

Webová administrace e-shopu

E-shop web administration

Bc. Kateřina Ježková

Diplomová práce
2007



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

*** nescannované zadání str. 1 ***

*** nescannované zadání str. 2 ***

ABSTRAKT

Aplikace „Webová administrace e-shopu“ slouží ke správě dat v Internetovém obchodě. Je to ta část, se kterou pracuje zaměstnanec Internetového obchodu, nikoliv zákazník. Aplikace umožňuje zadávání, vytváření, mazání a úpravy dat, jako jsou různé parametry zboží, objednávek, oddělení nebo například údaje o osobách. Kromě toho je zde možnost importu a exportu dat, především ceníků. Aplikace funguje přes webové rozhraní. Její data jsou uložena v databázi a obsluhována pomocí PHP skriptů. Jsou zde použity také technologie XML, XHTML a Javascript.

Klíčová slova: administrace, web, Internet, PHP, databáze, data

ABSTRACT

The application „E-shop web administration“ serves for data administration in an Internet shop. It is the part, an employee, not a customer of an Internet shop works with. The application enables entering, creating, deleting and change of data, such as various parameters of goods, orders, departments or for example human data. Besides, there is a possibility of data, especially price lists, import or export. The application works through the web interface. Its data are stored in a database and manipulated with the use of PHP scripts. Technologies of XML, XHTML and Javascript are also used here.

Keywords: administration, web, Internet, PHP, database, data

Chtěla bych poděkovat Ing. Zdence Prokopové CSc. za výborné vedení diplomové práce, Mgr. Zbyňku Divišovi za ochotu oponovat, Mgr. Evě Filípkové za kontrolu anglických překladů a neustálý přísun humoru a rodině a přátelům za podporu a za to, že to se mnou vydrželi.

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

Ve Zlíně

.....
Podpis diplomanta

OBSAH

OBSAH	6
ÚVOD.....	8
I. TEORETICKÁ ČÁST	10
1 PHP	11
2 ZNAČKOVACÍ JAZYKY	13
2.1 XML.....	13
2.2 XHTML.....	14
3 KASKÁDOVÉ STYLY.....	15
4 JAVASCRIPT	16
5 DATABÁZE.....	17
5.1 Databáze obecně.....	17
5.2 MySQL.....	17
6 NÁSTROJE PRO NÁVRH WEBOVÉ APLIKACE.....	19
II. PRAKTICKÁ ČÁST	20
7 NÁVRH DATABÁZE	21
7.1 Tabulky.....	21
7.1.1 Osoba	21
7.1.2 Obchod.....	21
7.1.3 Oddělení.....	22
7.1.4 Titulka.....	23
7.1.5 Zboží v oddělení	23
7.1.6 Zboží	24
7.1.7 Texty ke zboží.....	25
7.1.8 Typ zboží	25
7.1.9 Recyklační poplatek.....	26
7.1.10 Zboží dodavatele.....	26
7.1.11 Dodací lhůta.....	27
7.1.12 Název dodací lhůty	27
7.1.13 Ceny	28
7.1.14 Měna	29
7.1.15 Tvůrci zboží	29
7.1.16 Tvůrce	30
7.1.17 Typ tvůrce	30
7.1.18 Název typu tvůrce	31
7.1.19 Jazyk	31
7.1.20 Vlastnosti zboží	31
7.1.21 Vlastnost	32
7.1.22 Varianty zboží.....	32
7.1.23 Varianta.....	33
7.1.24 Příloha.....	33
7.1.25 Typ přílohy.....	34
7.1.26 Objednávka	34

7.1.27	Zboží objednávky.....	35
7.1.28	Stav objednávky.....	35
7.1.29	Název stavu objednávky	36
7.1.30	Zákazník.....	36
7.1.31	Skupina	37
7.1.32	Uživatel.....	37
7.1.33	Oprávnění uživatelů.....	37
7.2	Relace.....	38
8	GRAFICKÝ NÁVRH A OVLÁDÁNÍ APLIKACE	39
9	PROGRAMOVÁ STRUKTURA	43
9.1	Objektový návrh	43
9.2	Komunikace s databází.....	44
9.3	Generování XHTML	45
10	EXPORT A IMPORT DAT.....	47
10.1	Export dat	47
10.2	Import dat	49
11	UŽIVATELSKÁ PŘÍRUČKA	50
	ZÁVĚR	51
	ZÁVĚR V ANGLIČTINĚ.....	53
	SEZNAM POUŽITÉ LITERATURY.....	55
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	56
	SEZNAM OBRÁZKŮ	57
	SEZNAM PŘÍLOH.....	59

ÚVOD

Aby bylo možné provozovat Internetový obchod, je třeba vytvořit nejen vlastní webové stránky s nabízeným zbožím a možností objednávky, ale také administraci pomocí které budou tyto stránky spravovány.

Jestliže je vlastní obchod naprogramován pomocí určitých webových technologií, je výhodné, i když ne nutné, využít tyto technologie i pro naprogramování administrace. Pro každého je jistě důležitá finanční otázka, proto je dobré využít technologie, které jsou zdarma. Dobrou volbou se v tomto případě zdá skriptovací jazyk PHP a nějaká zdarma distribuovaná databáze, například MySQL.

Všechna data, která se mohou měnit by měla být uložena v databázi. Správný databázový návrh je tedy základem e-shopu a jeho administrace. V databázi by měla být uložena celá struktura dat e-shopu, například strom oddělení, typy, vlastnosti a varianty zboží, informace o uživatelích, dodavatelích a zákaznících apod. Dobré je nezapomínat ani na takové drobnosti jakou jsou stavy objednávek nebo dodací lhůty. Někdo by mohl polemizovat, že takové drobnosti je možné zahrnout do programového kódu, ale uchovávání těchto informací v databázi je přehlednější, jejich změna je snadnější a tato data mohou být navíc využívána i jinými systémy, např. logistickým nebo účetním.

Pro pohodlné zadávání dat uživatelem by mělo být vytvořeno vhodné rozhraní. Jak je napsáno výše, je výhodné k tomuto účelu využít webových technologií. Skripty by měly být napsány tak, aby bylo co nejsnadnější provádět případné úpravy nebo rozšíření aplikace. Proto je dobré programovat objektově. Také je třeba vytvořit třídy pro komunikaci s databází, protože poté, v případě změny databáze, stačí pouze upravit některé funkce v této třídě.

Co se týká grafiky administrace, měla by být co nejméně náročná, aby aplikace nebyla příliš pomalá, proto je lepší použít co nejméně obrázků. Ovládání by mělo být co nejsnazší a uživatelsky přívětivé. I když je rozhraní uživatelsky přívětivé, občas se stane, že uživatel neví jak dál. Proto je vhodné vytvořit uživatelskou příručku, která uživateli pomůže a navede jej správnou cestou.

Další důležitou věcí, která může značně usnadnit uživatelům život, je možnost exportu a importu dat. Takové ceníky dodavatelů jsou asi nejčastějšími importovanými daty, protože se velice často mění nejen ceny, ale například i dostupnost zboží a zapisovat

neustále tyto informace ručně by bylo dosti zdlouhavé. Ale import se může hodit i jindy, například při přechodu z jiné aplikace. Když je možné data importovat, pak je dobré vytvořit i možnost exportu. Opět se nabízí například ceník vybraného zboží, který bude rozesílán zákazníkům.

Poslední a důležitou věcí pro fungování administrace je její zabezpečení. To by mělo být v několika úrovních (žádná práva, pouze čtení ... administrátor). Pro malý obchod, který bude spravovat jeden uživatel ze svého počítače to sice není nezbytné, ale vhodné, ovšem pro více uživatelů se stává možnost přihlašování nutností.

I. TEORETICKÁ ČÁST

1 PHP

PHP je skriptovací jazyk, který je zpracováván na straně serveru a předává zpracované výsledky klientskému počítači. V současné době existuje nejnovější verze 5.2.2 [6]. PHP verze 5 je obohaceno oproti předchozím verzím o větší možnosti objektového programování.

Jedná se o volně šiřitelný software. Jeho další velkou výhodou je to, že je možné jej rozběhnout na různých serverech, takže je nezávislý na platformě. Nejčastěji však bývá využíván s webovým serverem Apache na operačním systému Linux.

PHP slouží k vytváření dynamických webových stránek. To znamená, že pomocí PHP můžete dynamicky generovat HTML kód. V dnešní době už si snad ani není možné představit větší web bez této vymoženosti.

PHP skripty je možné vpisovat přímo do HTML stránek a tím pádem generovat pouze část obsahu stránky dynamicky nebo psát „čisté“ PHP skripty.

I v PHP lze programovat objektově. Můžete vytvářet své vlastní třídy, jejich funkce a objekty a tím pádem se váš kód stane daleko přehlednějším než při klasickém procedurálním programování.

Sílnou stránkou PHP je nejen umožnění dynamičnosti, ale také možnost propojení s databázemi. Pro komunikaci s databázemi je možné využívat buď nativních funkcí PHP nebo propojení pomocí rozhraní ODBC.

Další předností je například možnost práce se soubory. Můžete mazat soubory na serveru, nahrávat nové pomocí webových formulářů nebo například procházet adresářovým stromem.

Pomocí PHP je také možné řešit zabezpečení celé aplikace. To často probíhá pomocí tzv. sessions. Ovšem je možné využít i zabezpečení pomocí HTTP protokolu. Sessions nejsou vhodné pro aplikace, které se snaží získat co nejvíce přístupů z vyhledávačů na jednu adresu, protože vytvářejí jednoznačnou adresu.

PHP je velice dobrým jazykem jak pro začátečníky, pro svou jednoduchost, která spočívá například v tom, že není třeba předem definovat typ proměnných. Také pro začátek můžete třeba jen generovat na stránce datum. Ovšem je stejně tak vhodný pro pokročilé,

pro svou obsáhlost. Pokročilé aplikace mohou být například kompletně naprogramovány objektově.

2 ZNAČKOVACÍ JAZYKY

2.1 XML

XML (Extensible Markup Language) je rozšiřitelný značkovací jazyk využívaný hlavně na webu pro uchovávání a výměnu informací [5]. XML neobsahuje předdefinované elementy jako například HTML, ale přesto má spoustu pravidel, která je třeba dodržet. Můžeme psát dokumenty XML vyhovující dvěma úrovním syntaktické příslušnosti. Podle této úrovně můžeme dokumenty rozdělit na správně strukturované dokumenty (well-formed document) a platné dokumenty (valid document).

Správně strukturované dokumenty obsahují XML prolog a jeden hlavní kořenový element, každý element musí být ukončen a správně zanořen (značky se nesmí křížit). Název typu elementu v počáteční značce se musí shodovat s názvem v ukončovací značce. V názvech typů elementů jsou rozlišována malá a velká písmena.

