a je nutné se mnohokrát vracet a opravovat kód.

5.1 Přepřerování ER modelu na schéma databáze

Vstupem je ER model, který je výsledkem datového modelování. Při návrhu se snažíme vyhnout opakování informace (redundance), nemožnosti reprezentovat určitou informaci a složité kontrole integritních omezení. Stav kdy jsme se vyhnuli zmíněným problémům se nazývá tzv. 3NF neboli třetí normální forma. Přejít od návrhu ke konkrétní databázi musí projít přes následující kroky:

- 1) Entitní množiny + vztahy. Tyto údaje vyplývají z ER diagramu.
- 2) Transformace na prvotní schéma. Každá entitní množina bude reprezentovat tabulku a atributy takových entitních množin budou názvy sloupců. Následující fáze toto schéma zpřesní a upraví.
- 3) Převod do první normální formy (1NF) [17]. Pokud je struktura v první normální formě, pak neobsahuje složené atributy. Konkrétně například položka adresa se převede na město, ulice, čp, psč, stát.
- 4) Tabulky se převedou do druhé normální formy (2NF). Tabulky jsou v 2NF pokud splňují pravidla pro 1NF a každý atribut, který není primárním klíčem je na primárním klíči úplně závislý. Problém se týká tabulek, kde za primární klíč musíme zvolit více položek než jednu. V takovém případě většinou tabulku rozdělíme na dvě, protože dochází k redundanci dat.
- 5) Tabulky převedeme do třetí normální formy (3NF). Tabulky jsou v 3NF pokud splňují podmínky 2NF a zároveň žádný atribut, který není primárním klíčem, není tranzitivně závislý na žádném klíči.
- 6) Návrh fyzické organizace. Normalizované schéma tvoří již hotové tabulky. Je potřeba k nim zvolit indexy a datové typy.
- 7) Vytvoření databázových objektů. Konečná fáze, kdy se tabulky zapíší pomocí SQL do databáze.

5.2 Implementace

Nebudou zde zmíněny veškeré detaily implementace. Bude uvedeno jen to podstatné a zajímavé. V této fázi již máme hotovou databázi a přistoupíme k samotné implementaci. V první řadě je třeba zvolit postup jakým budou tabulky zpracovány. Tabulky jsou závislé a není možné, jen tak začít implementovat.

5.2.1 Layout

V první fázi byla napsána základní stránka, která se rozdělí na jednotlivé části jak bylo popsáno výše a tyto byly uloženy do samostatných souborů. Tím je hotový základní vzhled. Pro vytvoření struktury byla použita html značka div. Je to dnes nejužívanější způsob tvorby stránek. Značka na rozdíl od tabulek dovoluje nezávislé rozmístění elementů po stránce. Vlastnosti každé takové značky jsou pak uloženy v souboru s CSS styly. Poté byly zpracovány databázové tabulky a vytvořeny části, které jsou nezávislé na ostatních. Jsou jimi diskuzní fórum, komentáře k výrobkům, aktuality a systém autentizace uživatele. Po autentizaci byl vytvořen formulář pro registraci uživatelů.

V další fázi byly vytvořeny části, které na sobě závisí v posloupnosti od nezávislé k závislé. Příkladem je produkt. Pro zadání produktu je potřeba mít vytvořeny kategorie, značky, velikosti a barvy. Jistě by tento způsob šel obejít tak, že při zadávání produktu budou neexistující části zadávány napevno a postupně budou doplněny.

5.2.2 Bezpečnost PHP

Jak již bylo zmíněno jazyk PHP je poměrně jednoduchý pro začátečníky, přesto nabízí mnoho pokročilých funkcí pro pokročilé programátory. Při psaní stránek pomocí PHP je třeba dodržet několik základních bezpečnostních pravidel. Jistější je používat require místo include. Vyhnete se tak nepředvídatelným situacím v případě nepřítomnosti požadovaného souboru. Je třeba ošetřit vstupy typu `include $_GET['page']` před vložením zákeřného kódu (např. `stranka.php?page=/etc/passwd`), před vložením neexistující stránky a před vložením nepovolených znaků. Pokud pro sql dotaz používáme vstupy z pole GET, pak je jistější za SQL dotaz vložit příkaz `LIMIT n`. Vyhnete se tak situaci, kdy by uživatel v adrese stránky vstup SQL dotazu změnil. V SQL dotazu je lépe přesně specifikovat která pole chceme vybrat. Z dotazu `SELECT * FROM users` by bylo možno získat nejen třeba jméno uživatele, ale i jeho heslo. Je lépe užít `SELECT username FROM users`. Jistější je zamaskovat PHP pomocí rewrite modulu, například přepsáním přípony z `.php` na `.html`.

5.2.3 Struktura stránek

Stránky s PHP a HTML kódem, které obsahují formuláře a manipulují se vstupy uživatele mají podobnou strukturu. Na začátku každého souboru je v komentáři název souboru s kompletní cestou a popisem obsahu. Formát vypadá takto:

```
/*
* /produkt/akce/index.php          zobrazí akce
* zobrazí pro administrátora všechny akce
*/
```

* přidává, upravuje

*

*****/

Další část obsahuje definice globálních proměnných potřebných pro tento soubor. Část další zahrne do obsahu soubor s horním a levým panelem. První z jmenovaných mimo jiné obsahuje globální proměnné jako je stupeň návštěvníka (zda je přihlášen, nepřihlášen, nebo je to administrátor), datum, čas, IP adresu návštěvníka, domovský adresář, atd. Další část kontroluje oprávnění vstupu uživatele na tuto stránku, viz. následující kapitola. Pokud soubor obsahuje formulář, pak tento formulář vyžaduje kontrolu vstupních údajů. Tato operace se děje v další části. Při úspěšné kontrole je povolen vstup do další části, kde je manipulováno s databází. Část poslední obsahuje samotný formulář. Ten je zobrazen při prvním vstupu a v závislosti na typu prováděné operace je zobrazen i po úspěšné manipulaci s databází. Za tímto formulářem je zahrnut do obsahu soubor se zápatím, který uzavírá celou stránku.

5.2.4 Kontrola autenticity osoby

Zvolený systém kontroly autenticity je popsán v kapitole 4.6. Zde bude zmíněna pouze implementace. Všechny operace probíhají v souboru /admin/index.php. Tento soubor je zahrnut do každé stránky.

Začněme od konce tohoto souboru. Na konci zjistíme zdali je uživatel přihlášen. Vezmeme z tabulky online uživatelů ten záznam s odpovídajícím session id, a pokud je u tohoto záznamu stupeň uživatele větší než nula a tato osoba provedla obnovení stránky před překročením doby povolené pro setrvání v systému, pak zobrazíme jméno a příjmení této osoby. Nastavíme globální proměnné definující stupeň uživatele, jeho jméno a příjmení. Tyto proměnné jsou užity v ostatních stránkách, které do svého obsahu zahrnou horní část. V opačném případě zobrazíme odkaz na přihlašovací formulář.

Soubor /admin/index.php provádí i operace související s přihlášením a odhlášením návštěvníka. Odhlášení spočívá ve smazání záznamu z tabulky online uživatelů. Každý uživatel bude při vstupu na stránku zaznamenán do tabulky online. Bude mu přidělen stupeň 0, což je nepřihlášený uživatel. Přihlášení znamená změnu stupně u tohoto záznamu v tabulce online s odpovídajícím session id. Zaznamená se i čas jeho vstupu pro případné automatické odhlášení. Zároveň se změní datum posledního přihlášení v tabulce osoby.

Jak již bylo dříve zmíněno jsou definovány tři stupně uživatele. Uživatel je buďto nepřihlášen, přihlášen nebo je to administrátor. Systém obsahuje soubory na které mají povolen vstup jen některé skupiny nebo skupina. Na každé stránce, kde potřebuji omezit vstupní osoby tak kontroluji globální proměnnou \$ONLINE_USER nebo \$ONLINE_ID. \$ONLINE_USER definuje stupeň uživatele, hodnota 0 je rezervována pro nepřihlášeného uživatele, hodnota 1 náleží přihlášenému a hodnota 2

odpovídá administrátorovi. \$ONLINE_ID definuje identifikační číslo případného uživatele. ID všech uživatelů, kteří nemají vytvořen účet v systému je 0. ID ostatních je 1 a více.

V praxi restrikce všech kromě administrátora vypadá takto

```
if ( $ONLINE_USER <> 2 ) {  
    printError(44,3,false,false,true,''); // Neautorizovaný přístup  
    exit;  
}
```

5.2.5 Kontrola vstupů z formulářů

Existují dva nejpoužívanější způsoby pro kontrolu vstupních údajů z formulářů. Prvním z nich je použití JavaScriptu. Tento způsob před spuštěním akce formuláře zavolá skript, který zjistí hodnoty prvků formuláře. Pokud jejich hodnota neodpovídá očekávaným mezím tak zobrazí vyskakovací okno s varováním, případně přesune kurzor na chybné políčko formuláře. Tato metoda je rychlejší, ale pro uživatele není příliš pohodlná. Každá chyba je zobrazena zvlášť.

Druhý způsob využívá druhého průchodu stejnou stránkou. Kontrola se tedy provedena až při druhém vstupu na stránku kdy je definována proměnná odpovídající názvu odesílacího tlačítka formuláře. Pokud bylo před průchodem stránky stisknuto potvrzovací tlačítko formuláře na této stránce, tak se povolí vstup do části, kde se kontrolují vstupní data. Prvky formuláře mají buďto textovou, číselnou, nebo multimediální hodnotu. Každý tento typ si vyžaduje jinou kontrolu. První dva zmíněné kontrolují jednou funkcí, kde jako parametr zadán očekávanou délku vstupu a hodnotu udávající typ vstupu. Jediným multimediálním vstupem jsou obrázky. Ty kontrolují samostatnou funkcí. Kontrolují jejich velikost, typ a rozlišení.

Výsledkem každé kontroly je buďto prázdná hodnota v případě úspěchu, nebo chybová hláška. Tato hláška je uložena do proměnné. Celkový výsledek je v pořádku, pokud jsou všechny proměnné prázdné. Jestli se vyskytla chyba, pak se zabrání vstupu do částí, kde se manipuluje s databází a zobrazí se znovu vstupní formulář. Tentokrát se u konkrétního prvku formuláře zobrazí chybová hláška definovaná dříve zmíněnou proměnnou. Celý cyklus se opakuje dokud vstup není korektní. Chybové hlášky jsou zobrazeny funkcí. Pokud je chyba kritická, například při chybě databáze, uloží funkce záznam o chybě do databáze. Každá chyba je očíslována. Pokud se vyskytne, tak z databáze se zjistí její číslo a lze pak najít její konkrétní výskyt ve zdrojovém kódu.

5.2.6 Nákupní košík

Nákupní košík je definován v sessions. Existuje možnost, kdy je košík definován v databázi. Toto řešení je jistě v pořádku, ale přibude zde režie databáze a obsah košíku se musí mazat pokud uživatel

odejde bez toho, aniž by se odhlásil. Košík uložený v sessions má tu výhodu, že pokud uživatel odejde bez odhlášení, tak košík bude po vypnutí prohlížeče automaticky smazán.

Session je cesta jak lze v PHP uchovat data pro pozdější přístup. Každý návštěvník naší stránky dostane jednoznačný identifikátor (32 znaků). Tento se nazývá session id. Start každé takové session je proveden buďto implicitně pomocí příkazu `session_register()`, nebo explicitně za pomoci `session_start()`. Na stránce kdy byla session spuštěna je přístupné globální proměnná `$_SESSION`.

