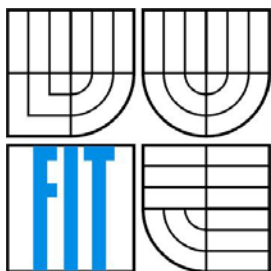




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INFORMATION SYSTEMS

POKROČILÁ WEBOVÁ APLIKACE  
ELEKTRONICKÉHO OBCHODU  
WEB APPLICATION OF E-SHOP

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

Tomáš Katolický

VEDOUCÍ PRÁCE  
SUPERVISOR

Lukáš Roman, Ing., Ph.D.

BRNO 2007

## Zadání bakalářské práce

Řešitel: **Katolický Tomáš**

Obor: Informační technologie

Téma: **Pokročilá webová aplikace elektronického obchodu**

Kategorie: Elektronický obchod

Pokyny:

1. Seznamte se s jazyky a prostředky pro tvorbu webových informačních systémů (XHTML, CSS, PHP, Javascript, MySQL).
2. Seznamte se s požadavky kladenými na elektronický obchod jisté firmy. Rozsah systému konzultujte s vedoucím BP. Požadavky podrobně analyzujte. Při návrhu se také zaměřte na bezpečnost systému. Systém musí umožňovat tisk faktury za zboží objednané zákazníkem. Systém musí také umět znázorňovat graficky statistiky z prodeje výrobků. Dále systém bude obsahovat nějakou jednoduchou klasifikační nebo predikční metodu. Výběr této klasifikace nebo predikce konzultujte s vedoucím BP.
3. Proveďte návrh systému.
4. Daný systém implementujte tak, aby byl použitelný v praxi pro danou firmu.
5. Zhodnoťte dosažené výsledky, porovnejte váš systém s existujícími systémy a navrhnete další možné rozšíření do budoucna.

Literatura:

- Kosek, J.: PHP - Tvorba interaktivních internetových aplikací
- Maslakowski, M.: Naučte se MySQL za 21 dní
- Písek, S.: JavaScript - efektní nástroj oživení WWW stránek

Při obhajobě semestrální části projektu je požadováno:

- Body 1), 2) a 3)

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese <http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Lukáš Roman, Ing., Ph.D.**, UIFS FIT VUT

Datum zadání: 1. listopadu 2006

Datum odevzdání: 15. května 2007

**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
Fakulta informačních technologií  
Ústav informačních systémů  
602 00 Brno, Božetěchova 2



---

doc. Ing. Jaroslav Zendulka, CSc.  
vedoucí ústavu

# LICENČNÍ SMLOUVA POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

## 1. Pan

Jméno a příjmení: **Tomáš Katolický**  
Id studenta: 43037  
Bytem: Alešova 17, 591 01 Žďár nad Sázavou  
Narozen: 02. 10. 1978, Třebíč  
(dále jen "autor")

a

## 2. Vysoké učení technické v Brně

Fakulta informačních technologií  
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305  
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....  
(dále jen "nabyvatel")

## Článek 1

### Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):  
bakalářská práce

Název VŠKP: Pokročilá webová aplikace elektronického obchodu  
Vedoucí/školitel VŠKP: Lukáš Roman, Ing., Ph.D.  
Ústav: Ústav informačních systémů  
Datum obhajoby VŠKP: .....

VŠKP odevzdal autor nabyvateli v:

tištěné formě	počet exemplářů: 1
elektronické formě	počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

## Článek 2 Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
  - ihned po uzavření této smlouvy
  - 1 rok po uzavření této smlouvy
  - 3 roky po uzavření této smlouvy
  - 5 let po uzavření této smlouvy
  - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

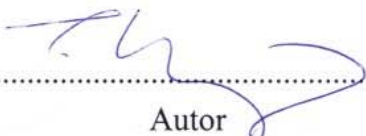
## Článek 3 Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: 3.5.2007 .....

.....

Nabyvatel

  
.....  
Autor

## **Abstrakt**

Práce se zabývá problematikou elektronického obchodu. Cílem bylo vytvořit na míru obchod pro firmu zabývající se prodejem vybavení koupelen, keramických obkladů a dlažeb, a věcí souvisejících s tímto sortimentem zboží. Systém byl implementován s využitím technologií PHP, CSS, JavaScript, MySQL, Apache.

## **Klíčová slova**

Elektronický obchod, informační systém, e-shop, XHTML, CSS, Javascript, PHP, MySQL

## **Abstract**

Aim of this essay was to learn programming languages and available tools for creation of information systems. Main part is focused on analyse, design and implementation of the Web Application of E-shop.

## **Keywords**

Information system, e-shop, XHTML, CSS, Javascript, PHP, MySQL

## **Citace**

Tomáš Katolický: Pokročilá webová aplikace elektronického obchodu, bakalářská práce, Brno, FIT VUT v Brně, 2007

# **Pokročilá webová aplikace elektronického obchodu**

## **Prohlášení**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Lukáše Romana, Ph.D. Další informace jsem čerpal ze zkušeností nabytých ve firmě TEKO, spol. s r.o., Brněnská 30, Žďár nad Sázavou, kde jsem nějakou dobu pracoval.

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

.....  
Tomáš Katolický  
10.5.2007

© Tomáš Katolický, 2007.

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..*

# Obsah

Obsah .....	1
Úvod .....	3
1 Specifikace požadavků .....	4
1.1 Rozvržení stránky .....	4
1.2 Nákupní košík .....	5
1.3 Výpis zboží .....	5
1.4 Detail zboží .....	5
1.5 Obrázky zboží .....	6
1.6 Objednávka zboží .....	6
1.7 Administrační rozhraní .....	6
1.7.1 Kategorie zboží .....	7
1.7.2 Zboží .....	7
1.7.3 Objednávky .....	7
1.7.4 Zákazníci .....	7
2 Využité technologie .....	8
2.1 Internet .....	8
2.1.1 HTTP .....	8
2.1.2 WWW .....	9
2.1.3 URL .....	9
2.2 Použité nástroje .....	9
2.2.1 WWW server .....	9
2.2.2 HTML .....	10
2.2.3 CSS .....	10
2.2.4 PHP .....	11
2.2.5 Javascript .....	11
2.2.6 SQL .....	12
3 Predikce .....	13
3.1 Metoda nejmenších čtverců .....	13
3.2 Lagrangeův interpolační polynom .....	15
3.3 Predikce vývoje cen .....	16
3.3.1 Způsob výběru metody .....	16
3.3.2 Presentace předpovědi cen .....	17
4 Analýza řešení .....	18
4.1 Role .....	18

4.1.1	Způsob zabezpečení systému .....	18
4.1.2	UML.....	19
4.2	Use Case Model .....	19
4.3	ER diagram.....	21
5	Implementace .....	22
5.1	Uživatelské prostředí.....	22
5.1.1	Přihlašování .....	22
5.1.2	Procházení zboží .....	22
5.2	Administrátorská část.....	23
5.2.1	Objednávky .....	23
5.2.2	Editace kategorií .....	23
5.2.3	Editace Zboží .....	23
5.2.4	Vkládání obrázků.....	23
5.3	Využití knihovny .....	25
5.3.1	libchart .....	25
5.3.2	AJAX Star Rating Bar .....	26
5.3.3	dTree .....	26
6	Porovnání systému s již existujícími.....	27
7	Možnosti rozšíření .....	28
8	Závěr .....	29
	Literatura .....	30
	Seznam příloh.....	31



# Úvod

V dnešní době je v jakémkoliv druhu podnikání, nabízející nějaké zboží či služby, nutné aby firma měla co nejvíce možností jak tyto produkty nabídnout zákazníkovi. Konkurence bývá vysoká, a proto bývá dalším krokem firem snaha zřídít si kromě klasického kamenného obchodu i obchod elektronický.

V první kapitole se věnuje možnostem elektronického obchodování a specifikaci požadavků na elektronický obchod pro naši firmu. Jedná se o firmu, která poskytuje zboží a služby v širším záběru, od prodeje instalátérského zboží, až po kompletní nabídku vybavení koupelen. Do analýzy požadavků patří obecná definice toho, co by systém měl umožňovat. Dále definice uživatelského rozhraní, tedy přístupu k systému, jakým způsobem bude možno systém využívat. Dále rozebereme jednotlivá řešení jak pro obchodníka, tak i s ohledem na požadavky zákazníků.

Druhá kapitola se věnuje popisu použitých implementačních nástrojů, PHP, MySQL, CSS, JavaScript, Apache. Je zde shrnuta jejich historie, výčet jejich vlastností a možností.

Třetí kapitola se zabývá popisem použitých metod pro predikci budoucí ceny zboží, způsobem jejího začlenění do aplikace, vlastním řešením a zkušenostmi s aplikací těchto metod.

Ve čtvrté kapitole se zabýváme analýzou řešení projektu. Naleznete zde návrh na rozdělení rolí a způsob zabezpečení. Součástí analýzy jsou i potřebné modely navrhovaného systému namodelované pomocí jazyka UML.

Pátá kapitola se zabývá implementací navrhovaného systému. Dále pak uživatelským prostředím, vytvořeném pro tento systém na základě analýzy požadavků firmy.

Následuje porovnání s obdobnými aplikacemi, které jsou provozovány na Internetu. Dále je uvedeno několik možností dalšího rozvoje systému, díky kterým by systém byl použitelnější, a pomohl firmě udržet si stále zákazníky.

Poslední kapitola uzavírá řešený problém, sumarizuje dosažené výsledky a zkušenosti získané při vývoji.