Platné dokumenty musí splňovat mnohem přísnější pravidla než správně strukturované dokumenty [5]. Aby byl dokument platný, obsahuje prolog správně nadefinovanou definici typu dokumentu (DTD) nebo je obsah a struktura dokumentu formátovaná podle schématu XML, které se nachází v samostatném souboru. Definice typu dokumentu přesně vymezuje elementy a atributy, které je možné použít.

Ovlivnit zobrazení XML dokumentu v prohlížeči je možné třemi různými způsoby.

Jedním z nich je Style sheet linking – napojování stylu. Styl je v samostatném souboru, který obsahuje příkazy pro formátování jednotlivých elementů XML. Je možné použít buď kaskádové styly (CSS) nebo XSLT (Extensible Stylesheet Language Transformation).

Další možností je vázání dat (Data Binding), což znamená svázání XML dokumentu se standardními elementy HTML.

Poslední možností je XML DOM Scripting, kde se pro zobrazení elementů vytvoří skripty v jazyce Javascript nebo VBScript.

2.2 XHTML

XHTML (Extensible Hypertext Markup Language) je jednou z možných aplikací XML. Ovšem jeho atributy jsou přesně vymezeny pomocí DTD.

3 KASKÁDOVÉ STYLY

Kaskádové styly (CSS – Cascading Style Sheet) představují v dnešní době hlavní mocný na tvorbu vzhledu HTML a XML dokumentů. Protože je snaha odlišit strukturu dokumentu od vzhledu, upouští se v HTML od používání značek na vytváření vzhledu (font, b, i apod) a místo toho se používají stále více kaskádové styly. V současné době existují CSS 2, ale už je v přípravě i verze 3.

Kaskádové styly je možné zapisovat třemi možnými způsoby. Jednak zápisem přímo k elementu pomocí atributu style nebo definicí do HTML hlavičky pomocí elementu style nebo do externího souboru. Nejvyšší váhu má přitom styl zapsaný nejpozději, tedy jednak nejpozději v rámci definice stylů a také atribut style „přebíjí“ vše ostatní.

Nevýhodou je stejně jako u Javascriptu nedokonalé prezentování stylů v jednotlivých prohlížečích.

4 JAVASCRIPT

Javascript je skriptovací jazyk zpracováváný na straně klienta. Využívá se v případech, kdy je zbytečné odesílat data na server a čekat na jejich odezvu.

Javascript je možné zapisovat přímo do HTML souboru nebo jej načítat z externího souboru. Velice často je používán k zjištění, zda jsou správně vyplněny informace ve formuláři ještě předtím než jsou data odeslána na server k dalšímu zpracování. Také bývá často používán ke grafickým změnám stránky (hlavně v kombinaci s kaskádovými styly) v závislosti na nějaké události.

Většinou to funguje tak, že nějakému elementu (například odkazu) a jeho události (například onclick) je přiřazena funkce v jazyce Javascript s akcí, která se má provést.

Bohužel ne všechny webové prohlížeče podporují Javascript stejně. Proto se mu někteří vývojáři dosti brání. I tak je ale dobré jej na vytváření některých drobných funkcionalit použít.

5 DATABÁZE

5.1 Databáze obecně

Databáze patří mezi jednu z nejpoužívanějších technologií vůbec. Existuje několik různých typů databází, tím nejpoužívanějším typem je jistě relační databáze [3].

V relační databázi jsou data uložena v tabulkách. Jednotlivé řádky tabulky představují jednotlivé záznamy a sloupce představují atributy neboli vlastnosti těchto záznamů.

Každá tabulka by měla obsahovat primární klíč, to znamená jednoznačnou identifikaci každého záznamu. Primární klíč může být jedna nebo více vlastností. Často se zřizuje primární klíč jako nějaké id.

Tabulky jsou vzájemně propojeny relacemi. Relace se realizují pomocí tzn. cizích klíčů. Cizí klíč je v jedné tabulce uložen jako odkaz na primární klíč jiné tabulky. Při správné práci s databází by měla být dodržena referenční integrita, což znamená, že pokud existuje cizí klíč jako odkaz na primární klíč a primární klíč bude změněn nebo smazán, pak musí být provedena taková opatření, aby se nestalo, že cizí klíč bude „směřovat nikam“.

Pro práci s daty v databázi je používán speciální jazyk SQL (Structured Query Language), který umožňuje vytvářet mazat a upravovat databáze, tabulky a vlastní data v tabulkách.

Další možností databází je například vytváření indexů. Pod tímto pojmem se skrývá dodatečná informace uložená v databázi o třídění sloupců a nebo skupin sloupců v tabulce, slouží pro zefektivnění a urychlení vyhledávání informací v databázi [9].

Samozřejmě jazyk SQL neobsahuje jen příkazy na úpravu dat, ale jeho silnou stránkou jsou především široké možnosti vyhledávání, filtrování a řazení dat.

5.2 MySQL

Databáze MySQL je nejčastěji používaná k uchovávání dat pro webové aplikace. Její propojení s PHP skripty a serverem Apache je velice častou záležitostí. PHP dokáže s touto databází komunikovat dokonce dvěma způsoby a to buď pomocí nativních funkcí nebo

pomocí rozhraní ODBC. Stejně jako PHP i databáze MySQL je zdarma. V základní verzi můžete tuto databázi obsluhovat buď pomocí příkazového řádku nebo ve spojení s PHP pomocí aplikace PHPMysqlAdmin, kde vše názorně vidíte a na základní operace nemusíte znát ani jazyk SQL.

6 NÁSTROJE PRO NÁVRH WEBOVÉ APLIKACE

Při vytváření webové aplikace s databází si sice můžete vystačit s použitím Notepadu, ale to je značně nepohodlné. Naštěstí existuje celá řada nástrojů, které nám práci usnadní.

Pro návrh databáze existují různé nástroje. Jedním z takových šikovných je CASE Studio 2, ve kterém se pracuje velice dobře. Tabulky jsou zobrazeny na kreslicí ploše spolu s relacemi, takže je krásně vidět jak se databáze vyvíjí. Navíc nemusíte nic psát v SQL, protože podle grafického návrhu databáze vám program sám vygeneruje skripty na vytvoření databáze a zkontroluje správnost databázového návrhu. Navíc si můžete vyexportovat grafický návrh jako obrázek a posléze vytisknout.

Pro vytváření PHP skriptů a HTML existuje také řada nástrojů. Jedním z nich je Zend studio, které umí i ladit PHP kód. Ovšem to není zdarma. Zdarma je například PSPad, který sice nemá debugger, ale velice pěkně se v něm píšou PHP skripty a HTML soubory. Nejen to, má zabudované i zvýrazňování syntaxí jiných typů souborů, umožňuje upravovat kód a velice snadno měnit kódování souboru.

II. PRAKTICKÁ ČÁST


7 NÁVRH DATABÁZE

Návrh databáze je jednou z nejdůležitějších částí návrhu e-shopu. Databáze je základním kamenem aplikace a pozdější změny v její struktuře by v některých případech znamenaly velké množství dalších změn i v programovém kódu. Proto je dobré si dát při návrhu obzvlášť záležet a přemýšlet i o případných možnostech co nejsnažšího rozšíření v budoucnu.

7.1 Tabulky

7.1.1 Osoba

V tabulce *Osoba* jsou evidovány všechny „zúčastněné“ osoby. Osoba může být právnická nebo fyzická, může se jednat o majitele, dodavatele, zákazníka e-shopu nebo uživatele aplikace. Povinný e-mail zaručuje možnost kontaktovat danou osobu. Ostatní údaje jsou nepovinné.

Název entity		Název tabulky							
Osoba		osoba							
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis		
1		Osoba	osoba	Integer	Not Nu	Unique			
2		Jméno	jmeno	Varchar(20)					
3		Příjmení	prijmeni	Varchar(20)					
4		Email	email	Varchar(100)	Not Nu				
5		Mobil	mobil	Char(14)					
6		Telefon	telefon	Char(14)					
7		Ulice	ulice	Varchar(30)					
8		Číslo popisné	cislo_p	Varchar(10)					
9		Město	mesto	Varchar(30)					
10		PSČ	psc	Varchar(6)					
11		Země	zeme	Varchar(100)					
12		Název firmy	firma	Varchar(100)					
13		IČO	ico	Varchar(18)					
14		DIČO	dico	Varchar(20)					

Obr. 1. Tabulka *Osoba*

7.1.2 Obchod

Tato tabulka obsahuje jednoznačnou specifikaci obchodu atributem *Obchod* a údaje o obchodu – *Název*, *Počet zboží*. Druhý z atributů značí celkový počet aktivního zboží, tedy

toho, co se právě prodává. Dále je zde uveden atribut *Majitel* jako odkaz na majitele obchodu. Jeden majitel může mít více obchodů. Je nepravděpodobné, že by jeden obchod vlastnilo více majitelů. Pro tuto situaci by bylo třeba přidat další tabulku. Atributy *Měna* a *Jazyk* značí defaultní jazyk a měnu pro obchod.

Název entity		Název tabulky							
Obchod		obchod							
<div style="display: flex; justify-content: space-between;"> ➤ </div>									
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis		
1		Obchod	obchod	Integer	Not Nu	Unique			
2		Název	nazev	Varchar(100)	Not Nu				
3		Popis	popis	Text					
4		Majitel	majitel	Integer	Not Nu				
5		Měna	mena	Mediumint	Not Nu				
6		Jazyk	jazyk	Integer	Not Nu				

Obr. 2. Tabulka Obchod

7.1.3 Oddělení

Obchod je dále rozdělen na oddělení. Každé oddělení spadá pod nadřazené oddělení. Hlavní oddělení mají jako nadřazené oddělení nastaveno jedno velké oddělení, označené 0 do kterého spadá všechno zboží z celého obchodu. Oddělení mohou a nemusí v daném okamžiku obsahovat aktivní zboží. Atribut *Počet zboží* značí, stejně jako u obchodu, právě prodávané zboží v daném oddělení a jeho pododděleních. Některá oddělení jsou výprodejová – atribut *Výprodej* je nastaven na 1. Atribut *Titulka* nastavený na 1 značí oddělení, které zobrazuje titulní stranu s vybraným zbožím z jeho pododdělení. Titulní strany jsou uloženy v tabulce *Titulka*. *Pořadí* říká, jak budou oddělení zobrazena za sebou v rámci svého nadřazeného oddělení.