Prvek zboží, který si může uživatel objednat je identifikován jedinou číselnou hodnotou. Je to identifikátor kusu. Kus je vlastně produkt v určité velikosti a barvě. V sessions je tedy nutné udržovat jen tento identifikátor a počet kusů této hodnoty. Do session je uložena jedna proměnná. Tato proměnná je inicializována jako pole. Toto pole bude použito jako asociativní pole, to znamená, že indexem prvku bude identifikátor kusu a hodnota prvku bude množství kusu. Úprava množství pak spočívá ve změně hodnoty na příslušném indexu. Smazání znamená provedení funkce `unset()` na příslušnou položku pole. Vymazání celého košíku provede funkce `session_unregister()`. Zobrazení nákupního košíku spočívá v průchodu asociativního pole. Pro každý kus se v databázi najde jeho název. Název se zobrazí jako odkaz na detail produktu. Cena produktu se vezme rovněž z databáze.

5.2.7 Kategorie a jejich zpracování do menu

Každá kategorie v databázi může mít jednu nadřazenou kategorii. Tímto způsobem dosáhneme vytvoření struktury katalogu. Nezbytnou součástí tohoto systému je důkladná kontrola. Systém vymezuje jasná pravidla vkládání nových kategorií.

- Stupeň zanoření není nijak omezen. Z praktického hlediska je reálně použitelné zanoření stupně tři. Při vyšším zanoření se jednak rozbalené levé menu nemusí vejít do jemu vymezenému prostoru a jednak systém ztratí přehlednost.
- Kategorie, která obsahuje podřazenou kategorii nesmí obsahovat produkty. Vytvoření systému, kde by tato podmínka nebyla splněna by nebyl problém, ale struktura obchodu by byla zmatená a nepřineslo by to žádný užitek. Tato podmínka je kontrolována při vkládání produktu.
- Kategorie která obsahuje produkty nesmí být kategorií nadřazenou. Tato podmínka odpovídá předešlé. Podmínka je kontrolována při vložení nové kategorie.

Levé menu je vytvořeno dvojnásobným průchodem struktury. V prvním kroku projdu tabulku kategorie v databázi a získané hodnoty uložím do pole. Toto pole pro jednu kategorii, která bude mít podkategorii a tato podkategorie bude mít podkategorii bude mít následující strukturu:

```
KATEGORIE = array([1]{2}, [2]{3})
```

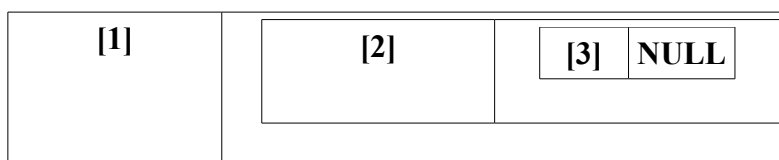
Graficky znázorněno

[1]	{2}
[2]	{3}

Pole bude asociativní a dvourozměrné. Každá položka reprezentovaná identifikátorem kategorie bude obsahovat pole, kde každý prvek bude tohoto pole reprezentovat identifikátor podřazené kategorie. V druhém průchodu z tohoto pole vytvořím n-rozměrné pole, čili jakýsi strom. Bude podobný předchozí struktuře, s tím rozdílem, že již každý prvek bude obsahovat pole svých podřazených kategorií, tedy i ten, který již podřazený je. Výsledek po zpracování předchozí tabulky bude vypadat takto:

```
KATEGORIE = array(1) [1]=>( array(1) [2]=>(array(1) [3]=>(array(0)) ) )
```

Graficky znázorněno



Tato konečná tabulka je zpracována pomocí rekurzivního průchodu do levého menu.

5.3 Modul rewrite

Rewrite modul je rozšíření serveru Apache, který nabízí mnoho užitečných funkcí.

Zapnutí nebo vypnutí procházení adresáře

Příkazem `Options Indexes` povolíme procházení adresáře pokud neexistuje výchozí soubor.

Přesměrování (i složitější)

Podrobněji viz. níže.

Zákaz přístupu z určitých IP adres nebo povolení jenom některých

```
# Zákaz IP adresy.  
deny from 215.123  
# Povolení IP adresy.  
allow from 215.123.156.189  
deny from all
```

Chybové stránky

Konkrétně to zajistí příkaz `ErrorDocument 404 /etc/404.html`. Pokud se tedy na

stránce vyskytne chyba 404, pak je uživatel přesměrován na stránku 404.html, kde se zobrazí příslušné varování. Bez tohoto příkazu by se zobrazila prázdná stránka s hlášením prohlížeče.

Kešování souborů

```
# Aktivace kešování
ExpiresActive On
# Čas expirace se nastaví na 6 minut.
ExpiresDefault 360
```

Kompresi přenášení souborů

```
AddOutputFilterByType DEFLATE text/css
AddOutputFilterByType DEFLATE application/x-javascript
AddOutputFilterByType DEFLATE text/html
```

Nastavení jiného výchozího souboru namísto index.html

Zajistí to příkaz `DirectoryIndex start.html index.html index.php`. Pokud tedy nechci, aby se při přístupu adresáře volal soubor index, nebo to z různých důvodů není možné, tak to změním.

K použití systému přepisování adres vedly následující důvody:

- Lepší čitelnost pro zákazníka. Zákazník na takové stránce bude mít snadnější přehled o tom, kde se právě nachází.
- Lepší akceptovatelnost pro vyhledávače. Je sice pravda, že některé vyhledávače dnes již parametry za otazníkem v url stránky akceptují úplně. Některé jako například Google akceptují pouze tři parametry. Některé vyhledávače ignorují parametr pokud se nazývá id. Přesto adresy bez otazníků zajistí maximální přijatelnost vyhledávači.
- Skrytí technologie. Při přepisování adresy nemá uživatel a vyhledávače možnost zjistit jaká technologie byla pro tvorbu stránky použita. Není tedy problém v případě potřeby technologii změnit bez toho, aby byly smazány statistiky vyhledávače. Zde nastává výjimka pro adresy, které nebyly přepsány pomocí rewrite modulu. Tyto sice neodkazují na konkrétní soubor (např. index.php), odkazují na adresář (např. /produkty/akce/), ale uživatel z nich může zjistit použitou technologii tím, že prostě zadá za adresu onen soubor index.php.

Všechny příkazy byly uloženy do souboru .htaccess. Existuje zde možnost uložit pravidla do souboru httpd.conf, ovšem tento je uložen v adresáři serveru apache. Není na komerčních serverech přístupný. Soubor .htaccess je výhodný v tom, že i běžný uživatel, který nemá přístup k nastavením serveru může změnit některá jeho nastavení. Soubor .htaccess je platný pro aktuální adresář a pro

všechny jeho podadresáře. Pokud se některá pravidla překrývají, pak je užito pravidlo bližší košile než kabát. Po změně pravidel není nutné server restartovat tak, jako je tomu u souboru httpd.conf. Z bohaté škály možností rewrite modulu byly využity jeho základní vlastnosti. Je to především přepisování adresy. Příklad, konkrétního použití s popisem.

Aktivuji přepisování příkazem

```
RewriteEngine On
```

Zapíšu přepisovací pravidla ve tvaru

```
RewriteRule ^([^\s/]+)/strana-([0-9]+)/[^\s/]+-([0-9]+)/kosik/$  
/kosik/?kategorie=$1&strana=$2&id_produkту=$3 [L,QSA]
```

Kde RewriteRule znamená, že se jedná o přepisovací pravidlo. Znak '^' uvozuje začátek řádku. Řádek podmínky ukončíme znakem '\$'. Podmínka se zapisuje jako regulární výraz, to znamená, že musíme všechny řídicí znaky uvodit zpětným lomítkem. V případě url adresy je to především tečka v adrese.

Uvnitř přepisovací podmínky můžeme skupiny znaků uzavřít do závorky. Poté je můžeme použít v adrese podstrčeného dokumentu jako jeho parametry. Zápis podmínky je obecný, a proto užíváme regulárních výrazů. Za podmínkou následuje mezera a za ní adresa stránky, kterou chceme podstrčit, nebo přesměrovat. V této adrese již nemůžeme použít regulárních výrazů. Adresu této stránky je lépe uvádět v celkovém tvaru, tedy celkovou cestu od kořenového adresáře. V podmínce jsme si do závorek uzavřeli výrazy, které chceme použít jako parametry. Tyto závorky jsou přístupné pod proměnnými \$x, kde x je pořadí závorky v podmínce zleva. Za adresou následuje skupina doplňujících parametrů v hranatých závorkách. V tomto ukázkovém příkladu je to L, což značí, že podmínky nechceme řetězit. Zabráníme tak případnému zacyklení. QSA je zkratka Query String Append. Znamená to, že pokud se v adrese, která odpovídá podmínce, a kterou jsme napsali do řádku v prohlížeči, vyskytnou parametry parametry za otazníkem, pak i tyto parametry budou připojeny za výslednou adresu.

5.4 Tvorba náhledů obrázků v PHP

Pokud budeme chtít k produktu vkládat fotky, musíme vyřešit jejich zobrazování na stránce. Jednou z možností je zobrazovat fotky v původní velikosti. Tento způsob by byl použitelný možná u jedné fotky v rozumné velikosti. Fotka by zabírala příliš mnoho prostoru a načítání stránky by se zpomalilo. Výhodnější způsob je každou fotku zobrazit jako zmenšený náhled (thumbnail). Můžeme po administrátorovi požadovat, aby zmenšeninu vytvořil a zadal společně s fotkou. Tento způsob by byl pohodlný pro zákazníka. Načítal by se jen malý náhled. Tento by fungoval jako odkaz na fotku v původní velikosti. Problém by měl administrátor, který by pro každou fotku musel náhled vytvořit a

správně dvojici do systému zadat.