# 1 Specifikace požadavků

Existuje řada způsobů jak začít prodávat po internetu. Asi nejjednodušší je vystavení seznamu produktů, které prodáváme. Obvykle pak stačí uvést cenu a nějaký identifikátor (např. číslo zboží), a vybídnout případného zájemce o koupi, aby nás kontaktoval telefonem, e-mailem, či jiným vhodným způsobem.

Na druhém konci pak stojí řešení ušité na míru našim potřebám, například přímé napojení internetového obchodu na skladový nebo účetní software firmy. Já se zde budu zabývat řešením na míru, tedy jedním z pokročilejších řešení.

## 1.1 Rozvržení stránky

Barvy či obrázky záleží čistě na vkusu majitele nebo obvykle kopírují barvy firmy. Podstatné je ale rozvržení na stránce. Zákazník očekává přehledné menu vlevo na stránce, kde jednotlivé položky odpovídají kategoriím sortimentu zboží, které jsou nabízeny. Kategorie by měli obsahovat případné podkategorie, a ty zas další podkategorie. Z hlediska orientace zákazníka v sortimentu zboží pak není doporučeno používat více než 3 zanoření, ale systém takhle omezen být samozřejmě nemusí.

Uprostřed stránky by měly být vhodně rozvržené produkty odpovídající dané kategorii, seřazené dle názvu nebo ceny. Dosti důležitá je úvodní stránka na které by mělo být zboží v akci, ve slevě, nejprodávanější, nejnavštěvovanější... atd. Je však dobré se držet hesla, že všeho moc škodí.

Někde vpravo, obvykle v horním rohu stránky zákazník očekává případný formulář pro přihlášení nebo odhlášení. Pod ním by mělo být zboží v košíku, jeho množství, uložil-li tam nějaké, s celkovou cenou nákupu. Dále ještě v pravé části bývá ještě nabídka například 5-10 nejprodávanějších položek ať už v dané kategorii, v případě, že je nějaká vybraná, nebo celkově.

Někde by měl být také vhodně umístěn formulář pro vyhledávání v obchodě, nebo odkaz na něj. Sice takto vypadá spousta internetových obchodů, a možná někoho napadne udělat něco originálnějšího, ale zákazníci jsou na toto rozvržení zvyklí a cokoliv, co by bylo jinak, u nich nezbuzuje důvěru. Důvěra a serióznost jsou důležitými vlastnostmi firmy. Zákazník se pak neobává svěřit své peníze firmě.

## 1.2 Nákupní košík

Při prohlížení sortimentu nabízeného zboží zákazník očekává, že v kterémkoliv okamžiku půjde uložit dané zboží do košíku. Vhodný je symbol nákupního košíku. Po kliknutí na takovýto obrázek očekává zákazník upozornění, že došlo k nějaké události. Toto upozornění nesmí být rušivé jako například přesměrování na novou stránku s obsahem košíku, tím by se zákazník snadno našel pokud hodlá nakupovat více položek. Obvykle je vhodné toto řešit speciálně vyhrazeným místem na stránce spolu s celkovou cenou a počtem položek v košíku. V těchto místech dále zákazník očekává odkaz na stránku, kde se mu zobrazí podrobný výpis položek v košíku a bude zde moci změnit počet kusů nebo položku odstranit úplně. Po každé takové změně je samozřejmost znovu přepočítat celkovou cenu nákupu.

## 1.3 Výpis zboží

Může být prováděn mnoha způsoby. Opět záleží na typu prodáváného zboží. Myslím, že nejvhodnější je „kartový“ výpis (viz. Obrázek 1.3) s možností změny na čistě řádkový bez obrázku. Ať už je však výpis realizován jakkoliv, měl by být především přehledný, zobrazovat podstatné informace, a především musí být seřaditelný. Podle ceny, výrobce, oblíbenosti... apod. Je-li položek příliš mnoho, stránkují se. Uživatel by měl mít sám možnost si změnit, kolik položek chce vypsát na stránku. Samotné stránkování by mělo být realizováno čísly stránek spolu s ikonami (nebo textem) „Další“ a „Předchozí“.

	<b>Riho LECCO</b> dél. 180cm, šíř. 130cm, výš. 52cm Lecco se svým výrazným futuristickým designem zaujme pozornost v každé koupelně. Jednoduchá, krásná a navíc s velkým odkládacím...
<b>11 246 Kč</b> s DPH	<a href="#">masážní</a> <a href="#">&gt; detail</a>
	<b>Riho MERKUR</b> dél. 135cm, šíř. 135cm, výš. 46cm Merkur je vana s naprosto originálním designem - je anatomicky tvarovaná a má prakticky umístěný odtok, který Vám umožní vykoupat se...
<b>8 676 Kč</b> s DPH	<a href="#">masážní</a> <a href="#">&gt; detail</a>

Obrázek 1.3: Kartový výpis zboží

## 1.4 Detail zboží

Podrobný výpis produktu by měl být opravdu podrobný, vypisujeme tedy všechny informace, které můžete o produktu poskytnout. Čím více informací, tím lépe. Samozřejmě podané informace musí být srozumitelné. Míra srozumitelnosti se opět odhaduje v závislosti na prodáváném zboží. V detailu zboží je dobré umístit tlačítko, které vrátí uživatele zpět na výpis zboží.

Jako další rozšíření zboží by bylo vhodné implementovat například graf vývoje cen, hodnocení oblíbenosti, případně diskusní příspěvky. Já jsem si kromě toho ještě zvolil predikci cen zboží, které budu zvláště věnovat celou kapitolu mé práce.

## 1.5 Obrázky zboží

Fotografiím zboží je třeba věnovat maximální pozornost. Obrázky totiž naše zboží skutečně prodávají. Do detailu zboží je nejlepší umístit tolik obrázků, kolik jen je možno. Zákazník si potřebuje na zboží sáhnout, prohlédnout ze všech úhlů, potěžkat, vyzkoušet, a jelikož není ve fyzickém obchodě jsou jedinou možností fotografie. Ty by měly být co nejkvalitnější, dostatečně velké a nejlépe přizpůsobené celkové grafice vašeho obchodu. Aby zboží na fotografii nepůsobilo zkresleně je vhodné k němu umístit nějakou velikostně známou věc pro porovnání. Zákazník pak má možnost si představit, jak asi vypadá věc ve skutečnosti, v porovnání s jemu známou věcí. V některých případech dostačuje textový popis rozměrů, ale spíše jen u tvarově jednoduchého zboží.

## 1.6 Objednávka zboží

Ve většině případů stačí velmi jednoduchý formulář s kontaktními údaji. Měl by být co nejjednodušší, není dobré nutit vyplňovat uživatele údaje, které bezpodmínečně k objednávce nepotřebujeme. Naopak u údajů, které jsou pro nás důležité je vhodné umístit nápovědu s příklady vyplnění. Předjdeme tak zbytečným problémům.

Po automatické kontrole správnosti objednávky se očekává, že zákazník bude upozorněn na úspěšné odeslání objednávky, a její kopie mu dojde na e-mail. V případě, že zákazník uvažuje o tom, že by u nás mohl nakupovat i v budoucnu, tak se může zaregistrovat, systém mu přidělí uživatelské jméno a heslo, pod kterým se může do obchodu v budoucnu přihlásit, a nakupovat již bez vyplňování svých kontaktních údajů. Dále by bylo vhodné, aby zákazník mohl zjistit stav své objednávky, případně historii objednávek minulých.

## 1.7 Administrační rozhraní

Existují asi dva nejrozšířenější způsoby administrace elektronického obchodu. Pomocí aplikace nainstalované na našem počítači, nebo přímo v internetovém prohlížeči. Zadáte do prohlížeče adresu svého obchodu, jste požádáni o uživatelské jméno a heslo, a po přihlášení se již nacházíte v administračním rozhraní.

Dále existují různé možnosti napojení na skladový a účetní systém firmy, nebo manuální, či automatické exporty dat z těchto systémů. Pro naši aplikaci budeme používat administrační rozhraní přímo v internetovém prohlížeči.

### **1.7.1 Kategorie zboží**

Je důležité řádně spravovat kategorie a podkategorie zboží. Přidávat, upravovat, odstraňovat. V případě, že by se někdo snažil smazat kategorii v které je nějaké zboží, musí být uživatel na tuto skutečnost důrazně upozorněn, případně by to bez vymazání, nebo přesunutí zboží do jiné kategorie vůbec nemělo jít smazat.

### **1.7.2 Zboží**

Jako nejjednodušší minimum pro správu zboží se považuje úprava atributů jako je název, popis, cena a obrázek. U obrázků by se automaticky měly vytvořit jejich náhledy a případně upravit jejich rozměry. Není vůbec špatné, máme-li možnost zadat související zboží. Další užitečnou funkcí je například možnost zneaktivnit/zaktivnit zboží.

Správce musí mít možnost kartu zboží libovolně upravovat, ale program si musí pohlídat, aby tyto úpravy nenarušily integritu databáze (např. vymazání zboží, které je v objednávce).

### **1.7.3 Objednávky**

Nejjednodušší řešení objednávek bývá seznam zboží, který je odeslán na specifickou e-mailovou adresu. Jakmile e-mail přijmeme, další postup záleží na čistě na organizaci práce v naší firmě. Lepším řešením bývá upozornění na novou objednávku, ale její vyřízení se pak zpracuje přes administrační rozhraní obchodu, kde s objednávkou můžeme dále pracovat. Zobrazit si její obsah, nastavit ve které fázi se právě nachází (přijata, zpracována, objednáno, expedována), o tomto může být pak informován i zákazník. Další postup obvykle záleží na potřebách konkrétní firmy.