Atributy		Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis			
1	PK	Oddělení	oddeleni	Integer	Not Nu	Unique				
2		Obchod	obchod	Integer	Not Nu					
3		Výprodejové oddě	vyprodej	Bit(1)	Not Nu					
4		Nadřazené odděl	nadrazene	Integer	Not Nu					
5		Název	nazev	Varchar(100)	Not Nu					
6		Počet zboží	pocet	Integer						
7		Titulka	titulka	Bit(1)	Not Nu					
8		Pořadí	poradi	Mediumint						

Obr. 3. Tabulka Oddělení

7.1.4 Titulka

V tabulce *Titulka* jsou uloženy sestavené titulní stránky oddělení. Primárním klíčem je dvojice *Oddělení*, *Zboží*. Je zde uloženo pouze zboží, které se v daném (titulkovém) oddělení objeví na titulní straně. Atribut *Typ zobrazení* udává jak bude zboží na titulní straně vypadat. Například typ zobrazení 1 bude velký obrázek a jen 2 kusy zboží vedle sebe, typ 2 bude jen odkaz v seznamu apod. Atribut *Pořadí* určuje v jakém pořadí bude zboží na titulní straně zobrazeno.

Atributy		Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis			
1	PK	Oddělení	oddeleni	Integer	Not Nu					
2	FK	Zboží	zbozi	Integer	Not Nu					
3		Typ zobrazení	typ_z	Tinyint	Not Nu					
4		Pořadí	poradi	Tinyint						

Obr. 4. Tabulka Titulka

7.1.5 Zboží v oddělení

Tato tabulka je propojením mezi tabulkami *Zboží* a *Oddělení*. Každé zboží musí být zařazeno v nějakém oddělení, může být i ve více odděleních.

Název entity		Název tabulky					
Zboží v oddělení		zbozioddeleni					
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>							
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	PK	Oddělení	oddeleni	Integer	Not Nu		
2	FK	Zboží	zbozi	Integer	Not Nu		

Obr. 5. Tabulka Zboží v oddělení

7.1.6 Zboží

Tabulka *Zboží* obsahuje údaje o daném zboží, které nejsou závislé na jazyku, měně nebo dodavateli. *Typ* odkazuje do tabulky *Typ zboží*, *Sleva v procentech* je sleva poskytovaná prodejcem v e-shopu. Je to sleva oproti doporučené maloobchodní ceně. Atribut *Zboží je ve výprodeji* označuje zboží, které je určené k výprodeji, většinou se jedná o poslední kusy zboží buď skladem u prodejce nebo dodavatele, které již nebudou v brzké době dodávány (např. modely loňské sezóny apod.). Atribut *Je zboží na prodej* označuje stav zboží v dané chvíli. Například 0 znamená, že je zboží povoleno k prodeji, 1 je to novinka (stav pro kontrolu novinek, než jsou povoleny k prodeji), 2 zboží je zakázáno, což znamená, že není nabízeno k prodeji. Stav se mohou libovolně měnit. Zboží které je zakázáno může být opět povoleno (např. jeden měsíc dodavatel nenabízel určité zboží a další měsíc ho má opět v nabídce). *Počet kusů skladem* je počet kusů skladem u prodejce. *Velikost zboží* určuje zda lze nebo nelze zboží poslat poštou jako balík. *Datum poslední změny automaticem* zaznamenává poslední změnu zboží při automatickém importu a *Datum poslední změny ručně* při ručním zápisu přes administraci. *Naposledy zakázáno* a *Naposledy povoleno* jsou data změny stavu na zakázáno nebo povoleno. Poslední dva atributy *Zakázat automatické ceny* a *Zakázat automatické změny ceny* se používají pouze při automatickém importu. Je totiž možné, že u některého zboží se budou ceny nastavovat nestandardním způsobem (jinak než výpočtem z nákupní ceny, rabatu apod.) nebo že zboží například nemá být povoleno i když ho dodavatel v ceníku nabízí.

Název entity		Název tabulky								
Zboží		zbozi								
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky	
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis			
1	PK	Zboží	zbozi	Integer	Not Nu					
2	FK	Typ	typ	Tinyint	Not Nu		Typ zboží			
3		EAN	ean	Varchar(20)						
4		ISBN	isbn	Varchar(20)						
5		Sleva v procentech	sleva	Tinyint	Not Nu					
6		Zboží je ve výprodeji	vyprodej	Bit(1)			0 ne 1 ano			
7		Je zboží na prodej	stav	Tinyint	Not Nu		Stav 0 ... povoleno k prodeji			
8		Počet kusů skladem	sklad	Integer	Not Nu					
9		Velikost zboží	velikost	Bit(1)			Atribut pro způsob přepravy			
10		Datum importu	d_import	Datetime						
11		Datum poslední změny	d_posledni_auto	Datetime						
12		Datum poslední změny	d_posledni_rucne	Datetime						
13		Naposledy zakázáno	d_zakazano	Datetime						
14		Naposledy povoleno	d_povoleno	Datetime						
15		Zakázat automaticky	zakazat_auto	Bit(1)	Not Nu		0 povoleno 1 zakazany z			
16		Zakázat automaticky	zakazat_auto_c	Bit(1)	Not Nu					

Obr. 6. Tabulka Zboží

7.1.7 Texty ke zboží

Zde se zapisují názvy a popisy ke zboží v různých jazycích. Primárním klíčem je proto dvojice *Zboží, Jazyk*.

Název entity		Název tabulky								
Texty ke zboží		texty								
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky	
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis			
1	PK	Zboží	zbozi	Integer	Not Nu					
2	FK	Jazyk	jazyk	Integer	Not Nu					
3		Název	nazev	Varchar(100)	Not Nu					
4		Popis	popis	Text						

Obr. 7. Tabulka Texty ke zboží

7.1.8 Typ zboží

Typem zboží se myslí, o jaké zboží se jedná. Například „mikrovltnka“, „hračka“, „doplňek stravy“ apod. Atribut *Minimální počet kusů skladem* udává kolik kusů musí být skladem u dodavatele, aby mohlo být zboží označeno že je „skladem“. Pokud se jedná o

nějaké zboží, co se často odebírá, mohlo by se stát, že jen dodavatel vyprodá dřív než bude objednáno a tudíž by pak došlo ke zpoždění nebo dokonce stornování objednávky.

Název entity		Název tabulky				
Typ zboží		zbozityp				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Typ	typ	Tinyint	Not Nu	Unique	
2	Název	nazev	Varchar(100)	Not Nu		
3	Minimální počet k	sklad_min	Integer	Not Nu		

Obr. 8. Tabulka Typ zboží

7.1.9 Recyklační poplatek

Recyklační poplatek je evidován pro různé typy elektrických spotřebičů. Podle typu se liší i výše poplatku. Poplatky se však nevztahují ke každému zboží, proto jsou v samostatné tabulce.

Název entity		Název tabulky				
Recyklační poplatek		recpoplatek				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Typ	typ	Tinyint	Not Nu		
2	Poplatek v Kč	poplatek	Decimal(10,2)	Not Nu		

Obr. 9. Tabulka Recyklační poplatek

7.1.10 Zboží dodavatele

Tato tabulka zahrnuje zboží nabízené dodavateli. Jedno zboží může dodávat více dodavatelů. Každý dodavatel má ke zboží přiřazen svůj kód – *Kód dodavatele*. Kódy dodavatelů na stejné zboží se mohou shodovat ale i lišit. Pokud se liší, může to způsobovat problémy při automatickém importu, protože může nastat situace, kdy není možné zjistit, že se jedná o stejné zboží, které je již evidováno od jiných dodavatelů. Při takové situaci je třeba zaevidovat zboží ručně a nastavit mu patřičný *Interní kód* (kód již existujícího zboží). *Rabat v procentech* můžou mít dodavatelé různý na různé zboží (např. ve smlouvě bude mít dodavatel stanoveno, že na hračky má prodejce 5% a na drogerii 10%). *Počet kusů skladem u dodavatele* není třeba dále rozvádět. Atribut *Aktuálně je vybráno* může být

nastaven na 1 vždy jen u jednoho dodavatele konkrétního zboží. Pro ostatní dodavatele je tento atribut nastaven na 0. U dodavatele u kterého je nastaveno 1 bude zboží objednáno, protože se jedná o „nejlepšího dodavatele“. Nejlepší dodavatel je vybrán podle nejlepší ceny a dodací lhůty, popř. ručně podle dalších kritérií.

Název entity		Název tabulky							
Zboží dodavatele		zboziodavatel							
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis			
1	Kód dodavatele	koddodavatele	Char(20)	Not Nu					
2	Dodavatel	dodavatel	Integer	Not Nu					
3	Interní kód	zbozi	Integer	Not Nu					
4	Rabat v procentech	rabat	Tinyint	Not Nu					
5	Počet kusů sklad	sklad_d	Integer	Not Nu					
6	Aktuálně je vybrán	aktualni	Bit(1)	Not Nu					
7	Dodací lhůta	dodlhuta	Tinyint	Not Nu					

Obr. 10. Tabulka Zboží dodavatele

7.1.11 Dodací lhůta

Dodací lhůta je doba od objednání zboží až po dopravení zboží na sklad prodejce. Protože si zákazníci mohou odebírat zboží osobně nebo si jej mohou nechat poslat, není do této lhůty započítána cesta zboží ze skladu prodejce k zákazníkovi. Atribut *Dodací lhůta* je pouze číslo, které je jednoznačný identifikátor, označující tuto dodací lhůtu. Atribut *Maximální počet dní* určuje po jaké době je propásnuta dodací lhůta (např. 1, 6, ...).

Název entity		Název tabulky							
Dodací lhůta		dodacilhuta							
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis			
1	Dodací lhůta	dodlhuta	Tinyint	Not Nu	Unique				
2	Maximální počet dní	maxdni	Tinyint	Not Nu					

Obr. 11. Tabulka Dodací lhůta

7.1.12 Název dodací lhůty

Zde se nacházejí názvy specifikující různé dodací lhůty (např. 2 – 6 dní, 24 hodin, ...) v různých jazycích.

Název entity		Název tabulky	
Název dodací lhůty		nazevdodlhuty	

Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis			
1	Dodací lhůta	dodlhuta	Tinyint	Not Nu					
2	Jazyk	jazyk	Integer	Not Nu					
3	Název	nazev	Varchar(100)	Not Nu					

Obr. 12. Tabulka Název dodací lhůty

7.1.13 Ceny

V tabulce *Cena* je primárním klíčem trojice *Měna*, *Kód dodavatele* a *Dodavatel*. *Nákupní cena bez DPH* je cena za kterou je zboží nakupováno od dodavatele. *Doporučená cena bez DPH* a *Doporučená cena s DPH* jsou ceny, které také určuje dodavatel. *DPH v procentech* je dáno zákonem. *Prodejní cena bez DPH* a *Prodejní cena s DPH* jsou ceny které jsou vypočítávány buď na základě rabatu z tabulky *Zboží dodavatele*, slevy z tabulky *Zboží* a nákupní ceny nebo ze slevy s doporučené ceny, záleží na dohodě s dodavatelem. Prodejní ceny by sice nemusely být uloženy v databázi, protože je lze vypočítat, ale jsou zde z důvodu zrychlení zobrazení, protože přepočítávat pokaždé všechny ceny by bylo velice zdlouhavé.