Třetí způsob je kompromis mezi oběma předešlymi. Necháme administrátora, aby zadal jen fotku v původní velikosti a její náhled vytvoříme, až při zobrazení stránky pomocí funkcí PHP. Pokud tedy zákazník zobrazí stránku s náhledem, pak prohlížeč stahuje jen velikost několika kilobajtů, což je velikost náhledu. Až když uživatel na tento náhled klikne, tak se teprve bude načítat kompletní velikost desítek kilobajtů. V praxi to funguje tak, že do html tagu pro obrázek jako zdroj nezadááme přímo fotku, ale zadávám soubor, který vrací datový typ obrázku.

```
` a `</div>`. Tomuto tagu lze přiřadit mnoho vlastností definující vzhled jeho obsahu a jeho celkové chování. Uvedeme některé vlastnosti týkající se jeho chování:

- `Float:left, right, none` – znamená jestli se element bude zarovnávat vlevo, vpravo nebo bude pouze jeden element na řádku.
- `Padding` – je to odsazení textu od vnitřního okraje rámu. Problém této hodnoty je, že ji prohlížeče zpracovávají po svém. Internet explorer tuto hodnotu zahrnuje do celkové šířky obdélníku. Pokud tedy je pro tag div definována šířka 100px a padding nastavíme na 10px, pak celková šířka bude činit 110px. Naproti tomu Mozilla padding do celkové šířky neuvažuje, a takto vytvořený element by byl zobrazen jako 100 pixelů široký. S tímto je třeba při tvorbě stránky počítat. Je také nutné počítat s tím, že i jediný pixel může naprosto rozhodit vzhled dokumentu.
- `Margin` – odsazení z venkovní strany obdélníku. Udává se stejně jako padding v pixelech.
- `Border` – ohraničení elementu. Můžeme mu přiřadit barvu, styl, tloušťku.
- `Width` – šířka elementu. Lze nastavit jako pevná hodnota, nebo jako procentuální vyjádření celkové možné šířky závislé na tom, kde se tag div nachází.
- `Height` – výška elementu.
- `Visibility` – nastavuje zda je element viditelný, či nikoli. Této vlastnosti jsem užil u levého menu,

kdy se v závislosti na aktuální pozici v katalogu zobrazují či skrývají položky menu (hodnoty visible a hidden).

V případě tohoto obchodu je jako základ celý dokument rozdělen do čtyř obdélníků. Hlavička obsahuje obrázek a základní údaje o košíku a osobě přihlášené do systému. Levá část obsahuje menu. Střed tvoří hlavní obsah a v pravé části se zobrazují doplňující informace. Tyto jednotlivé části pak uvnitř obsahují další elementy. Tento koncept je nejběžnější užívaný. Zajistí přehlednost a uživatelé internetu jsou na něj zvyklí.

## 5.7 Normy

Existuje několik používaných norem pro jazyk html. Některé jsou již minulostí, a některé jsou naopak základem moderního webu. Trend vývoje norem je striktně oddělovat formu od obsahu. V html dokumentu by tedy měl být pouze obsah a parametry jeho vzhledu by se měly ukládat do jiného souboru.

- HTML 2.0 – první definice dokumentu definující HTML jako aplikaci SGML. Dnes téměř nepoužívaný.
- HTML 3.0 – rozšířil předchozí normu o tabulky. Dnes se již neuvžívá.
- HTML 3.2 – maximálně kompatibilní norma. Zahrnuje základní prezentační prvky zavedené Netscapem a převzaté ostatními prohlížeči. Tato norma má stále své užití.
- HTML 4.0 a 4.1 – tato norma je rozdělena do tří sekcí:
  - Striktní – zahrnuje jen strukturální prvky.
  - Přejídný (transitional) – kvůli zpětné kompatibilitě kódu obsahuje i prvky z HTML 3.2.
  - Frameset – samostatný typ pro dokumenty, které používají rámce.

Pro tento projekt jsem zvolil normu HTML 4.1 transitional, která dovoluje větší toleranci pro psaní značek.



# 6 Testování a optimalizace webu

Testováním máme na mysli testování html kódu a použitelnosti, nikoli testování php programu. Rozebírat testování samotné aplikace by nemělo smysl, je příliš rozsáhlá a byla otestována a odladěna v průběhu vývoje jednotlivých částí.

Testovací veličiny lze rozdělit na dvě základní veličiny. Jsou jimi objektivní faktory. Jsou jimi ty, jenž lze exaktně změřit, a takto získané výsledky vyhodnotit. Druhou skupinou jsou subjektivní faktory. Přestože nejsou přesně změřitelné, tak jsou neméně podstatné jako skupina předešlá. Faktory se také liší tím, kdy je potřeba je testovat. Buď je testujeme a případné chyby odstraňujeme již v průběhu tvorby stránky (validita, hustota klíčových slov, informační architektura), nebo je postačí testovat až při vytvoření stránky (funkčnost odkazů). Všechny výše zmíněné vlastnosti bych doporučil testovat v průběhu tvorby jednotlivých stránek, protože procházet dodatečně všechny stránky systému a jednu po druhé testovat je dosti neefektivní.

## 6.1 Validita

Jedná se o snadno testovatelnou veličinu. Existuje několik nástrojů na testování validity. Asi nejznámější je na stránkách <http://www.w3.org>. Na tyto stránky lze buďto soubor nahrát, nebo na stránku umístit odkaz. Existuje i několik offline validátorů, jedním z nich je Total validator. Tento je přístupný také jako doplněk Mozilly Firefox. Tento validátor lze použít buď jako online verzi nebo jako offline program. Nevýhodou offline verze je fakt, že lze testovat jen html stránky, nikoli php soubory. Dalším možným produktem je <http://www.htmlvalidator.com>. K testování validity byl použit plugin Mozilly Html validator. Tento používá software Tidy vytvořený společností W3C. Jeho stránka je <http://tidy.sourceforge.net/>. Obchod při otestování validity generuje několik varování, která jsou zapříčiněna prohlížečem, který do stránky přidává JavaScripty za tag html. Stránky jsou definovány jako HTML 4.01, což není ta nejpřísnější varianta. Přejít na striktní HTML nebo XHTML by si vyžádal úpravu kódu.

## 6.2 Funkčnost odkazů

Jedná se o objektivní vlastnost, tedy lze testovat za pomoci pomocného programu jako je Link Checker přístupný na adrese <http://validator.w3.org/checklink>. Tento validátor prochází html stránku a kontroluje veškeré odkazované elementy včetně obrázku a externích css souborů. Všechny odkazy po provedených testech byly aktivní. Funkčnost odkazů by se měla provádět hlavně u často měněných stránek, jako jsou weblogy.

## 6.3 Přístupnost

Je to také poměrně snadno testovatelná vlastnost. Možnosti kontroly této vlastnosti jsou tři. Stránky lze zkontrolovat ručně. Je to zdarma, ale poměrně časově náročná operace. Stránky lze zkontrolovat pomocí různých automatizovaných nástrojů, kterých je na internetu celá řada. A nakonec můžeme nechat kontrolu na odborníkovi. Tato metoda je nejspolehlivější, ale není zadarmo.

Nejprve je tedy třeba zjistit pro koho je web určen a poté podle toho určit jaké vlastnosti přístupnosti musí stránka splnit. Existuje několik standardů – WCAG 1.0 (Web Content Accessibility Guidelines 1.0), Section 508 (americký zákon o přístupnosti webových stránek financovaných z veřejných zdrojů), Manifest Dogma W4 (Worth World Wide Web) a Blind Friendly Web. Pro testování byla použita automatická kontrola pomocí online aplikací. Zde je několik z nich:

- <http://sovavsiiti.cz/kontrast/>. Testování kontrastu barev.
- <http://colorfilter.wickline.org/>. Tato stránka zobrazí vaše stránky tak jak je vidí uživatelé s různými poruchami zraku.
- <http://webxact2.watchfire.com/>. Přístupnost celkově. Program předpokládá všechny možné situace kdy není použit JavaScript, není dostupná myš, je vypnutá grafika, nejsou dostupné barvy, jsou různé velikosti okna atd. Vlastnosti jsou rozděleny do čtyř základních skupin a to obecná přístupnost, kvalita, přístupnost a soukromí.
- <http://www.cynthiasays.com/mynewtester/cynthia.exe>. Jeden z nejlepších online validátorů.
- <http://www.wave.webaim.org/wave/>. Validátor graficky zobrazí přímo na stránce případné chyby.

### Výsledek testu kontrastu barev

Rozdíl jasu 236.182

Rozdíl barvy 693

Dostatečný kontrast: ANO

Rozdíl jasu bílého pozadí a černého textu je 255 a rozdíl barvy pak 755.

### Výsledek testy celkové přístupnosti

Pro testování přístupnosti jsem použil aplikaci <http://www.wave.webaim.org/wave/> a <http://www.cynthiasays.com/mynewtester/cynthia.exe>. Obě vygenerují množství údajů. Popisovat je by zabralo mnoho stránek. Z výsledků vyplývá, že pro co možná nejpřístupnější stránky je třeba testovat průběžně a je třeba dodržet mnoho detailů.

### Příklad viditelnosti stránky se zrakovou vadou protanopia



Obrázek 3.



Obrázek 4.

Pravidla tvorby přístupného webu pro účely novely Zákona č. 365/2000 Sb., o informačních systémech veřejné správy [15]:

1. Každý netextový prvek nesoucí významové sdělení má svou textovou alternativu.
2. Informace sdělované prostřednictvím skriptů, objektů, appletů, kaskádových stylů, obrázků a jiných doplňků na straně uživatele jsou dostupné i bez kteréhokoli z těchto doplňků.
3. Informace sdělované barvou jsou dostupné i bez barevného rozlišení.
4. Barvy popředí a pozadí jsou dostatečně kontrastní. Na pozadí není vzorek, který snižuje čitelnost.
5. Předpisy určující velikost písma nepoužívají absolutní jednotky.
6. Předpisy určující typ písma obsahují obecnou rodinu písem.
7. Obsah WWW stránky se mění, jen když uživatel aktivuje nějaký prvek.
8. Webová stránka bez přímého příkazu uživatele nemanipuluje uživatelským prostředím.
9. Nová okna se otevírají jen v odůvodněných případech a uživatel je na to předem upozorněn.
10. Na webové stránce nic neblinká rychleji než jednou za sekundu.
11. Webová stránka nebrání uživateli posouvat obsahem rámců.
12. Obsah ani kód webové stránky nepředpokládá ani nevyžaduje konkrétní způsob použití ani konkrétní výstupní či ovládací zařízení.
13. Webové stránky sdělují informace jednoduchým jazykem a srozumitelnou formou.
14. Úvodní webová stránka jasně popisuje smysl a účel webu. Název webu či jeho provozovatele je zřetelný.