### **1.7.4 Zákazníci**

Kromě klasické úpravy a kontroly vyplnění registračního formuláře, je vhodné mít možnost stanovit pro každého zákazníka individuální slevy. Tyto slevy by mělo jít aplikovat na každou skupinu zboží individuálně, tak jak je to běžné v obchodních firmách.

## 2 Využité technologie

V této části jsou vysvětlené pojmy a technologie, použité, nebo úzce související s implementací mé práce. Je zde shrnuta jejich historie, výčet jejich vlastností a možností.

### 2.1 Internet

Internet je celosvětová počítačová síť, která spojuje jednotlivé menší sítě, pomocí sady protokolů IP. Za datum jeho vzniku by se mohl považovat rok 1969 kdy vznikla síť pojmenovaná ARPANET, která propojovala americké univerzity. V současné době je k této síti připojena více jak miliarda uživatelů. Další informace můžete čerpat z [1]. Musíme si tedy uvědomit, že k našemu obchodu bude mít přístup každý a proto při návrhu aplikace nepodcenit bezpečnostní hledisko.

#### 2.1.1 HTTP

**HTTP** (Hyper Text Transfer Protocol) je internetový protokol určený původně pro výměnu hypertextových dokumentů ve formátu HTML. Tento protokol je spolu s elektronickou poštou tím nejvíce používaným, a zasloužil se o obrovský rozmach internetu v posledních letech.

Protokol funguje způsobem dotaz - odpověď. Uživatel (pomocí programu, obvykle internetového prohlížeče) pošle serveru dotaz ve formě čistého textu, obsahujícího označení požadovaného dokumentu, informace o schopnostech prohlížeče apod. Server poté odpoví pomocí několika řádků textu popisujících výsledek dotazu (zda se dokument podařilo najít, jakého typu dokument je atd.), za kterými následují data samotného požadovaného dokumentu.

Pokud uživatel bude mít po chvíli další dotaz na stejný server (např. proto, že uživatel v dokumentu kliknul na hypertextový odkaz), bude se jednat o další, nezávislý dotaz a odpověď. Z hlediska serveru nelze poznat, jestli tento druhý dotaz jakkoli souvisí s předchozím. Kvůli této vlastnosti se protokolu HTTP říká *bezstavový protokol* – protokol neumí uchovávat stav komunikace, dotazy spolu nemají souvislost. Tato vlastnost je nepříjemná pro implementaci složitějších procesů přes HTTP (např. právě internetový obchod potřebuje uchovávat informaci o identitě zákazníka, o obsahu jeho „nákupního košíku“ apod.). K tomuto účelu byl protokol HTTP rozšířen o tzv. HTTP cookies, které umožňují serveru uchovávat si informace o stavu spojení na počítači uživatele. Více viz [2].

## 2.1.2 WWW

World Wide Web neboli zkráceně WWW znamená ve volném překladu „Celosvětová pavučina“. Je to tedy označení pro aplikace internetového protokolu HTTP, čímž je myšlena soustava propojených hypertextových dokumentů. Dokumenty umístěné na počítačových serverech jsou adresovány pomocí URL, jehož součástí je i doména a jméno počítače. Název naprosté většiny těchto serverů začíná zkratkou www, i když je možné používat libovolné jméno vyhovující pravidlům URL. Více informací možno nalézt na [3].

## 2.1.3 URL

URL je zkratka anglického výrazu Uniform Resource Locator. Je to řetězec znaků s definovanou strukturou a slouží k přesné specifikaci umístění zdrojů informací (ve smyslu dokument nebo služba) na Internetu. URL definuje doménovou adresu serveru, umístění zdroje na serveru a protokol, kterým je možné zdroj zpřístupnit. Více viz [4].

Jednotlivá pole v URL: *protokol, doménové jméno, port, specifikace souboru, parametry*

Příklad URL adresy: `http://cs.wikipedia.org:80/w/wiki.phtml?title=URL&action=edit`

## 2.2 Použité nástroje

V této kapitole jsou popsány nástroje, použité pro vytvoření a běh mé aplikace. Využití externích knihoven a technologie. Ve zkratce lze říci, že základem mé aplikace je programovací jazyk PHP, SQL databáze a www sever, na kterém to všechno běží.

### 2.2.1 WWW server

Základním kamenem funkčnosti mé aplikace je WWW server. Aplikace je platformově nezávislá, takže je jedno, na jakém prostředí běží a dokonce i je zcela jedno, na jakém WWW serveru. Je nutné pouze dodržet to, aby server umožňoval použití jazyku PHP a některé z SQL databází. V současné době se na internetu používají 2 největší WWW servery, ale na více jak 80% z nich běží Apache. Je to způsobeno hlavně jeho platformovou nezávislostí a dále tím že jsou volně dostupné jeho zdrojové kódy.

Apache HTTP Server je softwarový webový server s otevřeným kódem pro Linux, BSD, Microsoft Windows a další platformy. Vývoj Apache začal v roce 1993 v NCSA (National Center for Supercomputing Applications) na Illinoiské univerzitě. Více naleznete na [3].

V současné době (květen 2007) je aktuální verze Apache 2.2.4 a tato verze byla také použita při vývoji aplikace. Není ovšem nutnost využívat tento server, stejně dobře poslouží i IIS server od společnosti Microsoft.

## 2.2.2 HTML

HTML je jedním z jazyků, definovaných pomocí ISO standardu Standard Generalized Markup Language (SGML). Jedná se o jazyk, určený k definici strukturálních jazyků (metajazyků). Umožňuje popsat pravidla, kterými se musí řídit text v daném jazyce. Definice HTML je tedy tvořena sadou SGML pravidel, určujících co se smí a co nikoli - čili syntax jazyka. Tato sada pravidel se nazývá Document Type Definition (DTD).

HTML je zkratka z anglického HyperText Markup Language, značkový jazyk pro hypertext. Je jedním z jazyků pro vytváření WWW stránek, který umožňuje publikaci stránek na Internetu. HTML je definováno množinou značek (tagů) a jejich atributů, které blíže specifikují vlastnosti jednotlivých elementů. Jednotlivé části textu se vkládají mezi značky úhlových závorek („<“ „>“) a tím udávají tomuto textu sémantiku.

## 2.2.3 CSS

CSS je zkratka pro anglický název Cascading Style Sheets, česky tabulky kaskádových stylů. Je to jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML.

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

CSS nabízí rozsáhlejší formátovací možnosti než samotné HTML. Např. pro formátování bloku textu – tj. určení vzdálenosti jejich elementů, či okraje stránky, nenabízí HTML nic. CSS má vlastnosti padding a margin. V HTML by bylo potřeba vytvořit složitou konstrukci vnořených tabulek.

Na všech stránkách webové prezentace by měly být všechny nadpisy stejné úrovně, seznamy, zdůrazněné části textu apod. stejného stylu. S použitím formátovacích možností HTML je to obtížné – u každého objektu, v každém dokumentu se vzhled objektu stále znovu nastavuje. S použitím CSS je to velmi jednoduché. Vytvoří se soubor stylu, který se připojuje k HTML dokumentu. Ve všech dokumentech jsou pak objekty stejného vzhledu. Provést změnu vzhledu znamená v případě používání CSS přepsání pouze jediného souboru – souboru stylů.

Výhodou CSS oproti starému formátování v HTML je, že kód a obsah webu je uložen v souboru .html, a veškerý design a formátování se načítá z jednoho souboru .css, který je většinou společný pro celý web. To znamená, že pokud máte v plánu změnu designu webu, stačí změnit pouze jeden soubor .css a změna se aplikuje na celý web. Také se soubor CSS uloží do mezipaměti prohlížeče, a pokud není změněn, tak se načítá pouze jednou. Zobrazení stránek se tím velmi urychlí.



Hlavní nevýhodou CSS je zatím stále špatná podpora v majoritních prohlížečích. Různé prohlížeče interpretují stejný CSS kód jinak a je někdy velmi obtížné jej napsat tak, aby se na všech prohlížečích výsledek zobrazil stejně. Více o CSS na [4].

## 2.2.4 PHP

PHP je skriptovací jazyk pro tvorbu dynamického webu. Původně byl napsán v PERLu a posléze přepsán do jazyka C z důvodu vyšší rychlosti. Tato zkratka původně znamenala Personal Home Page Tools, ale později, po řadě vylepšení a úprav je vykládána jako PHP: Hypertext preprocessor.

K rozšíření toho jazyka přispěla jeho jednoduchost na pochopení a syntaxe podobná jazyku C, multiplatformnost a otevřený zdrojový kód. Jeho hlavní nevýhody spočívají v tom, že je to interpretovaný, nikoliv kompilovaný jazyk, proto kdokoli má přímý přístup k serveru, může nahlédnout do zdrojových kódů našich PHP skriptů.

PHP kód se vkládá do HTML souborů mezi značky `<? ?>` případně `<?php ?>` rozdíl je pouze v tom, že druhé varianta je kompatibilní i se standardem XML, takže je lepší používat tuto.

V současné době (květen 2007) je k dispozici verze 5.2.2 nebo starší vývojová řada verze 4.4.7. Jazyk je neustále vyvíjen a je nejinstalovanějším modulem pro www server Apache.

Při studiu jazyka mi velmi pomohla kniha [5].