Název entity: Název tabulky:

Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky

	Klíč	Název	Název sloupce	Typ	Not Null	Unique	Popis
1	PK	Měna	mena	Mediumint	Not Null		
2	FK	Kód dodavatele	koddodavatele	Char(20)	Not Null		
3	FK	Dodavatel	dodavatel	Integer	Not Null		
4		Nákupní cena bez DPH	cena_n	Decimal(10,2)	Not Null		
5		Doporučená cena bez DPH	cena_d	Decimal(10,2)	Not Null		
6		Doporučená cena s DPH	cena_d_dph	Decimal(10,2)	Not Null		
7		DPH v procentech	dph	Tinyint	Not Null		
8		Prodejní cena bez DPH	cena_p	Decimal(10,2)	Not Null		
9		Prodejní cena s DPH	cena_p_dph	Decimal(10,2)	Not Null		

Obr. 13. Tabulka Cena

7.1.14 Měna

Tato tabulka představuje seznam použitých měn, kde *Název* popisuje měnu (např. Koruna česká) a *Jednotka* popisuje kód (např. CZK). *Měna* je číslo – jednoznačný identifikátor.

Název entity: Název tabulky:

Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky

	Klíč	Název	Název sloupce	Typ	Not Null	Unique	Popis
1	PK	Měna	mena	Mediumint	Not Null	Unique	
2		Název	nazev	Varchar(20)			
3		Jednotka	jednotka	Varchar(7)	Not Null		

Obr. 14. Tabulka Měna

7.1.15 Tvůrci zboží

Tvůrci zboží je název tabulky, která spojuje tabulku *Zboží* s tabulkou *Tvůrce*. Jedno zboží může mít více tvůrců a jeden tvůrce se může vyskytovat u většího množství zboží.

Název entity		Název tabulky				
Tvůrci zboží		zbozitivurce				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Tvůrce	tvurce	Integer	Not Nu		
2	Zboží	zbozi	Integer	Not Nu		

Obr. 15. Tabulka Tvůrce zboží

7.1.16 Tvůrce

Tvůrcem může být výrobce, autor, interpret, dirigent apod. Každé zboží může mít více tvůrců i stejného typu a jeden tvůrce může být u více než jednoho zboží. Atributy *Web* a *Obrázek loga* jsou odkazy na doplňující informace, které se často nacházejí především u výrobců.

Název entity		Název tabulky				
Tvůrce		tvurce				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Tvůrce	tvurce	Integer	Not Nu	Unique	
2	Název	nazev	Varchar(100)	Not Nu		
3	Web	web	Varchar(150)			
4	Obrázek loga	logo	Varchar(150)			
5	Typ	typ	Integer	Not Nu		

Obr. 16. Tabulka Tvůrce

7.1.17 Typ tvůrce

Typ tvůrce určuje o jakého tvůrce se jedná, zda je to autor, výrobce, interpret apod. Atribut *Pořadí zobrazení* určuje jak budou zobrazeny jednotlivé typy tvůrců za sebou. Pokud má zboží více tvůrců stejného typu, pak jsou tito tvůrci řazeni v rámci jednoho typu podle názvu.

Název entity		Název tabulky					
<input type="text" value="Typ tvůrce"/>		<input type="text" value="tvurcetyp"/>					
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>							
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1		Typ	typ	Integer	Not Nu	Unique	
2		Pořadí zobrazení	poradi	Mediumint	Not Nu		

Obr. 17. Tabulka Typ tvůrce

7.1.18 Název typu tvůrce

Zde se zadávají různé jazykové varianty názvů typů tvůrců. Název musí být definován pro každý typ alespoň v jednom jazyce.

Název entity		Název tabulky					
<input type="text" value="Název typu tvůrce"/>		<input type="text" value="tvurcetypnazev"/>					
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>							
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1		Jazyk	jazyk	Integer	Not Nu		
2		Typ	typ	Integer	Not Nu		
3		Název	nazev	Varchar(100)	Not Nu		

Obr. 18. Tabulka Název typu tvůrce

7.1.19 Jazyk

Tuto tabulku není třeba nějak složitě komentovat, obsahuje identifikátor jazyka a název jazyka ve vlastním jazyce.

Název entity		Název tabulky					
<input type="text" value="Jazyk"/>		<input type="text" value="jazyk"/>					
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>							
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1		Jazyk	jazyk	Integer	Not Nu	Unique	
2		Název jazyka	nazev	Varchar(100)	Not Nu		

Obr. 19. Tabulka Jazyk

7.1.20 Vlastnosti zboží

Většina zboží má nějaké vlastnosti. Podle těchto vlastností bývá zboží často vyhledáváno nebo řazeno. Každé zboží může mít více vlastností, ale každou pouze jednou.

Např. u televize je typickou vlastností velikost úhlopříčky. Vlastnosti mohou mít v jiném jazyce jinou hodnotu. Typickým příkladem jsou vlastnosti, jejichž hodnota je Ano/Ne, ale může se jednat i o zobrazení dané vlastnosti v jiných jednotkách podle národních zvyklostí.

Název entity		Název tabulky				
vlastnosti zboží		zbozivlastnosti				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Zboží	zbozi	Integer	Not Nu		
2	Vlastnost	vlastnost	Integer	Not Nu		
3	Jazyk	jazyk	Integer	Not Nu		
4	Hodnota	hodnota	Varchar(200)			

Obr. 20. Tabulka Vlastnosti zboží

7.1.21 Vlastnost

I vlastnosti jsou pojmenovány v různých jazycích různě. Toto je tabulka jazykových variant vlastností.

Název entity		Název tabulky				
vlastnost		vlastnost				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Vlastnost	vlastnost	Integer	Not Nu		
2	Jazyk	jazyk	Integer	Not Nu		
3	Název	nazev	Varchar(100)	Not Nu		

Obr. 21. Tabulka Vlastnost

7.1.22 Varianty zboží

Varianty zboží jsou vlastnosti zboží, které mohou pro jedno konkrétní zboží nabývat více různých hodnot. To je typické například pro oblečení (velikost, barva, obvod pasu atd.). Pokud má zboží varianty, jedná se stále o jedno konkrétní zboží, ale zákazník si může vybrat v jaké variantě zboží požaduje. Do objednávky se pak za kód zboží doplní i jeho varianta a tak vznikne speciální kód, který charakterizuje zboží v dané variantě. Bylo by však zbytečné tímto způsobem evidovat zboží v celém obchodě, protože by tak vznikalo mnoho redundantních dat. Protože varianta má více možných hodnot (pokud má jen jednu, jedná se o vlastnost), pak je v této tabulce primárním klíčem celá čtveřice *Zboží, Varianta,*

Jazyk a Hodnota. Stejně jako u vlastností zboží i hodnoty variant se mohou lišit v závislosti na jazyku.

Název entity		Název tabulky	
Varianty zboží		zbozivarianty	

Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis		
1		Zboží	zbozi	Integer	Not Nu				
2		Varianta	varianta	Integer	Not Nu				
3		Jazyk	jazyk	Integer	Not Nu				
4		Hodnota	hodnota	Varchar(200)	Not Nu				

Obr. 22. Tabulka Varianty zboží

7.1.23 Varianta

Tabulka obsahující názvy variant v různých jazycích.

Název entity		Název tabulky	
Varianta		varianta	

Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis		
1		Varianta	varianta	Integer	Not Nu				
2		Jazyk	jazyk	Integer	Not Nu				
3		Název	nazev	Varchar(100)					

Obr. 23. Tabulka Varianta

7.1.24 Příloha

Do této tabulky se zadávají všechny přílohy zboží jako jsou například trailery k filmům, mp3 ukázky skladeb, náhledy knih, obrázky atd. Atribut *Soubor s přílohou* obsahuje cestu k souboru na serveru, ve kterém se příloha nachází. Protože na externí odkazy často nebývá spoleh, je lepší co nejvíce příloh ukládat na server. *Pořadí* určuje jak budou za sebou zobrazeny jednotlivé přílohy. Samozřejmě vlastní zobrazení závisí také na typu přílohy (mp3, obrázek, ...). *Zobrazované pořadí* je určené pro číslované položky, např. seznam skladeb na CD.

Název entity		Název tabulky	
Příloha		priloha	

Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis		
1	PK	Příloha	priloha	Integer	Not Nu	Unique			
2		Typ	typ	Tinyint	Not Nu				
3		Zboží	zbozi	Integer	Not Nu				
4		Soubor s přílohou	soubor	Varchar(300)	Not Nu				
5		Pořadí	poradi	Mediumint					
6		Zobrazované poř	poradi_z	Varchar(10)					

Obr. 24. Tabulka Příloha

7.1.25 Typ přílohy

Tabulka *Typ přílohy* slouží k zadání seznamu možných typů příloh, tedy zda se jedná o mp3, obrázek nebo například trailer k filmu.

Název entity		Název tabulky	
Typ Přílohy		prilohatyp	

Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis		
1	PK	Typ	typ	Tinyint	Not Nu	Unique			
2		Název	nazev	Varchar(100)	Not Nu				

Obr. 25. Tabulka Typ přílohy

7.1.26 Objednávka

V této tabulce se zaznamenávají zákaznické objednávky. Je důležité evidovat, kdy bylo zboží objednáno a kdy dojde podle nejdelší dodací lhůty na zboží k expiraci objednávky. Z objednávky také musí být patrné do kterého obchodu patří a v jaké měně se rozhodl zákazník platit (to je zvlášť důležité v případě, že obchod nabízí možnost platit ve více měnách). Atribut *Stav* směřuje do tabulky stavů a říká jak daleko je objednávka v „koloběhu“.

Název entity		Název tabulky				
Objednávka		objednavka				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Objednávka	objednavka	Integer	Not Nu	Unique	
2	Obchod	obchod	Integer	Not Nu		
3	Zákazník	zakazanik	Integer	Not Nu		
4	Objednáno	objednano	Datetime	Not Nu		
5	Expirace	expirace	Datetime	Not Nu		
6	Měna	mena	Mediumint	Not Nu		
7	Stav	stav	Tinyint	Not Nu		

Obr. 26. Tabulka Objednávka

7.1.27 Zboží objednávky

Tabulka představuje jednotlivé položky vlastní objednávky. Primárním klíčem je dvojice *Zboží*, *Objednávka*. Je důležité zachytit stav zboží přesně ve chvíli, kdy bylo objednáno zákazníkem. Proto jsou zde atributy *Za cenu bez DPH*, *Za cenu s DPH* a *Dodavatel*, protože ty se v průběhu času mohou měnit. Atributy *Název v objednávce* a *Objednací kód* jsou *Název* z tabulky *Texty ke zboží* a *Zboží* z tabulky *Zboží*, doplněné o kódy a texty variant.