15. Webová stránka i jednotlivé prvky textového obsahu uvádějí své hlavní sdělení na svém začátku.
16. Rozsáhlé obsahové bloky jsou rozděleny do menších, výstižně nadepsaných celků.
17. Informace zveřejňované na základě zákona jsou dostupné jako textový obsah webové stránky.
18. Na samostatné webové stránce je uveden kontakt na technického správce a prohlášení jasně

vymezujiící míru přístupnosti webu a jeho částí. Na tuto webovou stránku odkazuje každá stránka webu.

19. Každá webová stránka má smysluplný název, vystihující její obsah.
20. Navigační a obsahové informace jsou na webové stránce zřetelně odděleny.
21. Navigace je srozumitelná a je konzistentní na všech webových stránkách.
22. Každá webová stránka (kromě úvodní webové stránky) obsahuje odkaz na vyšší úroveň v hierarchii webu a odkaz na úvodní WWW stránku.
23. Všechny webové stránky rozsáhlejšího webu obsahují odkaz na přehlednou mapu webu.
24. Obsah ani kód webové stránky nepředpokládá, že uživatel již navštívil jinou stránku.
25. Každý formulářový prvek má přiřazen výstižný nadpis.
26. Každý rám má vhodné jméno či popis vyjadřující jeho smysl a funkčnost.
27. Označení každého odkazu výstižně popisuje jeho cíl i bez okolního kontextu.
28. Stejně označené odkazy mají stejný cíl.
29. Odkazy jsou odlišeny od ostatního textu, a to nikoli pouze barvou.
30. Obrázková mapa na straně serveru je použita jen v případě, že nebylo možné pomocí dostupného geometrického tvaru definovat oblasti v obrázkové mapě. V ostatních případech je použita obrázková mapa na straně uživatele. Obrázková mapa na straně serveru je vždy doprovázena alternativními textovými odkazy.
31. Uživatel je předem jasně upozorněn, když odkaz vede na obsah jiného typu, než je webová stránka. Takový odkaz je doplněn sdělením o typu a velikosti cílového souboru.
32. Kód webových stránek odpovídá nějaké zveřejněné finální specifikaci jazyka HTML či XHTML. Neobsahuje syntaktické chyby, které je správce webových stránek schopen odstranit.
33. V meta značkách je uvedena použitá znaková sada dokumentu.
34. Prvky tvořící nadpisy a seznamy jsou korektně vyznačeny ve zdrojovém kódu. Prvky, které netvoří nadpisy či seznamy, naopak ve zdrojovém kódu takto vyznačeny nejsou.
35. Pro popis vzhledu webové stránky jsou upřednostněny stylové předpisy.
36. Je-li tabulka použita pro rozvržení obsahu webové stránky, neobsahuje záhlaví řádků ani sloupců. Všechny tabulky zobrazující tabulková data naopak záhlaví řádků a/nebo sloupců obsahují.
37. Všechny tabulky dávají smysl čtené po řádcích zleva doprava. Tento projekt všechny tyto požadavky nespĺňuje. Jejich splnění je záležitostí dalšího vývoje a optimalizace.

## 6.4 Datová velikost

Datová velikost souvisí s dobou načítání stránky. Do velikosti dat se započítává veškerý obsah

zobrazený na stránce. Doba načítání stránky by neměla překročit 15 vteřin. Je nutné stanovit pro koho je web určen. Internetový obchod by měl sloužit pohodlně i uživatelům s pomalým připojením. Na stránce by se měl nejprve zobrazit její textový obsah, a až poté obrázky. Tomuto napomáhá rozdělení struktury do značek div. Pro analýzu této vlastnosti lze použít:

- Web page analyzer (<http://www.websiteoptimization.com/services/analyze/>)
- Automatický audit stránky (<http://www.dobryweb.cz/nastroje/automaticky-audit.html>)
- GetContentSize (<http://www.holovaty.com/tools/getcontentsize/>). Tento produkt analyzuje podíl textového obsahu na celkové velikosti. Pro test jsem užil nástroje na adrese <http://www.websiteoptimization.com/services/analyze/>. Testoval jsem úvodní stránku.

### **Výsledky testu**

- Celková velikost stažených dat je 17712 B, z toho 2,2 kB připadá na html, 9,3 kB na obrázky a 6,1 kB na CSS.
- Doba načítání stránky pro modem 56K je 4,53s.
- Výsledek testu bylo doporučení na snížení počtu grafických komponent, objektů a externích CSS souborů.
- Celková velikost stažených dat 17 kB je dostatečně malá.

Test je třeba brát jako orientační. Za provozu bude úvodní stránka vypadat jinak, bude obsahovat více informací.

## **6.5 Použitelnost**

Definice: použitelnost webových stránek určuje míru a efektivitu úsilí, nutného k nalezení a získání potřebných informací. Určuje jak snadno se na stránkách uživatelé orientují, jak rychle pochopí jejich uspořádání a ovládání a jaký uživatelský zážitek si z nich odnesou.

Více někdy znamená méně. Weby mohou uživatele zaujmout neobvyklou grafikou, animacemi či rozvržením. Může se stát, že je toho přespříliš, uživatel by raději dal přednost jednoduché stránce, na které by se lehce orientoval. Použitelnost je velice rozsáhlá vlastnost. Stránka je dobře použitelná pokud se uživatel rychle a intuitivně dostane tam, kam chce. Základem dobře použitelných stránek je kvalitní informační architektura a propracovaná navigace. Použitelností se zabývají specializované firmy a je to poměrně finančně nákladná záležitost. U komerčních projektů se tedy jedná o nutnost, která se vyplácí. U projektů, které si na sebe nevydělají je nejlepší vzít laika, stránky mu ukázat a ptát se jej na dobře cílené otázky. Tím nejlépe zjistíme co jsme udělali špatně a co je třeba zlepšit. Stejný postup jsem zvolil i zde. Test odhalil mnoho, pro tvůrce, na první pohled banálních nedostatků. Čas jistě ještě mnohé nedostatky odhalí.

## 6.6 Optimalizace pro vyhledávače

Anglická zkratka je SEO (Search engine optimization). Jedná se rovněž o velice rozsáhlou oblast.

### Testovatelné faktory

- Hustota klíčových slov. Tato veličina je přesně měřitelná. Hodnoty pro nejvýznamnější slova by se měla pohybovat mezi 3 až 7 procenty. Zde musíme stanovit kompromis mezi hustotou slov a rozumě čitelným obsahem. Hustotu lze měřit pomocí aplikace Keyword density analyzer (<http://www.ranks.nl/tools/spider.html>).
- Meta informace. Tato vlastnost představuje pohled na stránku očima robota, který ji prochází. Je to vlastnost více subjektivní, přesto lze měřit například programem Meta Tag Analyzer (<http://www.widexl.com/remote/search-engines/metatag-analyzer.html>), nebo Poodle Predictor (<http://www.gritechnologies.com/tools/spider.go>).
- Pozice ve vyhledávačích. Vlastnost lze měřit ručně tím, že zadáme klíčové slovo do vyhledávače a spočítáme pozici na která se stránka zobrazí. Tuto zdouhavou práci ulehčí například Search Engine Position Checker (<http://www.webmaster-toolkit.com/search-engine-position-checker.shtml>), nebo skript Google Position (<http://www.vitdlouhy.cz/gp/>). Tato vlastnost lze změřit až po určité době provozu. Například vyhledávač Google prochází celý internet, a provádí indexování, každé čtyři týdny.

### Výsledek testu hustoty slov

K testování byla užitá aplikace na stránce <http://www.ranks.nl/tools/spider.html>. Testována byla úvodní stránka. Ta obsahuje celkově 111 slov z nichž je 65 unikátních. 44 slov se nachází v nadpisech, 12 slov je v alt značkách, 59 slov je v odkazech, 5 slov je psáno tlustě a 5 slov je psáno kurzívou. Požadovanou hustotu splnilo pět slov, adidas (9.01 %), xc6 (3.60 %), všechny (2.70 %), helma (2.70 %), akce (2.70 %). Stránka testuje rovněž fráze ze dvou, tří a čtyř slov. Tyto údaje jsou pouze pro ukázkou. Texty na stránce jsou zatím jen ve zkušební fázi. Názvy produktů jsou pouze fiktivní, texty taktéž.

### Výsledek testu meta tagů

K testování byla užitá stránka <http://www.widexl.com/remote/search-engines/metatag-analyzer.html>. Tato stránka zjistila, že titulek je v pořádku, description tag odpovídá obsahu stránky na 60%, keywords odpovídají obsahu stránky na 20%, robots a author jsou rovněž bez chyb. Aplikace kontroluje rovněž klíčová slova. Nejlépe se umístilo slovo adidas (hustota 10.99%). Ze všech

provedených testů vyplývá že stránka, co se optimalizace týče, není příliš kvalitně zpracována a pravděpodobně se, bez dalších úprav, nebude vyskytovat na prvních pozicích ve vyhledávačích.

#### **Další faktory ovlivňující výsledky vyhledávačů**

- Odkazy z internetu. Čím více webů na naši stránku odkazuje tím výše bude postavena ve výsledcích vyhledávačů. Docílit toho lze několika způsoby. Naše stránka může být natolik dobrá, že ostatní na ni budou sami bez vyzvání odkazovat. Můžeme často navštěvovanému webu zaplatit, aby na naši stránku umístil odkaz. U stránek s podobnou tématikou můžeme odkazy vyměnit.
- Sémantická bohatost. U speciálních elementů je třeba zdůraznit jejich příslušnost a to tak, že je vložíme do příslušného tagu, buďto nadpisu, seznamu, či tabulky.
- Klíčová slova. Vyhledávače z obsahu webu zjistí o čem vlastně je. Určují tak hlavně z nadpisů, URL adres a titulků stránek. Proto je důležité promyšleně tyto elementy zadávat. Existuje možnost klíčová slova zadávat přímo do hlavičky stránky. Mnohé vyhledávače (např. Google) tato slova neberou v potaz, a tak moderní weby tato slova již neuvádějí.

# 7 Propagace webu

Pro obchod jako takový, ale i pro ostatní komerční stránky, jsou zásadní dvě věci. Zákazník se o něm musí dozvědět a musí jej navštěvovat pravidelně. Jedině tehdy bude mít obchod úspěch. K tomuto je nezbytná propagace. Možností propagace je několik:

- Zavedení do internetového vyhledávače.
- Bannerová reklama.
- Zajistit, aby internetová média působící ve stejném oboru o obchodu psala a odkazovala na něj prostřednictvím odkazu.
- Aliance a sponzorování soutěží.
- Uvedení adresy webu v podpisu u emailu zaměstnanců, v našem případě administrátorů.
- Výměnná reklama na spřízněných webech.
- Reklama mimo internet. Všude, kde se vyskytne informace o naší firmě měla by tam být i adresa na internetovou prezentaci.