## 2.2.5 Javascript

JavaScript je klientský skript. To znamená, že se program odesílá se stránkou na klienta (do prohlížeče) a teprve tam je vykonáván. Protikladem klientských skriptů jsou skripty serverové, které jsou vykonávány na serveru, a na klienta jdou už jen výsledky. JavaScript je často zaměňován s Javou. Java je samostatný programovací jazyk a má s JavaScriptem pouze podobnou syntaxi. JavaScript je jazyk interpretovaný, tudíž se nemusí kompilovat. Kvůli bezpečnosti neumí JavaScript pracovat se soubory, neumí ani žádná data ukládat, jediné s čím je schopný pracovat jsou cookies.

Ke zpracování JavaScriptů je potřeba, aby ho podporoval použitý prohlížeč, a jelikož v dnešní době jsou běžně rozšířené prohlížeče již na dost vysoké úrovni, tak toto scriptování vesměs podporují. Ovšem je třeba počítat s tím, že uživatel může zpracování JavaScriptu v prohlížeči vypnout, a proto je třeba mu nabídnout náhradní variantu, která by byla taktéž funkční.

## 2.2.6 SQL

SQL je standardizovaný dotazovací jazyk používaný pro práci s daty v relačních databázích. SQL je zkratka anglických slov Structured Query Language (strukturovaný dotazovací jazyk).

### 2.2.6.1 Příkazy pro manipulaci s daty

Jsou to příkazy pro získání dat z databáze a pro jejich úpravy. Označují se zkráceně DML – Data Manipulation Language („jazyk pro manipulaci s daty“).

*SELECT* – vybírá data z databáze, umožňuje výběr podmnožiny a řazení dat.

*INSERT* – vkládá do databáze nová data.

*UPDATE* – mění data v databázi.

*DELETE* – odstraňuje záznamy z databáze

### 2.2.6.2 Příkazy pro definici dat

Těmito příkazy se vytvářejí struktury databáze – tabulky, indexy, pohledy a další objekty. Vytvořené struktury lze také upravovat, doplňovat a mazat. Tato skupina příkazů se nazývá zkráceně DDL – *Data Definition Language* („jazyk pro definici dat“).

*CREATE* – vytváření nových objektů

*ALTER* – změny existujících objektů

*DROP* – odstraňování objektů

### 2.2.6.3 Příkazy pro řízení dat

Do této skupiny patří příkazy pro nastavování přístupových práv a řízení transakcí. Označují se jako DCL – *Data Control Language* („jazyk pro ovládání dat“).

*GRANT* – vybírá data z databáze, umožňuje výběr podmnožiny a řazení dat.

*REVOKE* – vkládá do databáze nová data.

*BEGIN* – mění data v databázi.

*COMMIT* – odstraňuje záznamy z databáze

*ROLLBACK* – mění data v databázi.

### 2.2.6.4 Ostatní příkazy

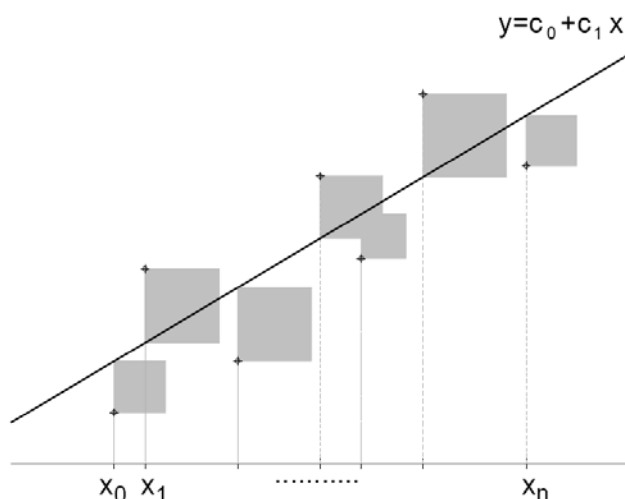
Do této skupiny patří příkazy pro správu databáze. Pomocí nich lze přidávat uživatele, nastavovat systémové parametry (kódování znaků, způsob řazení, formáty data a času apod.). Tato skupina není standardizována a konkrétní syntaxe příkazů je závislá na databázovém systému. V některých dialektech jazyka SQL jsou přidány i příkazy pro kontrolu běhu, takže lze tyto dialekty zařadit i mezi programovací jazyky. Více o jazyku SQL naleznete na [6].

# 3 Predikce

V této kapitole popíšu metody použité pro predikci ceny zboží a jejich využití v aplikaci. Pojem predikce označuje předpověď nebo odhad budoucího stavu. Na základě dat z minulosti se snažíme s větší či menší úspěšností předpovědět budoucí stav.

## 3.1 Metoda nejmenších čtverců

Hledáme mezi všemi funkcemi tohoto známého typu takovou, která prochází k zadaným bodům v jistém smyslu nejlépe.



**Obrázek. 3.1:** Hledáme přímku pro kterou je součet obsahu čtverců minimální

Výchozí situace je tato: Jsou dány body  $x_i, i = 0, \dots, n$ , a funkční hodnoty v nich  $y_i$ . Budeme hledat přímku o rovnici

$$y = c_0 + c_1 x,$$

která bude co „nejlépe“ procházet mezi body  $[x_i, y_i], i = 0, \dots, n$ . Chybu aproximace v  $i$ -tém bodě potom značíme jako  $e_i$ .

$$e_i = y_i - y(x_i) = y_i - c_0 - c_1 x_i$$

Jelikož body  $[x_i, y_i]$  jsou dány, chyba aproximace závisí pouze na koeficientech přímky  $c_0$  a  $c_1$ .

Ukazuje se, že vhodné kritérium pro určení onoho „co nejlepšího“ procházení je, aby součet druhých mocnin (neboli čtverců) chyb v jednotlivých bodech byl minimální. Tento součet značíme  $\rho^2$ . Chceme tedy minimalizovat funkci

$$\begin{aligned}\rho^2(c_0, c_1) &= (y_0 - c_0 - c_1 x_0)^2 + (y_1 - c_0 - c_1 x_1)^2 + \dots + (y_n - c_0 - c_1 x_n)^2 \\ &= \sum_{i=0}^n (y_i - c_0 - c_1 x_i)^2\end{aligned}$$

Veličinu  $\rho^2$  nazýváme kvadratická odchylka. Z diferenciálních rovnic více proměnných je známo, že nutnou podmínkou pro to, aby  $\rho^2(c_0, c_1)$  nabývala minima je splnění rovnic

$$\frac{\partial(\rho^2)}{\partial c_0} = 0 \quad \text{a} \quad \frac{\partial(\rho^2)}{\partial c_1} = 0$$

Parciální derivaci podle  $c_0$  potom vypočteme následovně

$$\begin{aligned}\frac{\partial(\rho^2)}{\partial c_0} &= 2(y_0 - c_0 - c_1 x_0)(-1) + 2(y_1 - c_0 - c_1 x_1)(-1) + \dots + 2(y_n - c_0 - c_1 x_n)(-1) = \\ &= -2((y_0 - c_0 - c_1 x_0) + (y_1 - c_0 - c_1 x_1) + \dots + (y_n - c_0 - c_1 x_n)) = \\ &= -2((y_0 + y_1 + \dots + y_n) + (1 + 1 + \dots + 1) + \dots + c_1(x_0 + x_1 + \dots + x_n)) = \\ &= -2\left(\sum_{i=0}^n y_i - c_0(n+1) - c_1 \sum_{i=0}^n x_i\right) \\ \frac{\partial(\rho^2)}{\partial c_1} &= \sum_{i=0}^n 2(y_i - c_0 - c_1 x_i)(-x_i) = -2 \sum_{i=0}^n (y_i x_i - c_0 x_i - c_1 x_i^2) = \\ &= -2\left(\sum_{i=0}^n x_i y_i - c_0 \sum_{i=0}^n x_i - c_1 \sum_{i=0}^n x_i^2\right)\end{aligned}$$

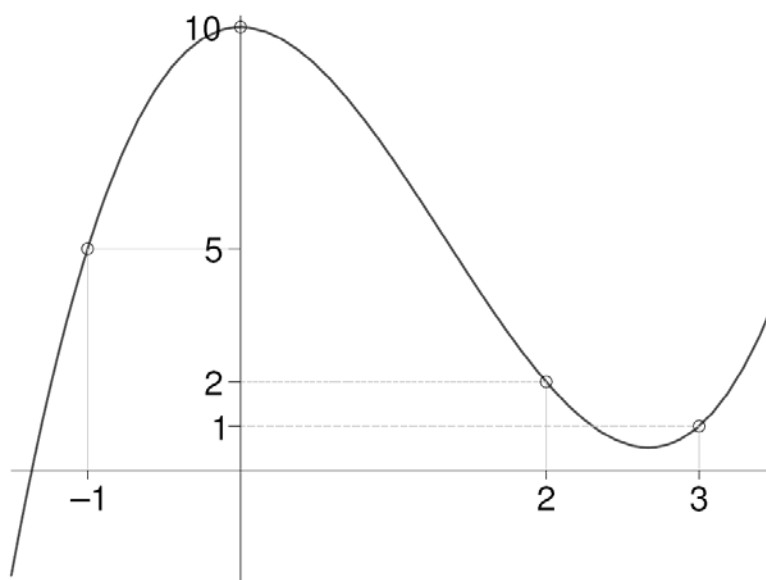
Položíme-li nyní vypočtené parciální derivace rovny 0, po snadné úpravě (vydělením -2 a převedením některých sum na druhou stranu rovnic) dostaneme tzv. normální rovnice s neznámými  $c_0$  a  $c_1$ :

$$\begin{aligned}c_0(n+1) + c_1 \sum_{i=0}^n x_i &= \sum_{i=0}^n y_i \\ c_0 \sum_{i=0}^n x_i + c_1 \sum_{i=0}^n x_i^2 &= \sum_{i=0}^n x_i y_i\end{aligned}$$

Pokud mezi uzly  $x_i$  najdeme alespoň dva různé (tj. pokud není  $x_0 = x_1 = \dots = x_n$  neboli pokud jsou vektory  $(1, 1, \dots, 1)$  a  $(x_0, x_1, \dots, x_n)$  lineárně nezávislé), má tato soustava jediné řešení. Toto řešení potom dosadíme do  $y = c_0 + c_1 x$ , a máme výslednou rovnici konkrétní přímky.