Název entity		Název tabulky				
Zboží objednávky		objednavkazbozi				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Objednávka	objednavka	Integer	Not Nu		
2	Zboží	zbozi	Integer	Not Nu		
3	Počet kusů	kusy	Mediumint	Not Nu		
4	Za cenu bez DPH	o_cena_p	Decimal(10,2)	Not Nu		
5	Za cenu s DPH	o_cena_p_dph	Decimal(10,2)	Not Nu		
6	Název v objednávce	o_nazev	Varchar(255)	Not Nu		
7	Objednací kód	o_kod	Varchar(255)	Not Nu		
8	Dodavatel	dodavatel	Integer	Not Nu		

Obr. 27. Tabulka Zboží objednávky

7.1.28 Stav objednávky

Tato tabulka zobrazuje seznam stavů od objednání až po vyexpedování zákazníkovi.

Název entity		Název tabulky				
Stav objednávky		objednavkastav				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Stav	Stav	Tinyint	Not Nu	Unique	

Obr. 28. Tabulka Stav objednávky

7.1.29 Název stavu objednávky

Zde jsou jazykové varianty názvu stavů objednávky. Takže například v češtině jsou názvy stavu „objednáno“, „skladem v Brně“ apod.

Název entity		Název tabulky				
Název stavu objednávky		objednavkastavnazev				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Stav	Stav	Tinyint	Not Nu		
2	Jazyk	jazyk	Integer	Not Nu		
3	Název	nazev	Varchar(50)	Not Nu		

Obr. 29. Tabulka Název stavu objednávky

7.1.30 Zákazník

Tabulka osob, které jsou zákazníci. Každý zákazník musí mít svůj *Login* a *Heslo*, aby mohl nakupovat v obchodě. Atribut *Skupina* říká o jakého zákazníka se jedná, některým skupinám mohou být poskytovány různé výhody.

Název entity		Název tabulky				
Zákazník		zakaznik				
<div style="display: flex; justify-content: space-between;"> Atributy Vlastnosti Rozšířené Indexy Alternativní klíče Ostatní Relace Storage Popis Poznámky </div>						
Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis
1	Zákazník	zakaznik	Integer	Not Nu	Unique	
2	Skupina	skupina	Tinyint	Not Nu		
3	Login	login	Varchar(20)	Not Nu	Unique	
4	Heslo	heslo	Varchar(200)	Not Nu		
5	Osoba	osoba	Integer	Not Nu		

Obr. 30. Tabulka Zákazník

7.1.31 Skupina

Tabulka popisuje skupiny zákazníků jako jsou např. „dealer A“, „obyčejný“, „zaměstnanec“ apod. *Sleva v procentech* se přičítá k normální slevě na zboží a tím pádem jsou zákazníci v určité skupině v nákupu zvýhodnění.

Název entity		Název tabulky							
Skupina		Skupina							
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis		
1	PK	Skupina	skupina	Tinyint	Not Nu				
2		Název	nazev	Varchar(20)	Not Nu				
3		Sleva v procentech	sleva	Tinyint	Not Nu				

Obr. 31. Tabulka Skupina

7.1.32 Uživatel

Tato tabulka je seznamem uživatelů, kteří mají přístup do aplikace. Každý uživatel má přidělený svůj vlastní *Login* a *Heslo* a oprávnění podle tabulky *Oprávnění uživatelů*. Atribut *Osoba* odkazuje na osobní údaje uživatelů.

Název entity		Název tabulky							
Uživatel		uzivatel							
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Nu	Unique	Popis		
1	PK	Uživatel	uzivatel	Integer	Not Nu	Unique			
2		Login	Login	Varchar(20)	Not Nu	Unique			
3		Heslo	Heslo	Varchar(200)	Not Nu				
4	FK	Osoba	Osoba	Integer	Not Nu				
5	FK	Oprávnění	opraveneni	Tinyint	Not Nu				

Obr. 32. Tabulka Uživatel

7.1.33 Oprávnění uživatelů

V tabulce se ukládají všechna oprávnění, která může uživatel nabýt. Oprávněním se myslí oprávnění na práci v programu, tedy zdali uživatel může jen prohlížet položky v programu nebo jich část upravovat nebo má právo provádět veškeré možné uživatelské operace.

Název entity		Název tabulky							
Oprávnění uživatelů		opraveneni							
<div style="display: flex; justify-content: space-between;"> ► </div>									
Atributy	Vlastnosti	Rozšířené	Indexy	Alternativní klíče	Ostatní	Relace	Storage	Popis	Poznámky
	Klíč	Název	Název sloupce	Typ	Not Null	Unique	Popis		
1	Prim	Oprávnění	opraveneni	Tinyint	Not Null	Unique			
2		Název	nazev	Varchar(20)					

Obr. 33. Tabulka Oprávnění uživatelů

7.2 Relace

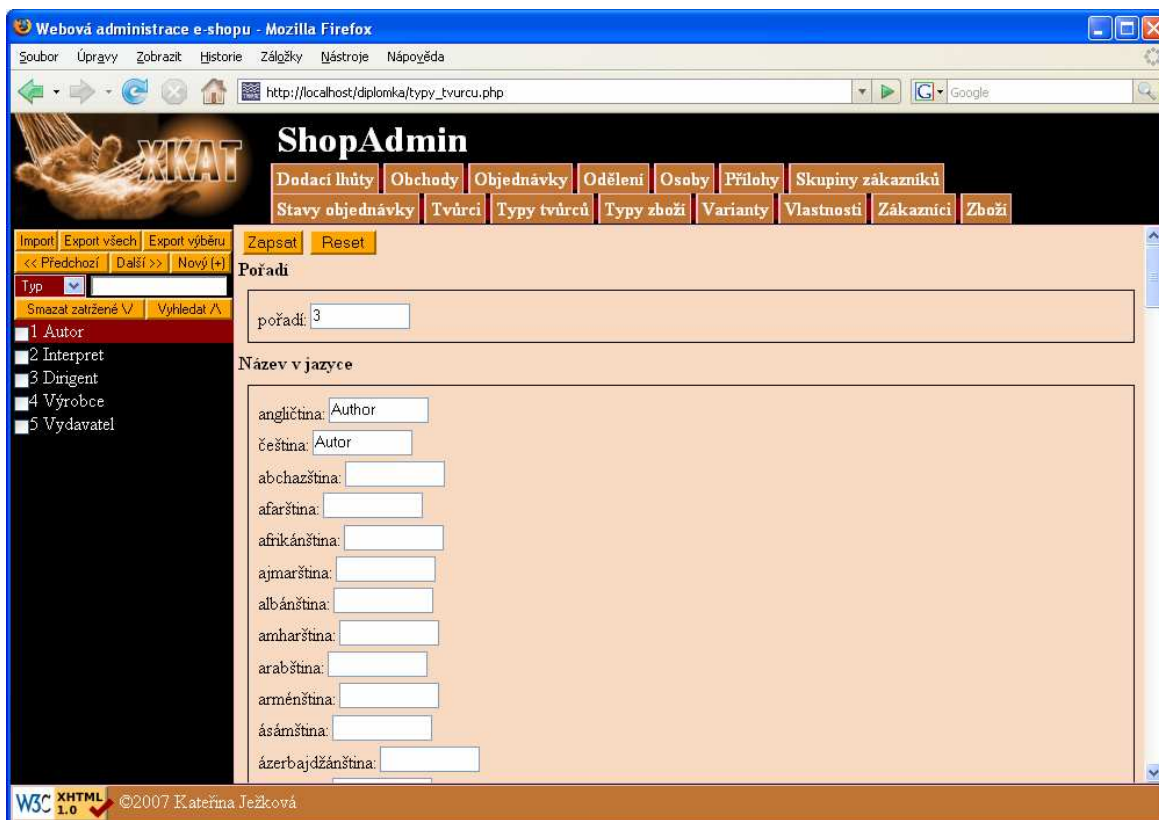
Strom	Entity	Atributy	Relace	Slovník	Indexy	Alternativní klíče
	Název	Parent entita	Child entita	Klíč		
1	Cený dodavatelů na zboží	Zboží dodavatele	Cený	Kód dodavatele,Osoba		
2	Defaultní jazyk obchodu	Jazyk	Obchod	Jazyk		
3	Defaultní měna obchodu	Měna	Obchod	Měna		
4	Dodací lhůta zboží dodavatele	Dodací lhůta	Zboží dodavatele	Dodací lhůta		
5	Dodavatel	Osoba	Zboží dodavatele	Osoba		
6	Dodavatel v objednávce	Osoba	Zboží objednávky	Osoba		
7	Jazyk názvu dodací lhůty	Jazyk	Název dodací lhůty	Jazyk		
8	Jazyk názvu stavu objednávky	Jazyk	Název stavu objednávky	Jazyk		
9	Jazyk názvu typu tvůrce	Jazyk	Název typu tvůrce	Jazyk		
10	Jazyk názvu varianty	Jazyk	Varianta	Jazyk		
11	Jazyk názvu vlastnosti	Jazyk	Vlastnost	Jazyk		
12	Jazyk textů ke zboží	Jazyk	Texty ke zboží	Jazyk		
13	Majitel	Osoba	Obchod	Osoba		
14	Měna dodavatelských cen	Měna	Cený	Měna		
15	Měna objednávky	Měna	Objednávka	Měna		
16	Název dodací lhůty	Dodací lhůta	Název dodací lhůty	Dodací lhůta		
17	Název stavu objednávky	Stav objednávky	Název stavu objednávky	Stav		
18	Název typu tvůrce	Typ tvůrce	Název typu tvůrce	Typ		
19	Objednávka zákazníka	Zákazník	Objednávka	Zákazník		
20	Objednávky v obchodě	Obchod	Objednávka	Obchod		
21	Oddělení obchodu	Obchod	Oddělení	Obchod		
22	Oprávnění uživatelů	Oprávnění uživatelů	Uživatel	Oprávnění		
23	Příloha zboží	Zboží	Příloha	Zboží		
24	Recyklační poplatek	Typ zboží	Recyklační poplatek	Typ		
25	Skupina zákazníka	Skupina	Zákazník	Skupina		
26	Stav objednávky	Stav objednávky	Objednávka	Stav		
27	Texty ke zboží	Zboží	Texty ke zboží	Zboží		
28	Titulkové oddělení	Oddělení	Titulka	Oddělení		
29	Tvůrci zboží (tvůrce)	Tvůrce	Tvůrci zboží	Tvůrce		
30	Tvůrci zboží (zboží)	Zboží	Tvůrci zboží	Zboží		
31	Typ přílohy	Typ Přílohy	Příloha	Typ		
32	Typ tvůrce	Typ tvůrce	Tvůrce	Typ		
33	Typ zboží	Typ zboží	Zboží	Typ		
34	Uživatel	Osoba	Uživatel	Osoba		
35	Varianty zboží (varianta)	Varianta	Varianty zboží	Varianta,Jazyk		
36	Varianty zboží (zboží)	Zboží	Varianty zboží	Zboží		
37	Vlastnosti zboží (vlastnost)	Vlastnost	Vlastnosti zboží	Vlastnost,Jazyk		
38	Vlastnosti zboží (zboží)	Zboží	Vlastnosti zboží	Zboží		
39	Zákazník	Osoba	Zákazník	Osoba		
40	Zboží dodavatele	Zboží	Zboží dodavatele	Zboží		
41	Zboží na titulce	Zboží	Titulka	Zboží		
42	Zboží objednávky (objednávka)	Objednávka	Zboží objednávky	Objednávka		
43	Zboží objednávky (zboží)	Zboží	Zboží objednávky	Zboží		
44	Zboží v oddělení (oddělení)	Oddělení	Zboží v oddělení	Oddělení		
45	Zboží v oddělení (zboží)	Zboží	Zboží v oddělení	Zboží		