Tyto prvky budou uplatněny jakmile aplikace bude mít reálné použití.

## 7.1 Zavedení do vyhledávačů a katalogů

Katalog na internetu lze přirovnat ke Zlatým stránkám. Stránky jsou v něm uvedeny v abecedním pořadí, nebo jsou tříděny podle kategorií. Příkladem jsou weby [www.seznam.cz](http://www.seznam.cz), [www.atlas.cz](http://www.atlas.cz), [www.zmije.cz](http://www.zmije.cz), nebo [www.yahoo.com](http://www.yahoo.com). Stránky v katalogu jsou přidávány ručně přes odkaz na stránce příslušného katalogu. Vyhledávač provádí fulltextovou indexaci stránek. Projde stránky a do své databáze uloží všechna relevantní slova, tj. ne předložky částice atd. U slova je odkaz, kde bylo nalezeno. Tato služba považuje za důležité:

- Slova v tagu META, kde se přímo zadávají klíčová slova.
- Tag TITLE, tedy titulek stránky zobrazující se v liště prohlížeče.
- Jednotlivé nadpisy H1..H6.
- Text titulní stránky. Ostatním stránkám je přikládána nižší priorita.



## 8 Srovnání s existujícími obchody

Pro srovnávání jsem volil pouze obchody s podobným zaměřením, tedy obchody se sportovním vybavením.

### **www.d-structure.com**

Stránka která nezobrazuje obchod jako svou prioritu. Prodává pouze čtyři druhy zboží. Jedná se především o centrum, kde se scházejí a komunikují osoby z dané komunity. Přednější jsou zde aktuality, fóra, diskuze, magazín. Přesto je stránka ve Spojených státech, ale i ve světě, velice populární.

### **www.blue-tomato.at**

Velice oblíbená rakouská stránka. Vedle obchodu, je zde spousta dalších rubrik jako magazín, či poradna. Nabízenými funkcemi je tento obchod srovnatelný se zde popisovaným projektem. Svou oblíbenost si zajistil mimo jiné i sponzorováním lyžařských středisek a různých akcí a závodů.

### **www.sportobchod.cz**

Tento obchod není nijak zajímavý. Jediná věc, která mne zaujala je propracované hodnocení produktů.

### **www.sportovniobchod.cz**

Jedná se o jednodušší obchod, se základními funkcemi. Obchod má podobnou navigaci. Produkty je také možno komentovat.

### **www.ski-eschop.cz**

Funkčně velice podobný obchod. Zákazníci mají vlastní účet. Produkty lze komentovat. Podobně se rozbaluje levé hlavní menu. Obchod obsahuje i poradnu.

### **www.sportovni-potreby.cz**

Vzhledově poměrně nepřehledný obchod. Nabízí široký sortiment.

### **www.shop-sport.cz**

Opět obchod s bohatou nabídkou. Obchod má rovněž systém aktualit. Obchod není vzhledově příliš zdařilý a používá v adrese stránku php s mnoha parametry, což pro vyhledávače není optimální.

# 9 Návrh rozšíření

## 9.1 Obecné

- Pokud bychom od obchodu vyžadovali umístění na předních příčkách ve vyhledávačích, pak by bylo dobré převést html kód do normy 4.1 strict. Tím dojde k dokonalému odstranění obsahu od značek udávající vzhled stránky. S tím souvisí optimalizace kódu a optimalizace na určitá slova, což je dlouhodobá záležitost, kterou se zabývají specializované firmy.
- Pro úspěch v široké konkurenci by bylo výhodné více propracovat design a celkové rozvržení objektů na stránce. Současný vzhled je víceméně pracovní. Rozvržení by bylo možná vhodné přepracovat tak, aby se stránka roztahovala přes celou šířku obrazovky a reagovala tak na rozlišení uživatele.

## 9.2 Část administrátora

- Aktuality jsou zadávány jako krátký text, s možností přidat jeden obrázek. Do budoucna by bylo dobré aktuality zadávat jako celý článek s možností vkládat fotky do textu. Na titulní straně by se pak zobrazila pouze jakási upoutávka na článek. Výsledek by pak vypadal jako stávající stav s tím, že by u každé aktuality byl odkaz na celý článek.
- Současný popis produktu je zadáván jako text. Mnoho stránek však popis zobrazuje jako tabulku parametrů. Bylo by tedy možno popis vkládat jako dvojice název parametru a jeho hodnota. Těchto dvojic by pak bylo možno zadat neomezeně mnoho. Do databáze by se pak ukládaly do jedné buňky tabulky a byly by vzájemně odděleny speciálním znakem, aby je bylo možno rozdělit do tabulkového výpisu.

## 9.3 Část uživatele

- V mnoha systémech je zabudována funkce zasílání novinek na email. Tyto zprávy by se zasílaly registrovaným uživatelům. Uživatel by měl možnost ve svém účtu nastavit, zda chce novinky zasílat, či nikoli.
- Heslo a email uživatele by se mohl uložit do cookies a tím pádem by je nemusel zákazník při každém přihlášení vyplňovat.
- Pod produktem by se mohly zobrazovat související produkty.
- U kategorie by se mohly zobrazit související kategorie.
- Samozřejmostí je přidání obchodních podmínek, podmínek dopravy, podmínky reklamace, kontakty, atd.

- Velice užitečnou vlastností u mnoha obchodů je možnost vybrat několik produktů a ty na tento účel uzpůsobené stránce porovnat. Porovnávat by bylo možno jen produkty ve společné hlavní kategorii a předpokladem by byly parametry zadané odděleně a jednotně.
- Při nákupu, které provedou firmy by se mělo požadovat identifikační číslo a fakturační adresa.
- Ne příliš užívanou funkcí internetových obchodů je export či import dat v xml formátu. Tato funkce se nabízí jako jedno z dalších rozšíření ať už pro výstup a vstup dat, tak pro sdílení dat s jiným subjektem.

# 10 Závěr

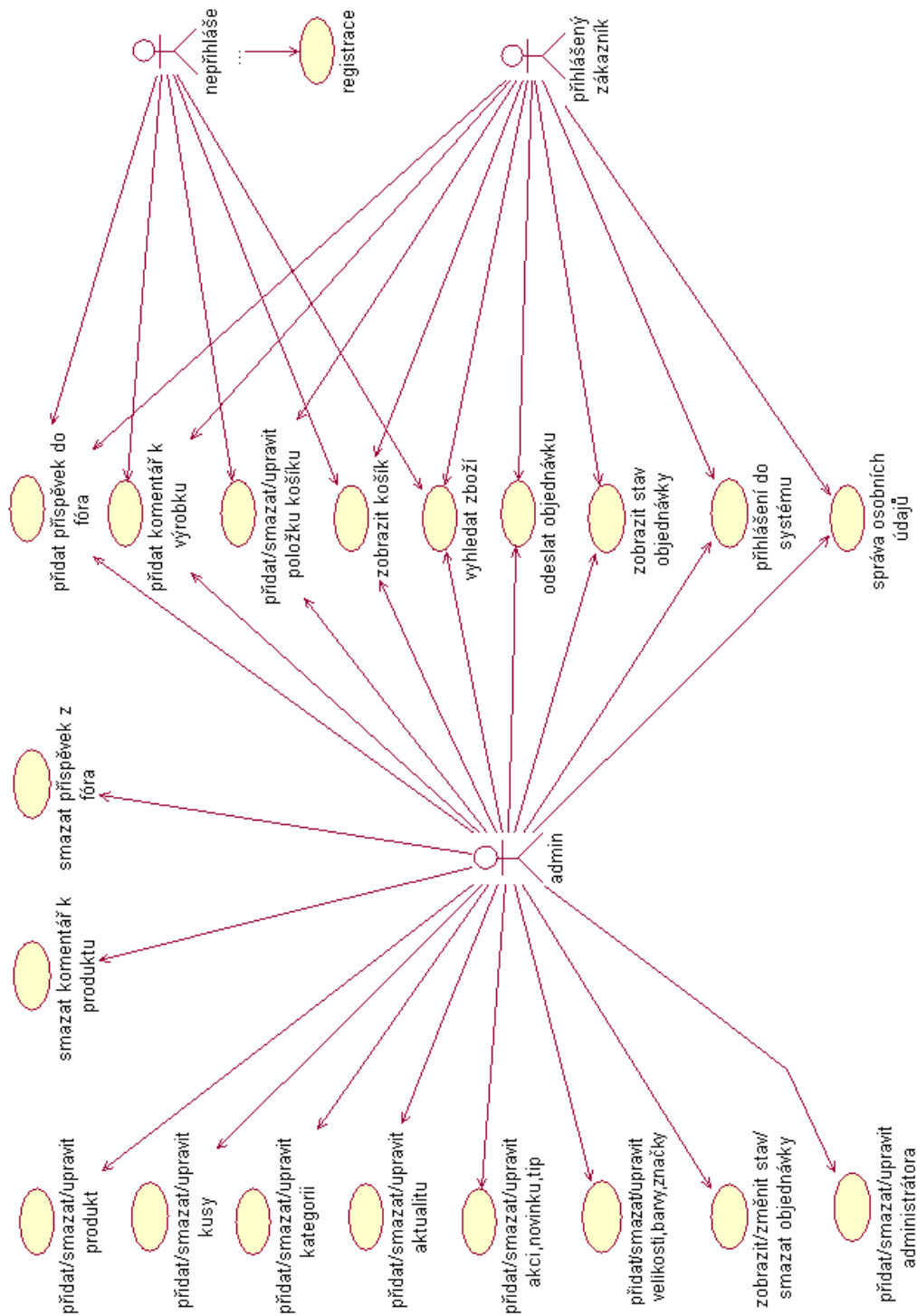
Internetový obchod umožňuje pohodlné nákupy v teple domova. Prostřednictvím internetu se dostane široký okruh zákazníků k nabízenému zboží. Internetový obchod nikdy nezavírá, nabízí přijatelnější ceny. Internetový prodej dokáže pružněji reagovat na požadavky zákazníka, obchodník nemusí mít všechno zboží skladem. V oblasti internetového prodeje je silná konkurence v důsledku snížení vlivu vzdáleností na přímý prodej. Trh v České republice podléhá světovému trendu globalizace. Na internetu čelí přímé konkurenci obchody, které mohou ležet na opačných koncích republiky. Do boje o zákazníka vstupuje více než fyzická poloha obchodu, technické provedení jeho internetových stránek. Programátor čelí vysokým požadavkům. Malé zaváhání při tvorbě aplikace může výrazně poškodit obchodníka. Zákazník je po špatné zkušenost nenávratně ztracen.. Pokud má jakýkoli obchod uspět pak nestačí jej pouze vytvořit. Je potřeba jej neustále aktualizovat a upravovat, tak aby byl atraktivní a oblíbený. Tyto podmínky není snadné splnit.

Internetový obchod, který jsem vytvořil je plně funkční a splňuje požadavky, které jsou od tohoto druhu internetové aplikace očekávány. Existuje mnoho obchodů, které nabízí více funkcí, jsou pro administrátora i uživatele zajímavější a existují obchody, které jsou více statické a nelze je tak snadno aktualizovat. Z čehož vyplývá zařazení tohoto projektu mezi střední třídu internetových obchodů. Po provedení několika testů bych jako hlavní nedostatek viděl obsah webu. Není optimalizovaný. Pro další vývoj bych doporučil přísnější normu a důslednější značení elementů stránek. V samotné funkčnosti lze jistě nalézt chyby, ty vyplynou z dlouhodobého používání aplikace a je třeba je odstranit postupně. I vhodná rozšíření, které se týkají nabízených funkcí vyplynou, až ze samotného reálného provozu.

# Literatura

- 1 Stig Sæther Bakken, Alexander Aulbach, Egon Schmid, Jim Winstead, Lars Torben Wilson, Rasmus Lerdorf, Andrei Zmievski, Jouni Ahto, Sestavil, Stig Sæther Bakken, Egon Schmid, *PHP Manuál*, vydáno 14-12-2001, copyright © 1997, 1998, 1999, 2000, 2001 PHP Documentation Group
- 1 Ing Pavel Přívětivý, *Seznámení s příkazy SQL*
- 2 TeX AB, Detron HB and MySQL Finland AB, *MySQL Reference Manual*, Copyright c 1997-2000
- 3 ©Netscape Communications Corporation All Rights Reserved, *JavaScript Reference*, December 12, 1997,
- 4 Danny Goodman, *JavaScript Bible GOLDEDITION*
- 5 Sian Si Alhir, *Learning UML*, Publisher O'Reilly, 2003
- 6 J. Zendulka, *Databázové systémy a návrh databází*, Konceptuální modelování
- 7 Dušan Jankovský, *Použití souboru .htaccess* [online], [cit. 10.1.2007]  
<<http://www.jakpsatweb.cz/server/htaccess.html>>
- 8 Lukáš Zapletal, *Regulární výrazy v příkladech* [online], 2003, [cit. 12.12.2006],  
<<http://www.root.cz/clanky/regularni-vyrazy-v-prikladech/>>
- 9 *Tvorba kvalitních webových stránek* [online], poslední revize 13.4.2005 [cit. 12.12.2006],  
<<http://tvorba-www-stranek.kvalitne.cz/>>
- 10 *PHP: Hypertext Preprocessor* [online], [cit. 12.12.2006], <<http://www.php.net>>
- 11 Larry Ullman, *PHP a MySQL*, Computer Press
- 12 Sdružení obrany spotřebitelů České republiky, *VÝSLEDKY TESTU INTERNETOVÝCH OBCHODŮ* [online], 2004 [cit. 10.1.2006],  
<<http://www.spotrebitele.info/priloha/testinternet2004.pdf>>
- 13 Kolektiv autorů společnosti Weinhold Andersen Legal a Andersen, *E-Obchod*
- 14 David Špinar, *Přístupnost: Web a weblog věnovaný přístupnosti webových stránek* [online], [cit. 20.12.2006], <<http://pristupnost.nawebu.cz/>>
- 15 Martin Labský, *Cvičení 13* [online], [cit. 10.12.2006], <<http://nb.vse.cz/~labsky/c13.html>>
- 16 Netscape Communications Corporation, *Client-Side JavaScript* [online], 1999 [cit. 5.1.2007],  
<<http://devedge-temp.mozilla.org/library/manuals/2000/javascript/1.3/guide/ClientGuideJS13.pdf>>
- 17 Ing. Willy WEIGLHOFER, *Jednoduchý návod pro Rational Roze 98* [online], [cit. 10.1.2006],  
<<http://www.fsfd.cvut.cz/cz/u210/pis/navod.doc>>
- 18 Skřivan Jaromír, *Databáze a jazyk SQL* [online], 4. 8. 2000 [cit. 23.1.2007],  
<<http://interval.cz/clanky/databaze-a-jazyk-sql/>>

# A USE CASE diagram



## A.1 Podrobný popis USE CASE modelu

USE CASE diagram viz. příloha A.

### Administrátor

- Přidat/smazat/upravit produkt – editace produktu.
- Přidat/smazat/upravit kusy – editace kusů konkrétního produktu
- Přidat/smazat/upravit kategorii – editace kategorií produktů