## 3.2 Lagrangeův interpolační polynom

U interpolace se požaduje, aby aproximující funkce měla s funkcí původní v určitých bodech stejné hodnoty, na rozdíl od metody nejmenších čtverců, kde má aproximující funkce procházet zadaným bodům v jistém smyslu nejbližší, ale přímo jimi procházet nemusí.



**Obrázek. 3.1:** Příklad zadaných bodů a vzhledu výsledného polynomu

Langrangeův interpolační polynom je jedním z neznámějších způsobů interpolace funkce zadané pouze v diskrétních bodech.

Interpolační polynom daný body  $[x_i, f_i], i = 0, \dots, n$  sestavíme pomocí polynomů  $l_i(x)$  takových, že

$$l_i(x_j) = \begin{cases} 1 & \text{pro } i = j \\ 0 & \text{pro } i \neq j \end{cases}$$

Lze snadno ověřit, že polynom

$$l_0 = \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)}$$

Má v  $x_0$  hodnotu 1 a v ostatních uzlových bodech hodnotu 0. Podobně dostaneme i ostatní polynomy

$l_i, i = 0, \dots, n$ :

$$l_i(x) = \frac{(x-x_0)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_0-x_1)(x_i-x_0)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}$$

Interpolační polynom  $P_n(x)$  nyní dostaneme jako kombinaci  $l_i(x)$ :

$$\begin{aligned} P_n(x) &= f_0 l_0(x) + f_1 l_1(x) + \dots + f_n l_n(x) = \\ &= f_0 \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)} + f_1 \frac{(x-x_0)(x-x_2)\dots(x-x_n)}{(x_1-x_0)(x_1-x_2)\dots(x_1-x_n)} + \dots \\ &\quad \dots + f_n \frac{(x-x_0)(x-x_2)\dots(x-x_{n-1})}{(x_n-x_0)(x_n-x_1)\dots(x_n-x_{n-1})} \end{aligned}$$

Více o interpolačních metodách naleznete na [7].

## 3.3 Predikce vývoje cen

Pro svůj systém jsem zvolil predikci cen zboží, odhadovanou na základě vývoji cen v minulosti. Predikce používá 2 matematické metody a to Lagrangův polynom a Metodu nejmenších čtverců. Každá z těchto metod používá zcela jiný princip (přímka / polynom) a proto se při selhání jedné metody je vysoká pravděpodobnost, že druhá metoda bude dávat rozumný výsledek. Aby byla předpověď pokud možno co nejpravděpodobnější vybírá se výsledek té metody, která se aktuálně nejlépe hodí na soubor dat.

### 3.3.1 Způsob výběru metody

Abychom byli schopní rozhodnout, která z metod dává lepší výsledek v konkrétním případě aplikujeme několik jednoduchých podmínek :

- Pokud některá z metod predikuje záporné číslo, je nesmyslné o této variantě vůbec uvažovat.
- Pokud byly poslední 2 nebo více cen shodné, tak s největší pravděpodobností bude i následující cena zboží stejná. Obzvláště ve firmě se sortimentem měnícím ceny maximálně párkrát do roka (dle zkušeností se jedná obvykle o čtvrtletní ceny).
- Pokud neplatí a) ani b) musíme rozhodnout kterou metodu zvolit, toto uděláme tak že necháme predikovat současnou cenu zboží pouze na základě historie cen minulých. Každá metoda nám vrátí svůj výsledek a pak stačí porovnat jeho odchylku výsledku od aktuální ceny. Metoda, která bude mít nejnižší odchylku bude použita.
- Výběr predikce ceny na delší dobu při klesajícím nebo stoupajícím trendu cen může i tak způsobit, že výsledek bude v záporných číslech, nebo v číslech, které jsou řádově jiné než

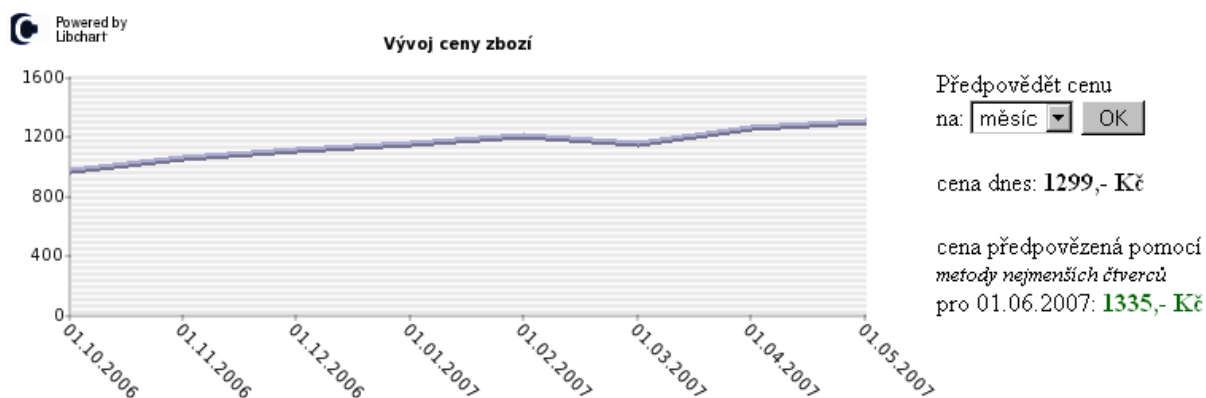
byla historická maximální cena nebo minimální cena. Toto bylo třeba vhodným způsobem omezit. Aktuálně jsem zvolil nastavení na 150% historického maxima a 50% minima. Toto nastavení bylo zvoleno dle zkušeností se sortimentem, který bude nabízen v tomto obchodě. Jakékoliv vyšší změny cen jsou vysoce nepravděpodobné a to i v časovém horizontu 12 měsíců.

Aplikací těchto pravidel se mi podařilo odladit systém predikce tak, že dává rozumné výsledky, které se dají očekávat.

V systému jsou tyto 2 matematické metody volány funkcemi  $Lagrange(x, y, bod\_x)$  a  $MetodaNejCtvercu(x, y, bod\_x)$ . Parametry  $x, y$  jsou předávány jako pole souřadnic ( $x$  – reprezentuje datum o ve formě čísel udávajících časové vzdálenosti mezi jednotlivými dny se změnou cen a  $y$  obsahuje adekvátní historickou cenu. V proměnné  $bod\_x$  je souřadnice, reprezentující datum na ose  $x$ ). Funkce vrací hodnotu na ose  $y$  tedy predikovanou cenu zboží.

### 3.3.2 Prezentace předpovědi cen

Po úvaze o způsobu, jak by bylo vhodné prezentovat výsledky této předpovědi, jsem se rozhodl je umístit po pravé straně grafu vývoje cen u každého zboží. Uživatel bude mít možnost zvolit si časový horizont který si přeje a také se dozví která z metod byla k předpovědi použita. Do grafu se ovšem tato hodnota nezakresluje z důvodu aby nemátla zákazníky.



Obrázek. 3.3.2: Způsob prezentace předpovědi cen

## 4 Analýza řešení

Tato kapitola obsahuje postup při návrhu systému. Nástroje pro tvorbu systému jsme si již zvolili v 4. kapitole.

### 4.1 Role

Z požadavku na aplikaci vyplývá, že jednotlivým uživatelům systému má být umožněno využívat pouze ty funkce, pro které budou mít patřičná oprávnění. Vytvářet seznam práv pro každého uživatele je v tomto případě neefektivní a zbytečné. Proto pro předem definujeme skupiny, které budou mít různé oprávnění k přístupu do systému.

Na nás je tedy zvolit, kolik těchto skupin bude. Při prvním pohledu na požadavky se hned nabízejí dvě základní. A to skupina *zákazníků* a skupina *zaměstnanců*. Tyto dvě skupiny budou používány nejčastěji. Kromě toho je dobré mít v každém systému jednu skupinu s nejvyššími oprávněními, která by dokázala spravovat systém a uživatele v něm. V naší aplikaci to bude skupina *Administrátor*.

#### 4.1.1 Způsob zabezpečení systému

Způsob zabezpečení systému se odvíjí od způsobu autorizace v naší aplikaci. Jelikož se jedná o veřejný internetový obchod se zákaznickou a administrátorskou sekcí, bylo třeba zvolit nějaký způsob jak autorizovat (obvykle přístupovým jménem a heslem) a bezpečně autentizovat (rozlišit jestli uživatel, který odesílá požadavek je skutečně tím, kdo k tomuto požadavku má patřičná oprávnění). V podstatě jsem vybíral ze dvou autentizačních metod.