Obr. 34. Relace

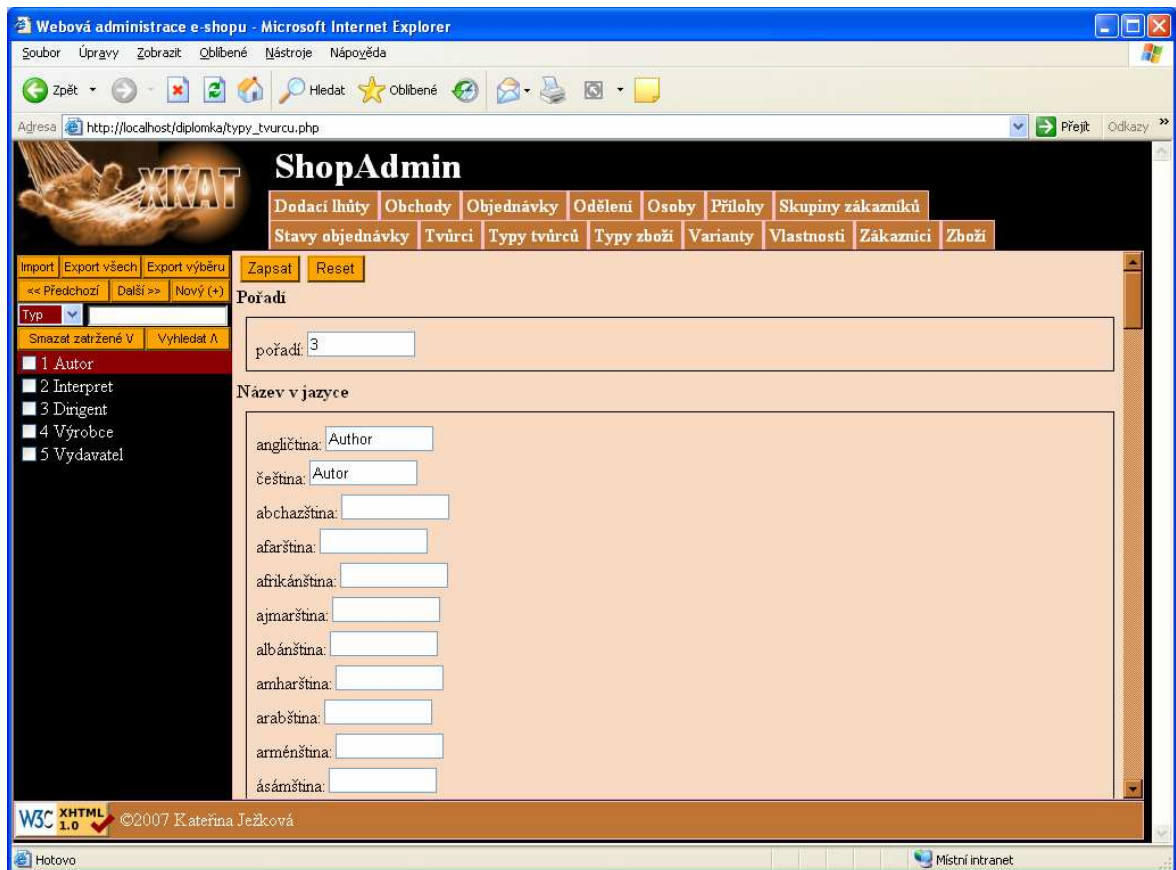
Význam jednotlivých relací není třeba popisovat, je patrný z popisu jednotlivých tabulek. Pro lepší přehled propojení tabulek slouží Příloha 2 této práce.

8 GRAFICKÝ NÁVRH A OVLÁDÁNÍ APLIKACE

Aplikace je vytvořena pomocí webového rozhraní. Byla otestována v prohlížečích Internet Explorer 6.0 (*Obr. 36*) a Mozilla Firefox 2.0 (*Obr. 35*) ve kterých se zobrazuje téměř totožně a funguje bez problémů. PHP skripty generují validní XHTML a to je dále upravováno pomocí kaskádových stylů. Pro některé speciality je použit Javascript. Ovládání je běžné jako u webových stránek a probíhá pomocí myši.



Obr. 35. Zobrazení v prohlížeči Mozilla Firefox



Obr. 36. Zobrazení v prohlížeči Internet Explorer

Grafický návrh byl proveden tak, aby byla aplikace dostatečně přehledná, působila příjemným dojmem a přitom aby grafika neprodložovala dobu načítání stránky. Základní rozdělení je vidět na obrázku (Obr. 37).



Obr. 37. Rozdělení okna aplikace

Rozdělení okna aplikace:

1. Horní menu
2. Ovládací prvky levého menu
3. Seznam položek levého menu
4. Detail
5. Zápatí
6. Logo

Logo jako jediné je tvořeno obrázkem. Zbytek grafiky je vytvořen pomocí kaskádových stylů. Kaskádové styly jsou definované v souboru *shop.css*. Tím pádem načítání grafiky nijak nebrzdí chod aplikace.

Levé menu je rozděleno na ovládací část a na část se seznamem položek. Celé levé menu je v podstatě jeden formulář a ovládací tlačítka jsou odesílací tlačítka s různými funkcemi. Přesná funkčnost tlačítek je popsána v uživatelské příručce. V seznamu je možné vidět 15 položek. Ostatní položky jsou na „dalších stránkách“, mezi kterými lze listovat pomocí tlačítek *Další* a *Předchozí*. Každá položka v seznamu navíc obsahuje zatrhávací políčko, které je využíváno při mazání nebo exportu položek do jiného formátu.

Horní menu slouží k přepínání mezi jednotlivými sekcemi. Po najetí na záložku sekce se písmo na záložce zbarví do žluta. V závislosti na sekci se mění obsah seznamu levého menu a struktura a obsah detailu.

Detail obsahuje formulář pro úpravu informací k položce. Na začátku se vždy zobrazí informace k první položce v seznamu dané sekce. Struktura formuláře se liší podle sekce. Všechny však obsahují tlačítka *Reset* a *Zapsat*.

9 PROGRAMOVÁ STRUKTURA

Vlastní program je napsán ve skriptovacím jazyce PHP, který generuje XHTML kód vytvářející vlastní webové rozhraní. V některých případech je použit i Javascript, například pokud uživatel vybere ve formuláři nějakou z hodnot otevře se nabídka související pouze s touto hodnotou. Javascriptu je obecně využito ke skrývání a zobrazování částí obrazovky.

9.1 Objektový návrh

Celá aplikace je naprogramovaná objektově. Je zde několik typů základních souborů. Jednak soubory s pomocnými třídami, což jsou *xml.inc.php* pro generování XML a XHTML kódu, *txt.inc.php* pro práci s textovými soubory a *db.inc.php* pro komunikaci s databázemi. Tyto třídy jsou využívány v definici tříd jednotlivých sekcí. Soubory obsahující definice tříd se jmenují *nazev_sekce.inc.php*. Každá sekce má jeden soubor *nazev_sekce.inc.php* a jeden soubor *nazev_sekce.php*. V souboru *nazev_sekce.php* se vytváří objekt a využívá se funkcí tříd ze souboru *nazev_sekce.inc.php*. Mezi tyto funkce patří například uložení dat, výpis, mazání apod.

Zde je ukázka funkce *zapis* ze souboru *typy_tvurcu.php*.

```

1 public function zapis($nazvy,$poradi){
2     include("init.php");
3     $db=new CDatabase($serverDB,$loginDB,$hesloDB,$nazevDB,$typDB);
4     if($this->typ==0){
5         echo "zapisu novy";
6         $db->zmenZaznamDB("INSERT INTO tvurcetyp
VALUES(null,$poradi)");
7         $zaznamy=$db->vyberZaznamDB("SELECT max(typ) as maximum FROM
tvurcetyp");
8         $max=$zaznamy[0]["maximum"];
9         while(list($jazyk, $nazev)=each($nazvy)){
10            if(strlen($nazev)>0){
11                $db->zmenZaznamDB("INSERT INTO tvurcetypnazev
VALUES($jazyk,$max,'$nazev')");
12            }
13        }
14    }
15    else{
16        //zmena
17        $zaznamy=$db->vyberZaznamDB("SELECT * FROM tvurcetypnazev
WHERE typ=$this->typ");

```

```
18         //pokud jazyk neexistuje zapise nový, pokud existuje a byl
        vyprazdnen musi ho smazat
19         while(list($jazyk, $nazev)=each($nazvy)){
20             if(strlen($nazev)>0){
21                 $zaznamy=$db->vyberZaznamDB("SELECT * FROM tvurcetypnazev
        WHERE typ=$this->typ AND jazyk=$jazyk" );
22                 if(Count($zaznamy)<1){
23                     // vloz nový nazev v tomto jazyce
24                     $db->zmenZaznamDB("INSERT INTO tvurcetypnazev
        VALUES($jazyk,$this->typ,'$nazev')");
25                 }
26                 else{
27                     $db->zmenZaznamDB("UPDATE tvurcetypnazev SET
        nazev='$nazev' WHERE jazyk=$jazyk AND typ=$this->typ");
28                 }
29             }
30             else{
31                 //zjistí zda prazdny nazev existoval v tabulce driv jako
        vyplneny, pokud ano, smaze ho
32                 $zaznamy=$db->vyberZaznamDB("SELECT * FROM tvurcetypnazev
        WHERE typ=$this->typ AND jazyk=$jazyk" );
33                 if(Count($zaznamy)>0){
34                     $db->zmenZaznamDB("DELETE FROM tvurcetypnazev WHERE
        jazyk=$jazyk AND typ=$this->typ");
35                 }
36             }
37         }
38         $db->zmenZaznamDB("UPDATE tvurcetyp SET poradi=$poradi");
39     }
40 }
41 }
```

9.2 Komunikace s databází

Komunikace s databází probíhá pomocí třídy *CDatabaze* uložené v souboru *db.inc.php*. Třída je navržena nejen pro práci s databází MySQL, ale i MSSQL, kterou je v případě potřeby možné také použít, ale prozatím na ní nebyla aplikace testována. Třída *CDatabaze* obsahuje funkce pro připojení, odpojení databáze, výběrové dotazy apod. Ostatní programový kód přistupuje k databázi právě pomocí funkcí této třídy. Pokud by tedy bylo potřeba použít aplikaci s jinou databází, pak stačí pouze upravit funkce této třídy, což je velkou výhodou.