- Přidat/smazat/upravit akci, tip, novinku – přidání, či odebrání produktu z novinek, akcí a tipů.
- Přidat/smazat/upravit velikosti, barvy, značky – editace barev, velikostí kusů a značek produktů.
- Zobrazit/smazat/změnit stav objednávky – editace objednávek, u objednávek lze změnit pouze stav.
- Přidat/smazat/upravit administrátora – odpovídá změně osoby, s tím rozdílem, že administrátor může měnit i roli osoby v systému (to zda je osoba administrátor či pouhý zákazník).
- Smazat komentář k produktu – administrátor může mazat komentáře.
- Smazat příspěvek z fóra – administrátor může smazat příspěvek z fóra.

### Přihlášený uživatel a administrátor

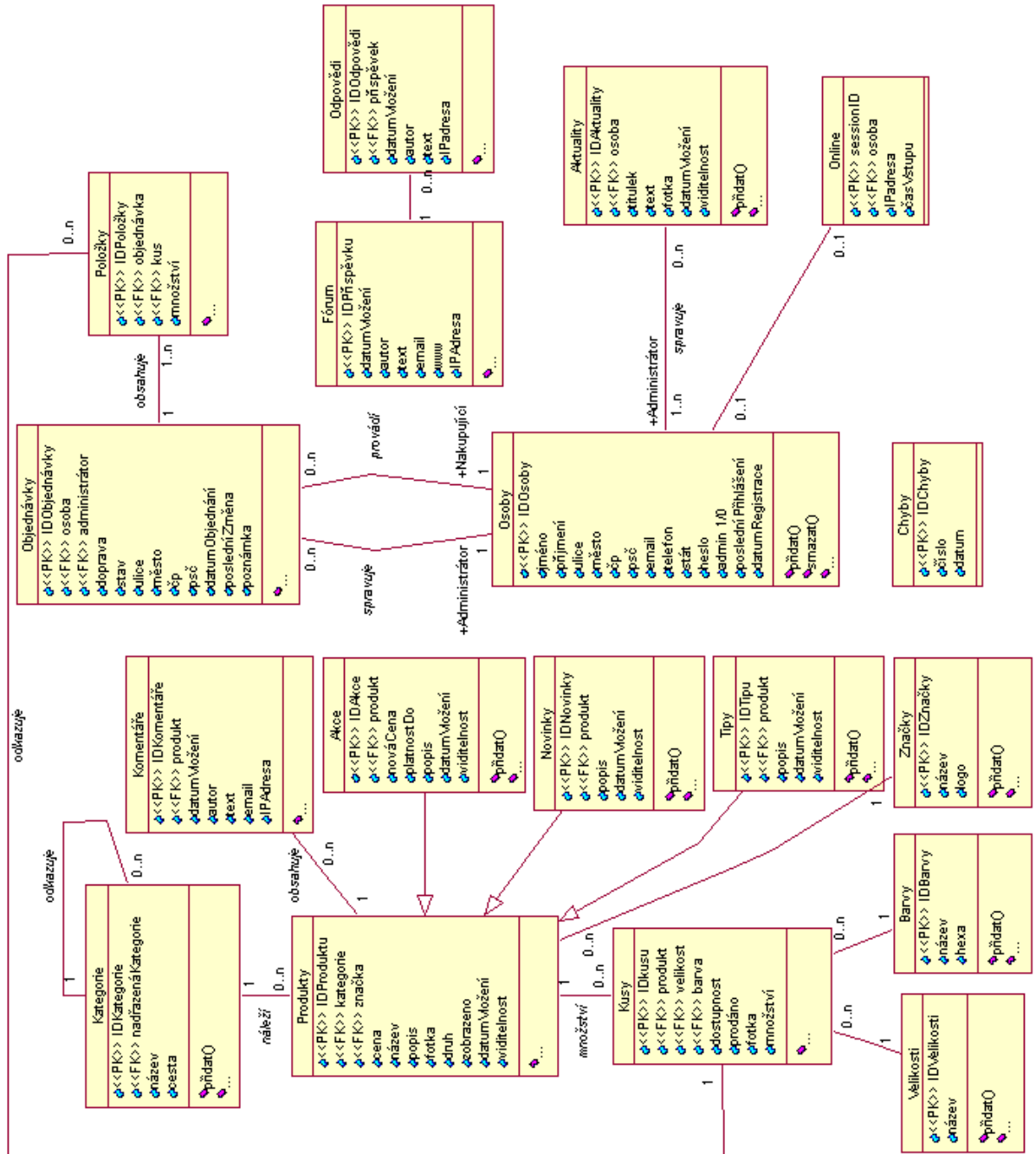
- Odeslat objednávku – akce znamená odeslání objednávky. I administrátor s tedy může objednat zboží. Nepřihlášený zákazník tak učinit nemůže.
- Zobrazit stav objednávky – zobrazení všech objednávek konkrétní osoby.
- Přihlášení do systému – vyplnění emailu a hesla a potvrzení.
- Správa osobních údajů – každá registrovaná osoba bude moci měnit své osobní údaje uložené v systému.

### Přihlášený uživatel, nepřihlášený uživatel a administrátora

Přidat příspěvek do fóra – příspěvek do fóra může přidat kdokoli.

- Přidat komentář k výrobku – okomentovat výrobek může kdokoli.
- Přidat/smazat/upravit položku košíku – přidávat a upravovat košík může i nepřihlášený uživatel.
- Zobrazit košík – obsah košíku bude kdykoli pro kohokoli přístupný.

# B ER diagram





## B.1 Podrobný popis ER diagramu

ER diagram viz. příloha B.

### Produkt - atributy

- IDProduktu– všechny produkty všech kategorií budou v jedné tabulce. Každý produkt bude mít unikátní identifikační číslo. Za produkt je považován model výrobku, nikoli konkrétní velikost či barva.
- Kategorie – každý výrobek náleží do kategorie. Položka kategorie je index, který odkazuje na název kategorie v tabulce kategorie. Kategorie je např. brusle, letní, atd.
- Značka – značka je index odkazující na název značky a logo v tabulce značky.
- Cena – cena produktu. Lze zadávat buďto bez dph nebo s. Ta druhá varianta bude moci být vypočtena a nebude tedy potřeba ji udržovat v systému. Základní hodnota je s dph, takto se i zobrazuje u zboží a v košíku.
- Název – bude sloužit mimo jiné jako titulek při zobrazení jak seznamu, tak i detailu produktu. Z názvu je nahrazením nepovolených znaků vytvořena adresa výrobku.
- Popis – detailní popis produktu zahrnující jak technické vlastnosti, tak ty ostatní. Při zadávání bude moci administrátor užít html tagů, aby bylo dosaženo přehlednosti.
- Fotka – hlavní fotka produktu. Bude zobrazena u popisu produktu. Jako fotku kusu je možno zadat další fotky.
- Druh – druh je index označující zda-li jde o pánskou, dámskou, či dětskou verzi produktu. Popis jednotlivých druhů, je definován napevno v konfiguračním souboru.
- Zobrazeno – číslo značící kolikrát byl produkt zobrazen. Počítá se jedna návštěva na session prohlížeče.
- Datum vložení – datum kdy byl produkt vložen do systému. Mění se i při úpravě produktu.
- Viditelnost – číslo 0 značí, že produkt nebude pro uživatele viditelný, 1 značí, že produkt bude normálně zobrazen.

### Produkt – operace

- Přidat – administrátor bude produkty přidávat. Přidat lze jen do kategorie, která nemá podkategorii.
- Smazat – smazání produktu. Smaže se i jeho fotka. Nelze smazat pokud jsou definovány nějaké kusy.
- Upravit – úprava informací o produktu uložených v databázi. Upravit lze i fotku.

### Kategorie – atributy

- IDKategorie – primární klíč tabulky. U produktu je to cizí klíč odkazující na kategorii.
- Nadřazená kategorie – cizí klíč odkazující na id nadřazené kategorie. Každá kategorie může mít několik podřazených a jednu nadřazenou kategorii.
- Název – název kategorie. Název musí být v tabulce kategorie jedinečný.
- Cesta – je vytvořena z názvu produktu nahrazením interpunkce a českých znaků podtržítkem. Cesta slouží pro generování adresy produktu.

### **Kategorie – operace**

- Přidat – přidá kategorii, nadřazená kategorie se volí ze všech kromě sebe sama.
- Smazat – smaže kategorii, pouze pokud nemá podřazené kategorie.
- Upravit – upravuje kategorii.

### **Kusy – atributy**

- IDKusu – primární klíč tabulky. Užije se při identifikaci zboží v košíku.
- Produkt – cizí klíč. Odkazuje na produkt, který vlastní daný kus.
- Velikost – cizí klíč. Odkazuje na velikost v tabulce velikostí.
- Barva – cizí klíč. Odkazuje na barvu v tabulce barvy.
- Dostupnost – číslo, odkazuje na název dostupnosti, definované napevno v souboru konfigurace.
- Prodáno – značí kolik kusů daného produktu se prodalo. Údaj je měněn ručně po odeslání daného kusu.
- Fotka – fotka kusu. Pokud je množství kusu 0, pak se zobrazí jen fotka. Lze užít jako fotogalerii produktu.
- Množství – počet kusů produktu. Lze měnit ručně pokud si někdo kus objedná.

### **Kusy – operace**

- Přidat – přidá kus k produktu.
- Smazat – maže kus.
- Upravit – úprava kusu.

### **Velikosti – atributy**

- IDVelikosti – primární klíč tabulky.
- Název – Název velikosti, např. L nebo 156.

### **Velikosti – operace**

- Přidat – přidá velikost.
- Smazat – maže velikost.
- Upravit – upravuje velikost.

#### **Barvy – atributy**

- IDBarvy – primární klíč tabulky.
- Název – název barvy, např. zelená, červeno hnědá.
- Hexa – hexadecimální zápis barvy. Může být použito jako náhrada fotky.

#### **Barvy – operace**

- Přidat – přidá barvu.
- Smazat – maže barvu.
- Upravit – upravuje barvu.

#### **Komentáře – atributy**

- ID komentáře – jednoznačný identifikátor.
- Produkt – cizí klíč, odkazuje na produkt, kterého se komentář týká.
- DatumVložení – datum vložení komentáře.
- Autor – autor komentáře.
- Text – text komentáře.
- Email – emailová adresa zadavatele.
- IPAdresa – IP adresa, ze které byl komentář odeslán.

#### **Komentáře – operace**

- Přidat – přidání komentáře k výrobku, může být provedeno kýmkoli
- Smazat – smazat komentář může jen administrátor.

#### **Akce – atributy**

- IDAkce – primární klíč tabulky.
- Produkt – cizí klíč, odkazuje na produkt, který je v akci.
- NováCena – cena produktu v akci, musí být vždy nižší než původní cena.
- PlatnostDo – platnost akce, po době platnosti se akce nebude zobrazovat.
- Popis – popis akce.
- DatumVložení – datum vložení akce do systému, mění se i při úpravě akce.

- Viditelnost – akce může a nemusí být viditelná, po uplynutí doby platnosti se stává neviditelnou.

#### **Akce – operace**

- Přidat – přidání produktu do tabulky akce, tuto operaci provádí administrátor.
- Smazat – smazání produktu z tabulky akce.
- Upravit – úprava akce.

#### **Novinky – atributy**

- IDNovinky – primární klíč tabulky.
- Produkt – odkazuje na produkt, který je novinkou.
- Popis – popis novinky.
- DatumVložení – datum vložení novinky do systému.
- Viditelnost – viditelnost novinky, novinka je buďto viditelná nebo ne.

#### **Novinky - operace**

- Přidat – Novinky lze odvodit z data přidání do systému, ale ne vždy je tento způsob výhodný, například pokud je naráz přidáno mnoho zboží a administrátor chce vypíchnout jen něco. Proto je lepší zadat novinky ručně.
- Smazat – smazání novinky
- Upravit – úprava novinky

#### **Tipy – atributy**

- IDTipu – primární klíč.
- Produkt – id produktu, který je tipem.
- Popis – popis tipu.
- DatumVložení – datum vložení tipu do systému.
- Viditelnost – viditelnost tipu. Doba platnosti je neomezená.

#### **Tipy – operace**

- Přidat – přidání tipu, provádí administrátor.
- Smazat – smazání tipu.
- Upravit – úprava tipu, tedy popisu, viditelnosti a produktu.

#### **Značky – atributy**

- IDZnačky – jednoznačný identifikátor značky.

- Název – název značky, po nahrazení nevhodných znaků se název bude podílet na adrese výrobku.

### **Značky – operace**

- Přidat – přidání značky, provádí administrátor.
- Smazat – smazání značky, včetně fotky. Nelze smazat značku v níž je nějaký produkt definován.
- Upravit – úprava typu, nelze smazat značku v níž je nějaký produkt definován.

### **Objednávky – atributy**

- IDObjednávky – identifikátor objednávky. Jedna objednávka znamená jeden nákup. Objednávka má pak položky.
- Osoba – cizí klíč odkazující na id osoby v tabulce osoby.
- Administrátor – značí id osoby, která je administrátorem a provedla změnu objednávky. Při objednání je tento údaj 0.
- Doprava – odkazuje na typ dopravy definovaný napevno v konfiguračním souboru.
- Stav – stav objednávky značí v jakém stádiu se nachází tedy např. „odesláno“, „zpracováno“, „vyřízeno“, ...
- Ulice – součástí objednávky je i adresa pro případ, že uživatel chce zboží poslat jinam.
- Město – město.
- Psč – poštovní směrovací číslo.
- DatumObjednání – datum kdy zákazník objednal zboží.
- Poznámka – poznámka, kterou může zákazník zanechat při objednání.

### **Objednávky – operace**

- Přidat – objednání zboží zákazníkem.
- Smazat- administrátor může objednávku smazat. Uživatel ne.
- Upravit – znamená změnit stav objednávky, nic jiného měnit nelze.

### **Položky – atributy**

- IDPoložky – primární klíč. Identifikuje položku.
- Objednávka – cizí klíč, odkazuje na objednávku, ke které položka náleží.
- Kus – odkazuje na kus, který byl objednán.
- Množství – množství objednaného kusu.

### **Položky – operace**

- Přidat – položka je přidána jakmile je zboží objednáno.
- Smazat – položka je smazána jakmile je smazána objednávka.

### **Fórum – atributy**

- IDPříspěvku – primární klíč.
- DatumVložení – datum vložení příspěvku do databáze.
- Autor – autor příspěvku.
- Text – samotný příspěvek.
- Email – email zadavatele.