**Cookies** odesílá od uživatele data s jednoznačným identifikačním klíčem (náhodně generovaný řetězec). Při dalším požadavku odešle prohlížeč s dotazem ty samé data, a tak bude server vědět, s kým právě komunikuje. Jeho velkou nevýhodou ovšem je, že funguje pouze pokud jej uživatel v prohlížeči nezakázal, a to bohužel není nikdy jisté.

**Sessions** představují prostředek v jazyku PHP, jak mezi jednotlivými přístupy zachovávat a předávat data. Ke spuštění session slouží funkce `session_start()`. Ta nejdříve zkontroluje, zda bylo aplikaci předáno jméno nějaké již probíhající session. Pokud ne, je vytvořena nová session. V opačném případě je znovu spuštěna předaná probíhající session. Všechna data jsou uložena v superglobálním poli `$_SESSION[]`. Zpracování sessions je o něco pomalejší než u cookies, ale se zvyšujícím se objemem přenášených dat je rozhodně sessions výhodnější.

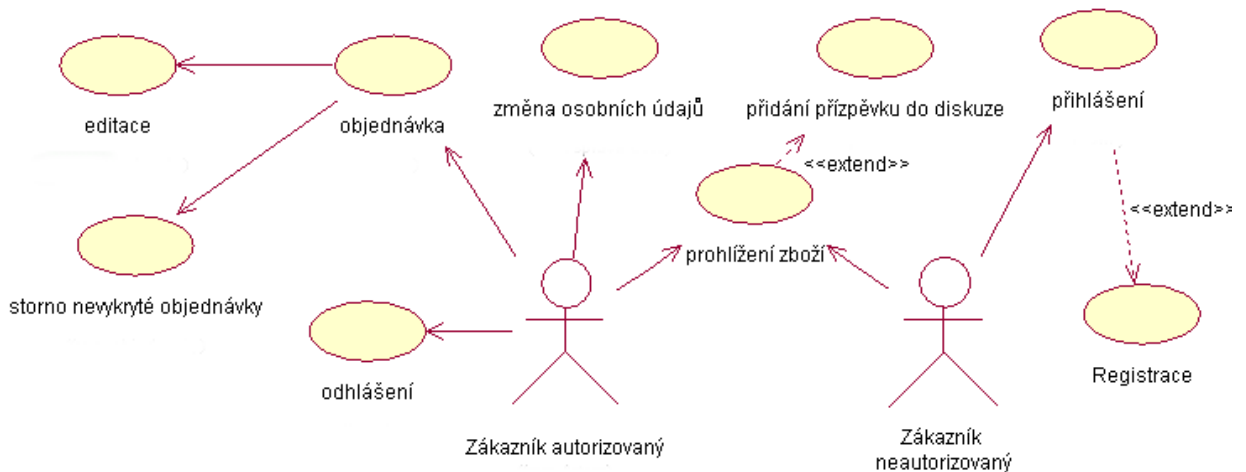
Po zvážení všech výhod a nevýhod jsem se tedy rozhodl využívat pro autentizaci uživatelů sessions.



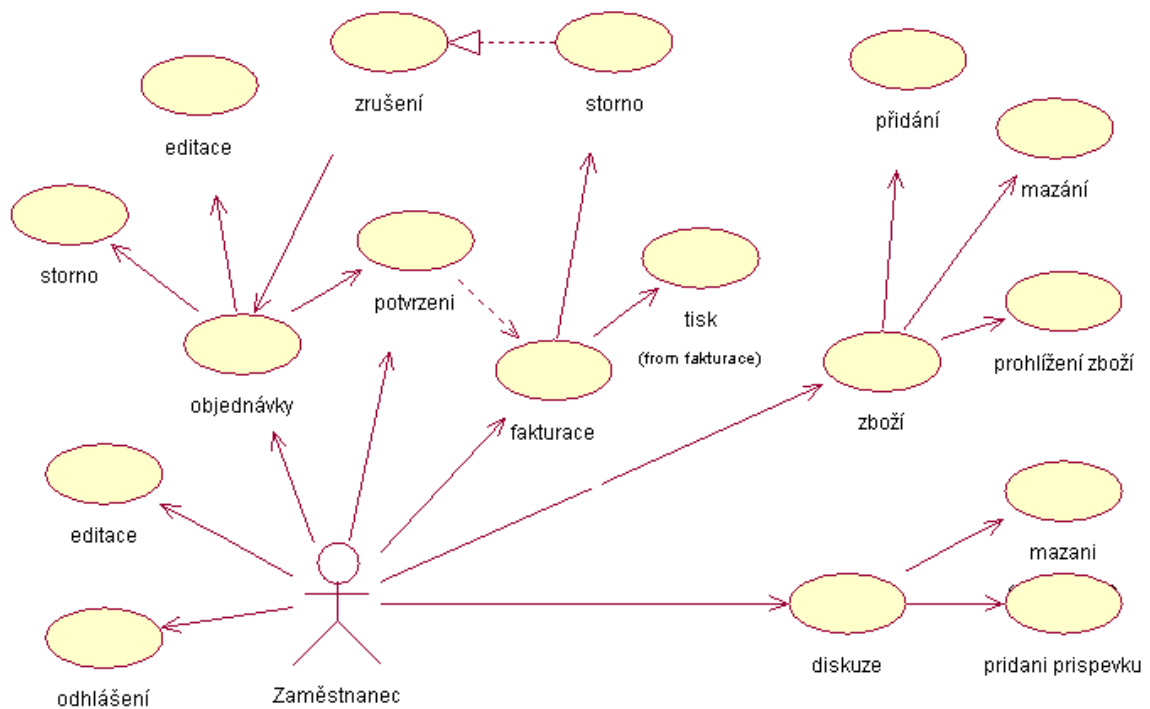
## 4.1.2 UML

Pro základní návrh systému z hlediska uživatelských rolí jsem použil modelovací jazyk UML (Unified Modeling Language), který se používá v softwarovém inženýrství k návrhu, specifikaci, dokumentaci a vizualizaci programových systémů.