Zde je ukázka funkce třídy *CDatabase* sloužící pro provedení výběrového dotazu.

```
42 public function vyberZaznamDB($dotaz){
43     $this->pripojDB();
44     $zaznamy=Array();
45     switch ($this->typ){
46         case "MSSQL":
47             if($vysledek=MSSQL_query($dotaz, $this->spojeni)){
48                 while($zaznam=MSSQL_Fetch_Array($vysledek)){
49                     $zaznamy[]=$zaznam;
50                 }
51             }
52             else{
53                 die("Chyba při provádění dotazu");
54             }
55             break;
56         case "MySQL":
57             if($vysledek=MySQL_query($dotaz, $this->spojeni)){
58                 while($zaznam=MySQL_Fetch_Array($vysledek)){
59                     $zaznamy[]=$zaznam;
60                 }
61             }
62             else{
63                 die("Chyba při provádění dotazu");
64             }
65             break;
66         default:
67             echo "Jiný nebo špatně definovaný typ databáze.";
68     }
69     $this->odpojDB();
70     return $zaznamy;
71 }
```

9.3 Generování XHTML

Generování XHTML probíhá pomocí jednak pomocí tříd uložených v souboru *xml.inc.php* a jednak pomocí načítání statického XHTML kódu. Aplikace je rozdělena do pěti částí.

Hlavička je uložena v souboru *hlavicka.php* a obsahuje výpis statického XHTML kódu, který obsahuje horní menu, definici funkcí Javascriptu a úvodní tag div levého menu.

Další částí je dynamicky se měnící obsah levého menu. Pak následuje *stred.php* ve kterém se nachází pouze ukončovací div tag levého menu a počáteční div detailu. Dále se dynamicky vypisuje detail a pak následuje statické zápatí uložené v souboru *paticka.php*.

Dynamické části se generují pomocí tříd pracujících pro určitou sekci. Každá sekce má minimálně dvě třídy a to na práci s položkou a jejími vlastnostmi a na práci se seznamem položek. O výpis seznamu i obsahů se stará třída pro práci se seznamem, která v sobě využívá funkce pro práci s jednotlivými položkami z třídy pro práci s položkou seznamu. Obě tyto třídy využívají funkcí tříd ze souboru *xml.inc.php*, které zajišťují výpis vlastního XHTML.

Třídy pro práci s položkami a seznamem jednotlivých sekcí jsou definované v souborech typu *nazev_sekce.inc.php*, vlastní zobrazení jednotlivých sekcí je seskládáno v souborech *nazev_sekce.php*. Tedy v horním menu jsou odkazy na soubory typu *nazev_sekce.php*, které využívají funkcí tříd ze souboru *nazev_sekce.inc.php* a načítají soubory *hlavicka.php*, *stred.php* a *paticka.php*.

Takže struktura programového kódu souboru *nazev_sekce.php* může vypadat zhruba takto:

```
72 include ("hlavicka.php");
73 include ("db.inc.php");
74 include ("xml.inc.php");
75 include ("nazev_sekce.inc.php");
76 $polozkysekce=new CSeznamPolozekVSekci();
77 funkce pro výpis seznamu, funkce pro zpracování akcí apod.
78 include ("stred.php");
79 funkce pro výpis obsahů, importního a exportního formuláře apod.
80 include("paticka.php");
```

Tento kód vygeneruje XHTML se seznamem všech zobrazitelných (maximálně 15) položek v levém menu a všech formulářů s obsahem k jednotlivým položkám, ovšem zobrazený bude pouze první formulář. Zobrazování dalších formulářů a změna barvy pozadí pod položkou které se týká aktuální formulář jsou řešeny pomocí Javascriptu.

10 EXPORT A IMPORT DAT

10.1 Export dat

K exportu slouží tlačítka *Export všech* a *Export výběru*, která s nacházejí mezi ovládacími prvky levého menu. Pokud uživatel zvolí *Export všech*, pak se vyexportují všechny vypsané položky v dané sekci (tedy buď úplně všechny nebo ty, které jsou omezeny filtrem), aniž by byla nějaká zatržená. Pokud chce uživatel exportovat jen část položek na které nelze použít filtr, musí tyto položky vybrat pomocí zatrhávacích políček a kliknout na tlačítko *Export výběru*.

Po stisknutí tlačítka *Export všech* nebo *Export výběru* se uživateli otevře exportní formulář. V tomto formuláři má možnost si vybrat, které vlastnosti položek bude chtít exportovat (Obr. 38). Například u osoby si vybere, že chce jméno, příjmení a město. Dále si uživatel může zvolit do jakého formátu bude chtít exportovat. Na výběr jsou dva – textový soubor a XML formát.

The screenshot shows a web form for exporting data. At the top, there are two buttons: 'Exportovat' (highlighted in orange) and 'Zpět'. Below the buttons is the section 'Formát exportu' with a dropdown menu currently set to 'Textový soubor'. The next section is 'Oddělovač' with a text input field containing a semicolon ';'. The final section is 'Exportovat vlastnosti' (Export properties), which contains a list of checkboxes for various attributes. The visible attributes are: 'typ', 'pořadí', and a series of 'název v jazyce' (name in language) for various languages including abchazština, afarština, afrikánština, ajmarština, albánština, amharština, angličtina, arabština, arménština, ásámština, and ázerbajdžánština.

Obr. 38. Formulář pro export

U textového souboru má uživatel možnost si vybrat oddělovač. Tato nabídka se objeví poté, co uživatel vybere, že chce exportovat do textového souboru. Pro většinu exportů postačí jako oddělovač středník, ale například pokud bude chtít uživatel exportovat

popisy zboží, například knih, může se stát, že se středník objeví i v tomto popisu a tím pádem by pak vyexportovaný soubor byl nepoužitelný pro další zpracování. Takovým opravdu bezpečným oddělovačem může být například `^` nebo nějaká „divočejší“ skupina znaků, například `@*@`.

Pro export se využívá tříd uložených v souborech `xml.inc.php` a `txt.inc.php`, které umožňují obecnou práci s textovými soubory a vytvoření XML výstupu. Tyto třídy dále využívá každá sekce ve svých třídách a koncový výstup přizpůsobuje typu svých položek.

Zde je ukázka třídy, kterou je možné vytvořit XML prvek.

```
81 class CXMLPrvek{
82     private $nazev;
83     private $obsah;
84     private $atributy=Array();
85
86     function __construct($nazev, $obsah, $atributy=Array()){
87         $this->nazev=$nazev;
88         $this->obsah=$obsah;
89         $this->atributy=$atributy;
90     }
91
92     public function vytvorXMLPrvek(){
93         $atributyRetez="";
94         if(Count($this->atributy)>0){
95             while(Current($this->atributy)){
96                 $atribut=Current($this->atributy);
97                 $atributyRetez.=" ".$atribut->vytvorXMLAtribut();
98                 Next($this->atributy);
99             }}
100         $prvek("<".$this->nazev;
101         if(StrLen($atributyRetez)!=0) $prvek.="&#x20;".$atributyRetez;
102         $prvek.="&#x20;>\n".$this->obsah."&#x20;&#x20;</".$this->nazev."&#x20;>\n";
103         return $prvek;
104     }
105 }
```


10.2 Import dat

Pro import je určeno tlačítko *Import* v ovládání levého menu. Po kliknutí na tlačítko se zobrazí formulář jako je na obázku (*Obr. 39*), ovšem pro každou sekci trochu jiný (s jinými možnostmi importu vlastností). Importovat lze pouze z textového souboru, který je možné vybrat pomocí pole *soubor*. Opět je možné nastavit oddělovač, podle toho, jaký je v textovém souboru obsažen.

Dalšími volbami je import jednotlivých vlastností. U každé vlastnosti lze nastavit, zda ji importovat či nikoliv. Pokud chce uživatel nějakou vlastnost importovat, vybere v jakém sloupci v textovém souboru se tato vlastnost nachází. Sloupce jsou číslovány od 1.

Textový soubor je zpracován ve třídě příslušné sekce, která používá funkce tříd uložených v souboru *txt.inc.php*. Funkce z těchto tříd nejdříve soubor zkontrolují a pak rozdělí na jednotlivé sloupce a položky. Vracené hodnoty již zpracovává třída příslušné sekce. Jakmile je soubor importován, nové položky se objeví v seznamu v levém menu a vypíše se hlášení o úspěšném importu. Při neúspěšném importu se vypíše hlášení o chybě.

The screenshot shows a web form for importing data. At the top, there are two buttons: 'Importovat' (Import) and 'Zpět' (Back). Below them is a section titled 'Textový soubor' (Text file) containing a text input field labeled 'soubor:' and a 'Procházet...' (Browse...) button. The next section is 'Oddělovač v textovém souboru' (Separator in text file) with a text input field labeled 'oddělovač:'. The final section is 'U vlastností, které chcete importovat vyberte ve kterém sloupci v textovém souboru se nacházejí.' (For properties you want to import, select in which column in the text file they are located). This section contains a list of properties, each with a dropdown menu. The first dropdown is labeled 'pořadí' (order) and is set to 'neimportovat' (do not import). The other dropdowns are labeled 'název v jazyce' (name in language) and are also set to 'neimportovat'. The languages listed are: abchazština, afarština, afrikánština, ajmarština, albánština, amharština, angličtina, arabština, arménština, and ásámština.

Obr. 39. Formulář pro import

11 UŽIVATELSKÁ PŘÍRUČKA

Uživatelská příručka je příloha PI této práce. Je rozdělena do čtyř částí.

První část se týká instalace. Zde jsou popsány požadavky na server po jejichž splnění je aplikace schopna běžet a také postup při jejím zprovoznění, to znamená vytvoření databáze a nastavení konfigurace. Také jsou zde zmíněny doporučené prohlížeče.