- Www – www adresa zadavatele.
- IPAdresa – id adresa zadavatele, lze užít pro antispam atd.

### **Fórum – operace**

- Přidat – přidá příspěvek do fóra.
- Smazat – administrátor bude mít možnost smazat příspěvky z fóra.

### **Odpovědi – atributy**

- IDodpovědi – jednoznačný identifikátor odpovědi.
- Příspěvek – odkazuje na příspěvek, ke kterému odpověď náleží.
- DatumVložení – datum vložení do databáze.
- Autor – autor příspěvku.
- Text – text příspěvku.
- IPAdresa – IP adresa ze které byl příspěvek vložen.

### **Odpovědi – operace**

- Přidat – přidat odpověď na příspěvek.
- Smazat – v případě smazání příspěvku z fóra budou smazány odpovídající odpovědi.

### **Osoby - atributy**

- IDOsoby – identifikátor zákazníka.
- Jméno – jméno zákazníka.
- Příjmení – příjmení.
- Město – město.

- Čp – číslo popisné.
- Email – email osoby.
- Telefon – telefon, osoby.
- Stát – tři znaky reprezentující stát. Pole se státy je definováno v konfiguračním souboru.
- Heslo – heslo používá zákazník pro přihlášení, bude v databázi uloženo zakódované jednocestným kódem (funkce *md5()*), nebude jej tedy možné rozluštit.
- Admin – hodnota je 1 nebo 0 a značí zda-li je osoba administrátor, či ne.
- Poslední přihlášení – datum a čas posledního přihlášení do systému.
- Datum registrace – datum registrace.

### **Osoba – operace**

- Přidat – nová registrace, provádí buď zákazník nebo administrátor.
- Smazat – osobu může smazat pouze administrátor.
- Upravit – zákazník upravuje svoje osobní údaje, administrátor všechny.
- Přihlásit – přihlášení do systému.
- Odhlásit – odhlášení ze systému.

### **Chyby – atributy**

- IDChyby – primární klíč tabulky chyby.
- Číslo – každá chyba má jednoznačné číslo, které ji identifikuje.
- Datum – datum kdy byla chyba způsobena.

### **Aktuality - atributy**

- IDAktuality – identifikátor aktuality.
- Osoba – id administrátora, které aktualitu vložil.
- Titulek – nadpis aktuality.
- Text – text aktuality.
- Fotka – fotka která s aktualitou souvisí, není povinným parametrem.
- DatumVložení – datum vložení aktuality do systému.
- Viditelnost- viditelnost aktuality.

### **Aktuality - operace**

- Přidat – přidání aktuality, tyto operace provádí administrátor.

- Upravit – úprava aktuality.
- Smazat – smazání aktuality.

#### **Online – atributy**

- SessionID – id aktuální session, se kterou uživatel vstoupil na stránky obchodu. Slouží jako primární klíč tabulky.
- Osoba – id osoby pokud je osoba zároveň přihlášena jinak 0.
- IPAdresa – ip adresa počítače uživatele.
- Čas vstupu – čas v sekundách od referenčního roku, užije se pro automatické odhlášení uživatele po určité době nečinnosti.



# C Uživatelská příručka

## C.1 Instalace

Instalace je poměrně jednoduchá. Je třeba provést několik kroků.

1. Nahrát všechny soubory do adresáře www nebo do adresáře domény třetího řádu, na serveru kde chceme obchod provozovat.
2. Přesunout soubor s přístupovým heslem a názvem databáze (shop\_dbconnect.php) z adresáře kde jsou všechny soubory a adresáře aplikace do kořenového adresáře webu, tedy tam, kde se na něm nikdo mimo administrátora nedostane. Důležité je, aby se soubor nevyskytoval v adresáři aplikace.
3. Dále je potřeba vytvořit databázi a její název a přístupové heslo vložit do výše zmíněného souboru shop\_dbconnect.php. Položka \$pocitac má hodnotu 'mysql' nebo 'localhost' v závislosti na tom kam aplikaci instalujeme. Položka \$uzivatel je pevně dána poskytovatelem databázové služby.
4. Instalace bude potřebovat ftp připojení, aby mohla nastavit přístupová práva adresářům, do kterých se bude zapisovat. Údaje pro ftp připojení budou rovněž v kořenovém adresáři serveru v souboru shop\_ftp\_dbconnect.php. Tento soubor je rovněž potřeba přesunout do kořenového adresáře webu.
5. Nyní spustíme skript uložený na adrese /instalace/. Na hlavní stránce skriptu jsou dvě možnosti. Buďto můžeme vytvořit tabulky a vložit příslušné hodnoty, nebo můžeme tabulky odstranit. Heslo a email pod kterým se bude přihlašovat první administrátor jsou v souboru shop\_dbconnect.php. Tuto hodnotu můžeme před samotným vytvářením tabulek změnit. Základní hodnota email/heslo je default/default.
6. První tlačítko spustí vytváření tabulek. Připojí k databázi a vytvoří potřebné tabulky. Do vytvořených tabulek vloží počáteční hodnoty (jedna aktualita, produkt, kus, značka, barva, velikost, jeden administrátor, akce, novinka, tip, kategorie). Skript se dále pokusí připojit přes ftp k serveru a nastavit práva zápisu adresářům, do kterých se bude zapisovat. Pokud se to nepovede, zobrazí upozornění, že uživatel bude tato práva muset nastavit ručně aplikací poskytovanou poskytovatelem serveru, nejčastěji WebFtp.
7. Druhá možnost zruší všechny tabulky z databáze. Tuto možnost použijeme pokud chceme obchod odinstalovat, nebo přeinstalovat.
8. Nyní je potřeba v souboru /top.php na začátku nastavit domovský adresář webu. Jedná se o proměnnou \$HOME\_DIR. Její obvyklý formát je podobný '/DISK2/WWW/obchod.cz/'.
9. V další fázi uživatel provádí změnu potřebných nastavení v souboru /config.php. Všechny parametry jsou uvnitř popsány.

## C.2 Obecné prvky

### Levé menu

Levé menu slouží pro základní navigaci v obchodu. Jeho prvky tvoří názvy kategorií. Stupně menu jsou od sebe graficky odlišeny. Podřazené kategorie se zobrazují odsazeny a v jiné barvě tak, aby bylo jasné, která kategorie náleží pod kterou. V menu se rozbalí pouze jedna hlavní kategorie a pouze aktuální stupeň zanoření. Zamořená menu není nijak omezeno, přesto by nemělo být větší než 3-4 stupně. Při zanoření vyšším by se menu nemuselo vměstnat na přiřazenou šířku.

### Horní lišta

Horní lištou je myšlen řádek v horní části stránky. Tento řádek v levé části obsahuje informaci o aktuálním stavu košíku a odkaz na stránku, kde se zobrazí jeho detailní obsah. Košík může být buďto prázdný, nebo se zobrazí počet položek a celková cena s DPH.

V pravé části se zobrazuje informace o uživateli. Uživatel je buďto nepřihlášen, pak se zobrazí odkaz na stránku s přihlašovacím formulářem. Pokud je uživatel přihlášen pak se zobrazí jeho jméno a příjmení a odkaz, který pro kliknutí uživatele odhlásí.

### Pravá lišta

Pravá část stránky zobrazuje doplňující informace. Nejprve jsou to tipy. Tipy jsou produkty, na které chce administrátor upozornit. Tip je definován názvem produktu a jeho popisem. Název produktu odkazuje na detail produktu.

Pod tipy se zobrazuje několik nejprodávanějších produktů. Jejich počet lze změnit v konfiguračním souboru. Tyto produkty jsou zobrazeny jako názvy dotyčných souborů s odkazy na jejich detail.

### Úvodní stránka

V horní části se zobrazí několik nejnovějších aktualit. Aktualita je zpráva týkající se obchodu, nebo čehokoli jiného co se týká tématu. Aktualita se zobrazí jako její nadpis, text, fotka (volitelně), autor a datum vložení. Součástí aktualit je odkaz na všechny aktuality. Počet aktualit na úvodní stránce lze měnit v konfiguračním souboru.

Pod aktualitami se zobrazí podobný výpis novinek a produktů v akci. Novinka se zobrazí jako název produktu, jeho fotka a cena. U akce se navíc zobrazí původní a nová snížená cena. Výpis produktu provádí společná funkce. Vzhledově stejný výpis tedy můžeme najít u novinek, akcí a u výpisu kategorie.

## **Aktuality**

Aktuality zobrazují stejné informace o aktualitě jako úvodní stránka. Rozdíl je ten, že se zobrazí všechny aktuality. Jsou rozděleny do stránek. V horní části se zobrazí odkazy na následující a předchozí stránky aktualit. Aktuality mohou být skryty. Tyto nejsou běžnému uživateli viditelné. Pro administrátora se zobrazí z rozdílným spodním orámováním.

## **Novinky a akce**

Stránka s celkovým výpisem všech novinek a akcí zobrazuje všechny novinky a akce definované administrátorem. Vše se zobrazí na jednu stránku, jelikož v systému by bylo nesmyslné mít definováno příliš mnoho novinek a aktualit. Ztrácel by se tak původní funkce a to upoutat pozornost zákazníka. Počet novinek a aktualit v systému není nijak omezen. Lze omezit pouze jejich počet na úvodní stránce. Akce vložené jako skryté se zobrazí jen administrátorovi.

## **Tipy**

Tipy jsou zobrazeny v pravém sloupci jako fotka a popis tipu. Fotka a název jsou odkazy na detail produktu.

## **Fórum**

Celý obchod by měl sloužit nejen jako místo, kde se dá něco koupit, ale také by sem měli lidé chodit jen tak něco sdělit, či se na něco zeptat. Proto jsem přidal i klasické diskuzní fórum.

Výhodou tohoto fóra je, že administrátor může příspěvky a případně odpovědi mazat. Takto může učinit stisknutím tlačítka u každého příspěvku. Toto tlačítko se zobrazí jen přihlášenému administrátorovi.

Na každý příspěvek lze odpovědět. V pravé části příspěvku je k tomu určené tlačítko.

## **Zobrazení produktů v dané kategorii**

Produkty v dané kategorii se zobrazí při stisku odpovídající položku v levém menu. V horní části je formulář pro vyhledávání produktů. Produkty lze vyhledat pouze v aktuální kategorii a to podle značky a rozmezí ceny. Pod tímto formulářem je stránkování. Produktů se očekává mnoho, a tak je stránkování nezbytnost. Počet zobrazených produktů lze měnit v konfiguračním souboru. Náhled produktu je stejný jako výše zmíněné novinky a akce. Pokud je název produktu příliš dlouhý, pak se zobrazí jen jeho část.

## **Zobrazení detailu produktů**

V levé části detailu produktu je jeho popis. Je to konkrétně název, značka, popis, typ a cena. V pravé