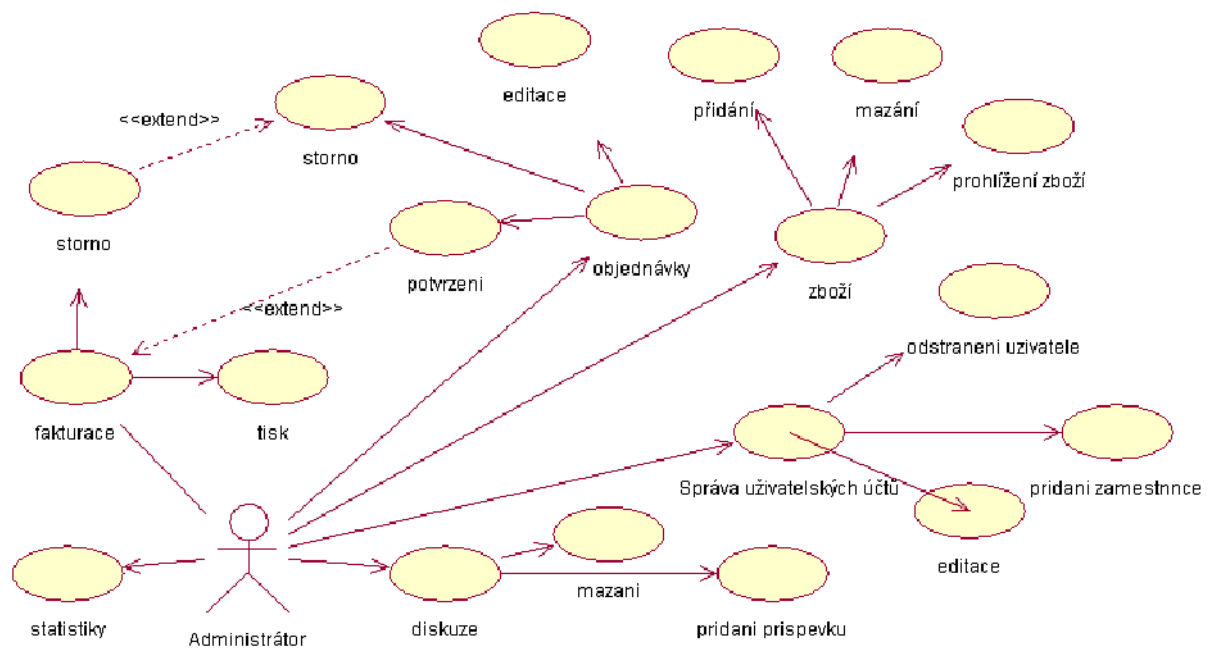
## 4.2 Use Case Model



Obrázek. 4.2.1: Use Case Model Zákazníka



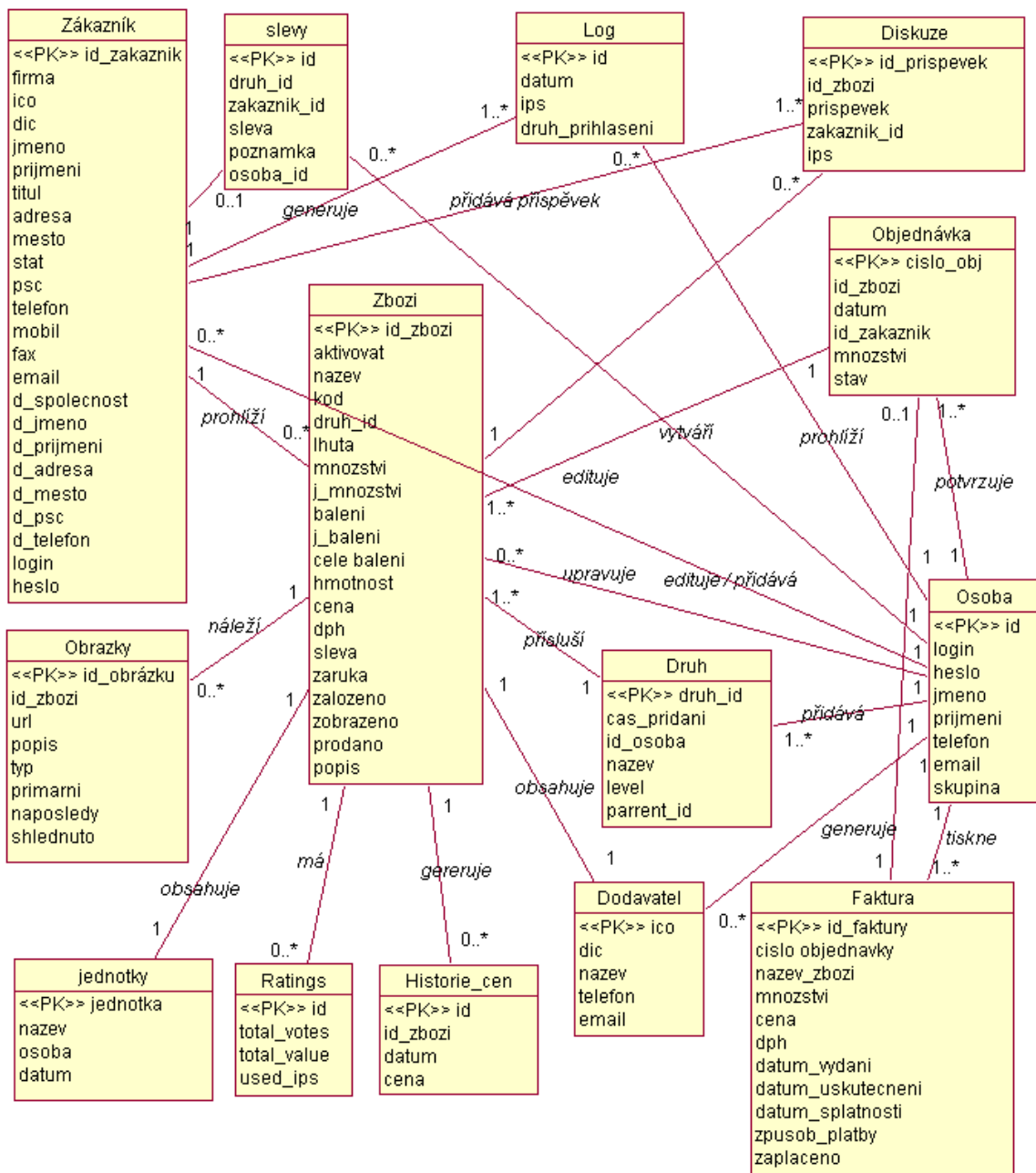
Obrázek. 4.2.2: Use Case Model Zaměstnanec



Obrázek. 4.2.3: Use Case Model Administrátora

## 4.3 ER diagram

ER diagram byl zaveden a poprvé použit Peterem Chenem v roce 1976 [8]. Brzy došlo k jeho rozšíření a stal se obecně uznávaným standardem. Diagram obsahuje typy entit a typy vztahů. **Entitní typ** představuje nezávisle existující objekt reálného světa. **Vztahový typ** je vazba mezi dvěma nebo více typy entit. Samotný pojem **entita** či **vztah** označuje výskyt (instanci) typu entity či typu vztahu.



Obrázek. 4.3: ER diagram systému

# 5 Implementace

Tato kapitola pojednává o požadavcích na aplikaci a jejich řešení v praktické realizaci projektu.

## 5.1 Uživatelské prostředí

Uživatelské prostředí je hlavní část aplikace, se kterou přijde zákazník do styku. Musí být tedy maximálně jednoduchá, a přitom uživatelsky přívětivá, aby nikoho neodradila od nákupu.

### 5.1.1 Přihlašování

Po analyzování požadavků jsem se rozhodl, že do základní nabídkové části systému lze vstoupit bez jakékoliv autentizace. Kdokoliv tedy uvidí přehled nabízeného zboží a může procházet jednotlivé kategorie a vybírat.

Autorizovat se tedy bude muset zákazník až v době, kdy to bude nezbytně nutné. Tedy těsně před odesláním objednávky.

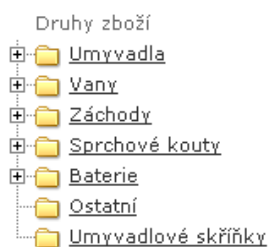
Nový zákazník se má možnost se kdykoliv zaregistrovat. Registrace nového zákazníka jsem se snažil udělat maximálně jednoduše. Všechny zadávané hodnoty jsou testovány na správnost zadávaných hodnot a uživatel je informován o chybách v jednotlivých kolonkách.

Uživatelé, kteří již dříve v obchodě nakupovali se můžou přihlásit a získají tím přístup ke svým předešlým objednávkám, a mohou mít nastaveny i jiné ceny (slevy), pokud mají z důvodů vyšších odběrů domluveny vyšší slevy. Po přihlášení se zákazníkovi rovněž ukáže datum a čas jeho posledního přístupu. Toto je vhodné hlavně z důvodu bezpečnostních, aby si každý mohl zkontrolovat jestli jeho heslo nebylo náhodou vyraženo, a nepřihlašuje se na jeho účet někdo jiný.

Dále bylo aplikováno automatické odhlášení klienta po určité době nečinnosti (v současné době 30 minut). Toto zabezpečení částečně zabrání zneužití účtu na počítači sdíleném více uživateli v případě, že uživatel nemá základní bezpečnostní návyky, a odhlášení ze systému nepovažuje za nutné.

### 5.1.2 Procházení zboží

Procházení zboží bylo opět třeba udělat co nejjednodušeji, aby žádný zákazník neměl problém si vybrat zboží, které ho zajímá. V levé části aplikace má uživatel stromovou strukturu jednotlivých kategorií zboží. Touto strukturou může libovolně procházet, přičemž zvolení kategorie vede k výpisu náležejícího zboží.



Obrázek. 5.1.2: Stromová struktura pro výpis kategorií

## 5.2 Administrátorská část

Pro potřeby spravování aplikace elektronického obchodu, bylo nutné vytvořit systém, s pravidly, pro přístup k jednotlivým částem nabídky. Do administračního módu se mohou tedy přihlásit pouze ti uživatelé, vlastníci patřičné oprávnění.

### 5.2.1 Objednávky

Zaměstnanec vidí objednávky jednotlivých uživatelů, a má možnost nastavit jejich stav (objednáno, na cestě, vyřízeno). Po nastavení objednávky na vyřízeno, se automaticky vygeneruje faktura, kterou má možnost vytisknout jak zaměstnanec, tak je současně přístupná i u zákazníka v historii jeho nákupů.

### 5.2.2 Editace kategorií

Pro editaci kategorií zboží se využívá stejná stromová struktura, jako pro procházení zboží (Obrázek. 5.1.2). Pro zaměstnance je to tedy jednoduché a intuitivní řešení. Lze vytvořit libovolnou strukturu kategorií zboží. Mazat lze samozřejmě pouze neprázdné uzly této stromové struktury.

### 5.2.3 Editace Zboží


Pro editaci zboží slouží zvláštní formulář, přímo navržený pro potřeby konkrétní firmy. Kromě klasických atributů zboží, jsou zde méně obvyklé věci jako prodej pouze celých balení, hmotnost (některý sortiment je velice těžký a zákazník může mít problém ho v osobním automobile odvézt) a balení spolu s jednotkou balení. U některých druhů zboží totiž nemusí souhlasit množství a balení (např. nabídka 1 m<sup>2</sup> a v balení je např. 11 ks).

### 5.2.4 Vkládání obrázků

Další podstatnou výhodou tohoto obchodu je možnost u každého zboží přiložit obrázek. Dokonce je mé řešení pružné natolik, že dokáže přiložit libovolné množství obrázků. Uživatel nemusí ani

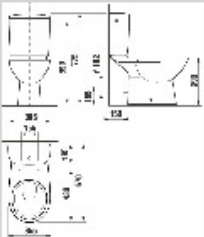
provádět žádnou úpravu fotografií, náhledy a miniatury se generují automaticky pomocí GD knihovny, která je součástí PHP. Jediná jeho práce spočívá v tom že zadá popis obrázku, zvolí jestli ho ponechat v nezměněné formě nebo ho upravit na konkrétní rozlišení (640x480, 1024x768, atd.). Nejlépe si lze tento formulář představit podle příloženého obrázku 5.2.4.

Máte vložené 2 obrázky :



**Popis obrázku:**

základní obrázek



**Popis obrázku:**

základní obrázek

**Vkládání obrázků ke zboží...**

**Popis:**  **Velikost:**

Obrázek. 5.2.4: Vkládání obrázků ke zboží

## 5.3 Využité knihovny

### 5.3.1 libchart

Knihovna libchart je jednoduchá knihovna pro tvorbu grafů pomocí PHP. Grafy jsou uloženy jako obrázky ve formátu PNG. Knihovna umožňuje tvorbu sloupcových (vertikálních nebo horizontálních) a kruhových grafů. Pro její využívání je nutné mít na serveru PHP s podporou GD (2.0 nebo vyšší) a FreeType (2.0 nebo vyšší).