V druhé části příručky je popsáno webové rozhraní a ovládání aplikace. Okno aplikace je „rozebráno“ do několika částí jako jsou ovládání levého menu, seznam levého menu, horní menu apod. Je zde popsáno přepínání mezi jednotlivými sekcemi a mezi položkami sekcí. Velká pozornost je věnována ovládacím tlačítkům levého menu.

Třetí část je nejrozsáhlejší sekcí. Zabývá se popisem jednotlivých formulářů v oblasti detailu. Každá tato část je nazvána podle sekce ve které se formulář objevuje (Osoby, Dodací lhůty, Obchody ...). Jedná se o rozbor formuláře na části, v příručce označované jako vlastnosti, které jsou jednotlivě popsány. Nepopsané zůstávají pouze vlastnosti k nimž by popis byl zbytečný, jako jsou například jméno nebo příjmení. Kromě popisu jednotlivých vlastností příručka u některých „méně populárních“ vlastností radí, jak a k čemu je možné je využít. Také jsou uživatelé navigováni do jiných sekcí, kde můžou upravit položku na níž je odkaz v nějakém formuláři, ve kterém lze tuto položku pouze vybrat ale ne upravovat.

Poslední částí příručky je kapitola o exportu a importu dat. Zde jsou uživatelé poučeni, kterými tlačítky se export a import provádí a jaké formuláře je při těchto akcích čekají. Také je zde ukázáno jak široké či úzké jsou možnosti exportu a importu a co vše a jak si mohou uživatelé nastavit. Dále je zde pojednání o používání oddělovačů v textových souborech a rady jaký oddělovač a proč použít.

Příručka obsahuje i obrázky, hlavně v části o webovém rozhraní. Je svázána kroužkovou vazbou s pevnější přední a zadní částí a tištěná pouze jednostranně, což umožňuje uživateli snadné listování a zaručuje „delší trvanlivost“. Pro zpřehlednění je v záhlaví každé stránky název kapitoly. Na začátku je obsah, což je také podpora pro rychlé vyhledání potřebné informace.

ZÁVĚR

Aplikace byla vytvořena pomocí PHP skriptů a databáze MySQL. Programový kód je přehledný a snadno rozšiřitelný, protože PHP skripty jsou naprogramovány objektově. Součástí aplikace je pěkné webové rozhraní, které není graficky náročné a je snadno ovladatelné. Toto rozhraní je tvořeno validním XHTML kódem, který je generován pomocí PHP skriptů. Pro drobná vylepšení je použit i Javascript a CSS pro grafické úpravy. Uživatel k aplikaci přistupuje přes webový prohlížeč. Data jsou rozdělena do několika sekcí a editována pomocí webových formulářů. Uživatelé mohou exportovat a importovat data. Exportovat mohou do formátu XML nebo do textového souboru s různými oddělovači. Importovat je možné pouze z textového souboru, ale opět s libovolným oddělovačem. Aplikaci provede uživatele uživatelská příručka.

Výhodou aplikace je její nezávislost na operačním systému a typu serveru. Stačí pouze funkční PHP 5 a databáze MySQL, také verze 5. Drobnými úpravami by bylo možné dosáhnout i nezávislosti na databázi. Také formát importovaných a exportovaných dat není závislý na platformě. Aplikace byla otestována na dvou prohlížečích, a to Internet Explorer 6.0 a Mozilla Firefox 2.0. Uživatel má možnost si vybrat. Zobrazení v obou prohlížečích je téměř totožné.

Nevýhodou aplikace je složitá instalace, která je ale popsána v uživatelské příručce. Také rychlost je vhodná pouze pro malé množství dat, s větším množstvím dat by bylo třeba udělat optimalizaci aplikace. Prozatím může aplikace spolehlivě fungovat pro malý obchod, pro větší obchod by bylo třeba udělat ještě pár úprav.

Aby mohla aplikace fungovat i pro větší a velký obchod, pak by bylo nutné ji zabezpečit. V návrhu databáze je s přihlašování uživatelů již počítáno. Stačí pouze přizpůsobit programový kód.

Dále by bylo třeba optimalizovat rychlost načítání dat, například vytvořením „tempovacích“ tabulek v databázi. To by nijak neovlivnilo základní strukturu databáze, jen by došlo k jejímu rozšíření.

Pro export a import většího množství dat by pak bylo dobré vytvořit dávkové zpracování, protože na většině serverů je doba běhu PHP skriptu omezena na 30 sekund a tento čas není dostatečný ke zpracování většího množství dat.

Aplikaci je možné dále vylepšovat. Jedním z takových vylepšení by mohlo být například vzájemné propojení sekcí přes jejich formuláře v detailu. Tím se rozumí například pokud je někde odkaz na osobu, pak by bylo možné pomocí speciálního tlačítka přejít přímo do sekce Osoby a editovat údaje o této osobě.

ZÁVĚR V ANGLIČTINĚ

The application was created with the use of PHP scripts and MySQL database. Program code is transparent and easily extensible, because PHP scripts are object programmed. Nice web interface, which isn't graphically difficult and is easily manageable, is a part of the application. This interface is formed with the valid XHTML code which is generated with the use of PHP scripts. There is Javascript and CSS for graphical form used for minor improvements. The user approaches the application through the web browser. Data are divided into several sections and edited with the use of web forms. Users can export and import data. They can export data into the XML format or into a text file with varied separators. User's guide leads user through the application.

The advantage of the application is its independence from an operating system and a type of server. It's enough only functional PHP 5 and MySQL database, also the version 5. It would be able to reach database independence in terms of little modifications. The format of imported and exported data is also platform independent. The application is tested for two browsers, namely Internet Explorer 6.0 and Mozilla Firefox 2.0. The user has a possibility to choose. The display in both browsers is almost identical.

The disadvantage is its complicated installation, nevertheless, it is described in the user's guide. Further the speed is suitable only for small amount of data, with more data it would be necessary to do the application tuning. For the present the application can function for a minor firm. For a major firm or an emporium it would be required to do a few more modifications.

It would be necessary to make safe the application to be able to work for a major firm or an emporium. It is bargained for users logging in the database project. It only suffices to adapt the program code.

Consequently it would be needful to tune a data counting speed, for example with the creating "temporary" tables in the database. It wouldn't influence the base structure of the database, it would only expand it.

Then it would be suitable to create batch process for more data export and import, because on most of the servers the running time of PHP script is limited up to 30 seconds and this time isn't enough for more data processing.

It is possible to next upgrade the application. One of these upgrades it would be for example interconnection sections through their forms in detail. It means for example if anywhere is refer to a man, than it would be possible with use of special button cross directly into the section Osoba and edit this man data.

SEZNAM POUŽITÉ LITERATURY

- [1] ZELDMAN, J. *Tvorba webů podle standardů*: Computer Press, Brno 2004. ISBN:80-251-0347-1.
- [2] GUTMANS, A., BAKKEN, S. S., RETHANS, D. *Mistrovství v PHP 5*: CP Books, a. s., Brno 2004. ISBN:80-251-0779-X.
- [3] GROFF, J., R., WEINBERG, P. N. *SQL – Kompletní průvodce*: CP Books, a. s., Brno 2005. ISBN:80-251-0369-2.
- [4] TEAGUE, J., C. *DHTML a CSS pro WWW*: Softpress, Praha 2005. ISBN:80-86479-77-1.
- [5] YOUNG, J., M. *XML krok za krokem*: Computer Press, a. s., Brno 2006. ISBN:80-251-1070-2.
- [6] *PHP: Hypertext Preprocessor* [online]. [cit.2007-05-11]. Dostupný z WWW: <<http://www.php.net>>
- [7] *MySQL AB :: The world's most popular open source database* [online]. [cit.2007-05-11]. Dostupný z WWW: <<http://www.mysql.com>>
- [8] *Welcome! The Apache Software Foundation* [online]. [cit.2007-05-11]. Dostupný z WWW: <<http://www.apache.org>>
- [9] *ADBlog Jak na databáze I – text spotu* [online]. [cit.2007-05-20]. Dostupný z WWW: <http://adobrovolny.net/uvod/124/jak_na_databaze_I>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

XML	Extensible Markup Language, rozšiřitelný značkovací jazyk
HTML	HyperText Markup Language
XHTML	Extensible HyperText Markup Language
DTD	Data Type Definition
DOM	Document Object Model
XSLT	Extensible Stylesheet Language Transformation
PHP	Hypertext Preprocessor
CSS	Cascading Style Sheet
SQL	Structure Query Language

SEZNAM OBRÁZKŮ

<i>Tabulka Osoba</i>	21
<i>Tabulka Obchod</i>	22
<i>Tabulka Oddělení</i>	23
<i>Tabulka Titulka</i>	23
<i>Tabulka Zboží v oddělení</i>	24
<i>Tabulka Zboží</i>	25
<i>Tabulka Texty ke zboží</i>	25
<i>Tabulka Typ zboží</i>	26
<i>Tabulka Recyklační poplatek</i>	26
<i>Tabulka Zboží dodavatele</i>	27
<i>Tabulka Dodací lhůta</i>	27
<i>Tabulka Název dodací lhůty</i>	28
<i>Tabulka Ceny</i>	29
<i>Tabulka Měna</i>	29
<i>Tabulka Tvůrce zboží</i>	30
<i>Tabulka Tvůrce</i>	30
<i>Tabulka Typ tvůrce</i>	31
<i>Tabulka Název typu tvůrce</i>	31
<i>Tabulka Jazyk</i>	31
<i>Tabulka Vlastnosti zboží</i>	32
<i>Tabulka Vlastnost</i>	32
<i>Tabulka Varianty zboží</i>	33
<i>Tabulka Varianta</i>	33
<i>Tabulka Příloha</i>	34
<i>Tabulka Typ přílohy</i>	34
<i>Tabulka Objednávka</i>	35
<i>Tabulka Zboží objednávky</i>	35
<i>Tabulka Stav objednávky</i>	36
<i>Tabulka Název stavu objednávky</i>	36
<i>Tabulka Zákazník</i>	36
<i>Tabulka Skupina</i>	37

<i>Tabulka Uživatel</i>	37
<i>Tabulka Oprávnění uživatelů</i>	38
<i>Relace</i>	38
<i>Zobrazení v prohlížeči Mozilla Firefox</i>	39
<i>Zobrazení v prohlížeči Internet Explorer</i>	40
<i>Rozdělení okna aplikace</i>	41
<i>Formulář pro export</i>	47
<i>Formulář pro import</i>	49

SEZNAM PŘÍLOH

P I Příloha 1 Uživatelská příručka

P II Příloha 2 Návrh databáze