části se zobrazí případná fotka. Pokud produkt není zadán společně s fotkou je zobrazena implicitní náhrada. Pod popisem produktu je tlačítko sloužící k objednání. Pokud produkt obsahuje kusy se zadanými velikostmi, pak se zobrazí rolovací menu pro výběr konkrétního kusu. Za políčkem počtu kusů jsou dvě tlačítka dávající možnost změnit počet kusů. Rozmezí zadané hodnoty se pohybuje od 1 až po hodnotu změnitelnou v konfiguračním souboru. Následují případné fotky produktu. Jsou tím myšleny případné fotky jednotlivých kusů. Fotky jsou stejně jako fotka produktu zobrazeny jako náhled s odkazem na původní velikost. Pod fotkami se pro administrátora zobrazí menu úpravy produktu, jeho kusů, akcí, novinek a tipů. Zároveň se vypíše hodnota počtu zobrazení a počtu prodaných kusů produktu. Pod administrátorským menu je formulář pro vkládání komentářů k výrobku. Výrobek může komentovat kdokoli. Administrátor může příspěvky mazat.

### **Navigace**

Na každé stránce, která je zanořena v logické struktuře obchodu se v horní části zobrazí řádek navigace. Tento řádek zobrazí jednotlivé nadřazené sekce tak, jak jdou od úvodní stránky přes produkty až po jednotlivé akce u detailu produktu. Každá sekce se zobrazí jako odkaz na sebe samu. Aktuální sekce se zobrazí pouze jako text. Nad tímto řádkem je jako nadpis nejvyššího stupně zobrazen název aktuální sekce. To znamená jestli se jedná o detail produktu, či sekci kde přidávám kusy, akce, atd.

## **C.3 Uživatelská část**

### **Přihlašování**

Pro přihlašování jsem vyčlenil samostatnou stránku. Výhodou je, že se formulář pro přihlášení nemusí zobrazovat někde v horní nebo pravé sekci a nekazí to tak vzhled. Přihlašovací formulář obsahuje pouze pole email a heslo. Je také zobrazen odkaz na stránku, kde dojde k resetu hesla, pokud jej uživatel zapomněl.

### **Registrace**

Stránka s registrací obsahuje formulář. Prvky formuláře tvoří pole s požadovanými osobními údaji. Jsou jimi jméno, příjmení, město, ulice, číslo popisné, poštovní směrovací číslo, email, telefon, stát a heslo. Všechny údaje jsou povinné kromě ulice. Pokud je uživatel z vesnice pak ulici nemůže zadat.

### **Úprava osobních údajů**

Tato sekce zobrazuje podobný formulář jako nová registrace s tím, že do polí jsou vyplněny známé hodnoty. Pole heslo je blokováno pro úpravu. Až po kliknutí do zaškrťovacího políčka se pole pro

heslo otevře a uživatel je může znovu zadat. Administrátor může navíc měnit roli uživatele, tzn. jestli jde o administrátora či nikoli.

### **Zobrazení objednávek**

Každý přihlášený zákazník může zobrazit své předchozí objednávky. Objednávky se zobrazí jako stručný tabulkový výpis s odkazem na detail. V detailu jsou všechny údaje o objednávce. Uživatel nemá možnost jakkoli objednávku měnit. Administrátor může měnit strav objednávky a může ji také smazat. Pod detailem objednávky jsou zobrazeny jednotlivé položky. Název produktu je pak odkazem na jeho detail.

### **Košík**

Košík obsahuje informaci o množství a ceně produktu. Produkt je zobrazen jako název, ten je odkazem na svůj detail. Vedle názvu je případná velikost a barva konkrétního objednaného kusu. Položku košíku je možno smazat. Lze také změnit počet kusů změnou políčka množství a kliknutím na tlačítko přepočítat. Další dvě tlačítka slouží pro vyprázdnění a objednání zboží.

### **Objednávka**

Objednávka probíhá ve třech krocích. Nejprve jsou přidány položky do košíku a stiskne tlačítko objednat. Následuje vyplnění adresy na kterou bude zboží zasláno a typu dopravy. V posledním kroku se zobrazí všechny údaje o objednávce a uživatel je vyzván k jejímu potvrzení. Po potvrzení je uživatel upozorněn, zda objednávka byla či nebyla úspěšně provedena, a že potvrzení přijde na email.

### **Změna hesla**

Stránka se změnou hesla obsahuje výzvu k vyplnění jména, příjmení a emailu. Tyto tři údaje stačí k identifikaci zákazníka. Systém totiž nedovoluje registraci dvou různých emailů. Může nastat situace, že někdo zná jméno, příjmení a email osoby registrované v systému a zadáním těchto údajů může této osobě údaje změnit. Dotyčná osoba pak na svém emailu nalezne informaci, že jeho údaje byly změněny. Tento nedostatek lze odstranit doplňující otázkou, jak je tomu v podobných případech. Heslo systém vygeneruje náhodné, složené z písmen a znaků o délce nastavitelné v konfiguračním souboru. Základní délka je 8 znaků.

Po vyplnění údajů je změněn záznam v databázi a zaslán email s novým heslem uživateli.

## C.4 Část administrátora

### Přidat aktualitu

Přidání aktuality probíhá pomocí jednoduchého formuláře. Zadává se titulek, text, případná fotka a zadá se viditelnost. Úprava aktuality probíhá na totožném formuláři, jen lze zpřesnit zda chceme původní fotku smazat, změnit či zachovat.

### Přidat produkt

Přidání produktu je obdobné aktualitě. Navíc je třeba zvolit značku, druh, kategorii, cenu, název a popis. V popisu produktu je možno užít html značky. Html značky budou sloužit k lepší přehlednosti v popisu produktu. Zde je na administrátora kladen požadavek základní znalosti html kódu. K přehledné strukturovanosti textu je třeba jen několik základních formátovacích značek jako tlustý font `<b>`, nečíslovaný seznam `<ul>`, prvek seznamu `<li>`, kurzíva `<i>`, odkaz `<a href="">`. Kategorie produktu lze zvolit na stránce, která se otevře po stisknutí příslušného tlačítka. Kategorie jsou tam hierarchicky vypsané, podobně jako v levém menu. Produkt lze vkládat jen do kategorie, která nemá podřazenou kategorii. Fotka pro produkt i kusy je omezena velikostí 1500 kb, a rozměry 1600 x 1000 px. Tyto hodnoty lze měnit v konfiguračním souboru.

### Kusy

Operace s konkrétním produktem je možno provádět ze stránky jeho detailu. Po kliknutí na tlačítko upravit kusy se zobrazí tabulka se všemi kusy. Pod touto tabulkou je formulář pro přidání nového kusu. U kusu je nutno zadat velikost, barvu, dostupnost, množství a případnou fotku. Pokud zadáme množství 0 a zadáme fotku, pak je fotka zobrazena jako fotka produktu, přestože kus nelze objednat. Upravit kus lze na další stránce. Zde lze modifikovat množství prodaných kusů.

### Osoby

V části osoby si administrátor zobrazí registrované osoby. Osoba se zobrazí krátce na řádku a obsahuje odkaz na její smazání a zobrazení detailů. Osoby lze v této části řadit podle ID, jména, příjmení, města a emailu. Osoby jsou stránkovány.

Pokud administrátor klikne na odkaz zobrazení detailu, zobrazí se kompletní údaje o osobě. Zobrazí se také tlačítko pro úpravu údajů. Tato část je přístupná také osobě, která není administrátorem s tím rozdílem, že administrátor může upravovat a zobrazovat kohokoli, zákazník jen sebe.

### Objednávky

Strana s objednávkami zobrazí tabulkový výpis objednávek. Co řádek to objednávka. Objednávky lze

seřadit podle jména, příjmení, čp., data objednání, data poslední změny, počtu kusů a administrátora. Každý řádek obsahuje odkaz na detail. Detail obsahuje všechny dostupné údaje o objednavce, jejím autorovi, administrátorovi, který změnil její stav a o jednotlivých kusech. Na této stránce lze objednávku smazat.

### **Online uživatelé**

Uživatelé online jsou uživatelé, kteří vstoupili na stránky obchodu. Jsou jednoznačně identifikováni svým session id a uloženi v databázi. Uživatelé mohou být přihlášení i nepřihlášení. U přihlášených uživatelů se zobrazí odkaz na jejich detaily.

### **Chyby**

Stránka s chybami slouží ke kontrole korektní funkce obchodu. Za chybu je považována chyba systému. Chyba může vzniknout při nahrávání fotky na server, při práci s databází. Jsou to tedy chyby způsobené systémem nikoli uživatelem. Každá chyba má svoje identifikační číslo. Každá chyba má v systému přiděleno jednoznačné číslo. Podle něj lze přesně určit, která část kódu chybu způsobila. U chyby je uloženo datum jejího vzniku.

### **Kategorie**

Stránka s kategoriemi je určena pro přidání úpravu a smazání kategorie. U kategorie je nutné určit její nadřazenou kategorii. Jako tuto lze zvolit všechny kromě sebe sama. Cesta kategorie je určena z jejího názvu odstraněním nepovolených znaků. Za nepovolené znaky jsou považovány české znaky, interpunkce a speciální znaky jako ,/'%@\$^&=#\!|-;. Měnit kategorie lze přímo ve výpisu.

### **Velikosti**

Velikosti jsou zadány jen svým názvem. Změnit velikost je možno ve výpisu. První velikost je vložena při instalaci databáze. Tato velikost nelze smazat ani upravit. Je určena pro produkty u nichž nelze velikost určit.

### **Barvy**

U barvy lze kromě názvu zadat také její hexadecimální vyjádření. Tuto hodnotu lze použít v případě nepřítomnosti fotky produktu. V současné verzi tato hodnota není využita.

### **Značky**

U značky lze kromě názvu zadat i její logo. Upravit a smazat značku lze na zvláštních stránkách. Je tomu proto, že logo si vyžaduje více operací. Je tak zachována přehlednost. Na stránce pro úpravu

značky je možno změnit název a buďto zachovat, smazat, nebo vložit nový obrázek. Logo nesmí být větší, než 200x200 px a 50kB. Tyto hodnoty lze měnit v konfiguračním souboru. Obrázky jsou ukládány do adresáře /produkty/znacky/loga. Logo každé značky není v současné verzi využito. Využití by našlo ve výpisu dostupných značek, nebo přímo u detailu produktu.

### **Akce, tipy, novinky**

Na stránce detailu produktu lze modifikovat tyto tři kategorie. Každý produkt tedy lze nastavit jako novinku, akci, nebo tip. Pokud je parametr již nastaven pak jej lze upravit nebo smazat. U akce je požadována nová cena. Tato cena musí být nižší než ta stará. Datum platnosti lze vybrat ze stránky, která se otevře po stisku příslušného tlačítka. Kalendář je vytvářen automaticky do roku 9999.



# D Náhledy aplikace