Používání knihovny je jednoduché. Nejdřív musíme vytvořit nový objekt třídy, kterou zvolíme podle požadovaného typu grafu (`HorizontalChart`, `VerticalChart` nebo `PieChart`). Při inicializaci můžeme třídě předat požadovanou výšku a šířku grafu. Knihovna obsahuje několik důležitých metod :

`setTitle($Title)` Řetězec se zobrazí jako titulek nad grafem.

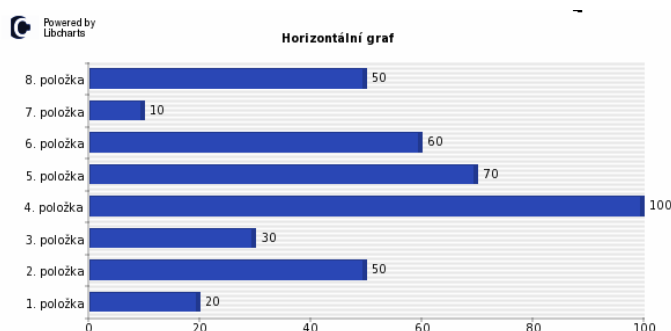
`setLabelMarginLeft($Margin)` Nastavení velikosti levého okraje (podobnou funkci mají `setLabelMarginRight`, `setLabelMarginTop` a `setLabelMarginBottom`).

`setMargin($Margin)` Nastavení velikosti všech okrajů najednou.

`getLowerBoundary` a `getUpperBoundary` Nastavení minimální (lower) a maximální (upper) hodnoty na ose y.

`addPoint($Point)` Vložení nové položky do grafu.

`render($File)` Vytvoří graf a uloží ho pod názvem \$File.



Obrázek. 5.4.1: Příklad horizontálního grafu

S prací s touto knihovnou jsem byl spokojen, chybí snad jen větší možnosti nastavení parametrů. Jelikož podporu GD (2.0 nebo vyšší) a FreeType (2.0 nebo vyšší) mívají instalace Apache+PHP dnes již standardně, a nebylo třeba nic doinstalovávat, nepředpokládám tedy žádné problémy při nasazení systému.

### 5.3.2 AJAX Star Rating Bar

Je hlasovací skript v PHP & MySQL, který umožní uživatelům hodnotit věci na www stránkách. Jeho hlavní výhodou je že funguje bez nutnosti refreshu stránky. Pokud tedy není zapnut Javascript v prohlížeči skript funguje i nadále s ovšem s nutností refreshu stránky po hlasování.



Hodnocení zboží: 5.0/10 ( hlasoval 1 uživatel )

**Obrázek. 5.4.2:** Příklad zobrazení hodnocení zboží

Tento skript se také v praxi osvědčil, bylo ho nutné pouze počestit, přidat skloňování češtiny a nastavit správně cesty ke zdrojovým souborům.

### 5.3.3 dTree

dTree je skript generující Javascriptové menu. Je velmi jednoduchý na použití, mezi jeho výhody patří především možnost vkládat jednotlivé uzly do stromu v náhodném pořadí, což se výborně hodí pro použití s databází. Umožňuje neomezený počet uzlů, pamatuje si stav otevřených uzlů při přechodu mezi stránkami. Generuje XHTML 1.0 strict validní kód. Má možnost vložení alternativních obrázku ke každému uzlu.



## 6 Porovnání systému s již existujícími

Internetových obchodů se sortimentem obkladů dlažeb a sanitární keramiky je zatím poměrně málo. Za nejznámější obchod mezi zákazníky považují <http://www.siko-koupelny.cz/> patřící firmě SIKO KOUPELNY a.s., která provozuje řetězec obchodů ve větších městech naší republiky.

Tento obchod je samozřejmě zpracován tak jak se u firmy této velikosti očekává. Nemá příliš chyb, ale ani nemá pro uživatele tolik možností nabídky. U každého zboží je pouze malý popis, a kromě ceny nenabízí tento obchod žádné další informace o dodávce (množství skladem, balení, hmotnost, dodací lhůta...). Zajímavou funkcí může být Siko oblíbené, do které ho uživatel ukládá oblíbené položky. U obchodů s velkým množstvím nabízeného zboží a rozsáhlého sortimentu je toto pravděpodobně užitečná funkce. Tento obchod běží na platformě ASP .NET. Do způsobu ani možností administrace toho obchodu nevidím, nelze tedy říci v čem by byla lepší nebo horší. Po zvážení všech pro a proti lze říci, že při malé investici do dalšího vývoje je pravděpodobné, že bude i má aplikace na stejné úrovni.

Další z nalezených obchodů podobného zaměření je <http://www.ceske-vany.cz/>. Za zmínku stojí jeho výborné grafické zpracování. Obsahuje zajímavé možnosti výběru van podle tvarů. Díky tomu, že je úzce orientován pouze na sortiment van, mohla být jeho uživatelská přívětivost dotažena na velmi vysokou úroveň.

Systém obdobného rozsahu a složitosti jako můj projekt je provozován na <http://www.koupelny-vybaveni.cz/>.

Největší nevýhodou těchto systémů je jejich tržní cena. Obvykle se takovéto systémy bez naplnění dat dají pořídit v řádově v desítkách tisíc korun. Jelikož jsem souhlasil se zveřejněním práce může si někdo s obdobným sortimentem moji aplikaci stáhnout zcela zdarma.

Samozřejmě existují i mnohem rozsáhlejší projekty, které jsou zdarma např. GNU projekt Zen Cart, ale díky jeho velkému rozsahu a obecnosti, nejsou jeho úpravy na míru zcela jednoduché.

## 7 Možnosti rozšíření

Má aplikace elektronického obchodu poskytuje v současné době nadprůměrné možnosti nabídky a prezentace zboží. Určitě však je mnoho možností jak program ještě vylepšit a doplnit. Zde je výčet nápadů, které by bylo možné dále aplikovat :

- Online napojení na skladový systém firmy, což v současné době z technických důvodů nelze realizovat (Novell 3.11 na IPX s obsazeným limitem 5 uživatelů)
- Dále by bylo pro zákazníky vhodné přidat žádané slevové akce a novinky
- Přidání modulu pro rozesílání reklamních informací o novinkách a akcích
- Použití některého šablonovacího systému (např. SMARTY), který lépe oddělí aplikační logiku od prezentace dat a zjednoduší budoucí správu a úpravy aplikace
- Aplikování některého profesionálního designu pro lepší vzhled.

## 8 Závěr

Cílem mé práce bylo prostudovat problematiku informačních systémů používaných pro prodej a nabídku zboží na Internetu, navrhnout a vytvořit elektronický obchod na míru podle specifických požadavků firmy. Hlavním cílem bylo zorientovat se v problematice a navrhnout možné řešení.

Práce na tomto projektu pro mne byla velmi přínosná, neboť jsem se při ní naučil, jak provádět vývoj webové aplikace tak, aby práce byla efektivní a nevznikaly nepředvídatelné problémy. Pochopil jsem tedy, že pouhé programování nestačí a kvalitní návrh aplikace je opravdu důležitý.

Důležitým cílem práce bylo dostat vývoj aplikace do podoby, kdy bude reálně použitelná. Což si myslím, že se mi povedlo, i přes poměrně velkou rozsáhlost tohoto projektu.

Celkově mi práce aplikaci přinesla hlavně značné množství zkušeností z praktického využití teoretických poznatků získaných při studiu.

# Literatura

- [1] Wikipedie. *Otevřená encyklopedie*. [online], 2007, [rev. 1.5.2007],[cit. 3.5.2007].  
URL <http://cs.wikipedia.org/wiki/Internet>
- [2] Wikipedie. *Otevřená encyklopedie*. [online], 2007, [rev. 1.5.2007],[cit. 3.5.2007].  
URL <http://cs.wikipedia.org/wiki/Http>
- [3] Wikipedie. *Otevřená encyklopedie*. [online], 2007, [rev. 1.5.2007],[cit. 4.5.2007].  
URL <http://cs.wikipedia.org/wiki/Www>
- [3] Wikipedie. *Otevřená encyklopedie*. [online], 2007, [rev. 1.5.2007],[cit. 4.5.2007].  
URL <http://cs.wikipedia.org/wiki/Url>
- [4] Wikipedie. *Otevřená encyklopedie*. [online], 2007, [rev. 1.5.2007],[cit. 4.5.2007].  
URL <http://cs.wikipedia.org/wiki/Css>
- [5] W. Jason Gilmore. *Velká kniha PHP 5 a MySQL*. Brno, Zoner Press 2005.
- [6] Wikipedie. *Otevřená encyklopedie*. [online], 2007, [rev. 1.5.2007],[cit. 4.5.2007].  
URL <http://cs.wikipedia.org/wiki/Sql>
- [7] RNDr. Břetislav Fajmon, Ph.D. a Mgr. Irena Růžičková. : *Matematika 3*, Skriptum, FEKT VUT v Brně, 2006.
- [8] Sanders G. L., *Data modeling*, boyd & fraser publishing company, 1995.

# Seznam příloh

1. Příložené CD obsahuje:
  1. Zdrojové kódy programu.
  2. Ukázkový SQL skript se vzorkem dat.
  3. Návod na instalaci aplikace.
  4. Elektronická verze technické zprávy ve formátu PDF.