

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

VYUŽITÍ VÍCEÚROVŇOVÝCH ASOCIAČNÍCH PRAVIDEL V INTERNETOVÉM OBCHODU

BAKALÁŘSKÁ PRÁCE

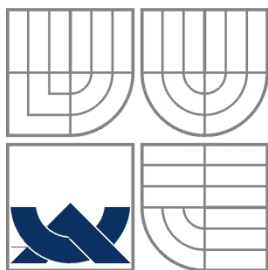
BACHELOR'S THESIS

AUTOR PRÁCE

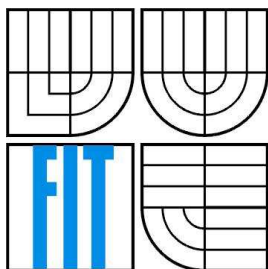
AUTHOR

MICHAEL KRÁLÍK

BRNO 2012



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

VYUŽITÍ VÍCEÚROVŇOVÝCH ASOCIAČNÍCH PRAVIDEL V INTERNETOVÉM OBCHODU

UTILIZATION OF MULTIPLE LEVEL ASSOCIATION RULES IN AN E-SHOP

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MICHAEL KRÁLÍK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. VLADIMÍR BARTÍK, Ph.D.

BRNO 2012

Zadání bakalářské práce

Řešitel: **Králík Michael**

Obor: Informační technologie

Téma: **Využití víceúrovňových asociačních pravidel v internetovém obchodu**
Utilization of Multiple Level Association Rules in an e-Shop

Kategorie: Databáze

Pokyny:

1. Analyzujte požadavky na internetový obchod, který bude zahrnovat také vedení skladu, automatické zpracování objednávek, vedení plateb, daňovou evidenci a zobrazování různých statistik. Návrh konzultujte s vedoucím.
2. Seznamte se s problematikou dolování dat a zaměřte se přitom na víceúrovňová asociační pravidla.
3. Po dohodě s vedoucím zvolte vhodný algoritmus pro dolování víceúrovňových asociačních pravidel a zamyslete se nad jeho možným využitím v prostředí obchodu.
4. Navržený internetový obchod včetně algoritmu zvoleného v bodě 3 implementujte.
5. Ověřte funkčnost implementovaného řešení a proveďte experimenty s vhodně zvoleným vzorkem dat.
6. Zhodnoťte dosažené výsledky a další možné pokračování v tomto projektu.

Literatura:

- Han, J., Kamber, M.: Data Mining - Concepts and Techniques, 2nd Edition. Morgan Kaufmann Publishers, 2006.

Při obhajobě semestrální části projektu je požadováno:

- SP byl již obhájen.

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).

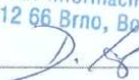
Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo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

Vedoucí: **Bartík Vladimír, Ing., Ph.D., UIFS FIT VUT**

Datum zadání: 1. listopadu 2011

Datum odevzdání: 16. května 2012

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


doc. Dr. Ing. Dušan Kolář
vedoucí ústavu

Abstrakt

Tato bakalářská práce se zabývá návrhem a implementací kompletního informačního systému pro potřeby elektronického obchodu, v jehož prostředí bude využito víceúrovňových asociačních pravidel pro dolování dat. Informační systém bude dále obsahovat vedení skladu, automatické zpracování objednávek, vedení plateb, daňovou evidenci a zobrazování různých statistik. K implementaci tohoto systému byly použity technologie HTML, Latte, CSS, JavaScript, Nette Framework, PHP a MySQL.

Abstract

This thesis deals with design and implementation of a complete information system for the needs of electronic shop, where multiple level association rules will be used for data mining. The information system will include warehouse management, automated order processing, payment management, tax records and display various statistics. Technologies such as HTML, Latte, CSS, JavaScript, Nette Framework, PHP and MySQL were used to implement this system.

Klíčová slova

Informační systém, databáze, dolování dat, víceúrovňová asociační pravidla, MySQL, Nette, framework, PHP, HTML, CSS, Latte, JavaScript, webová aplikace, elektronický obchod, administrace, statistiky

Keywords

Information system, database, data mining, multiple level association rules, MySQL, Nette, framework, PHP, HTML, CSS, Latte, JavaScript, web application, electronic shop, administration, statistics

Citace

Králík Michael: Využití víceúrovňových asociačních pravidel v internetovém obchodu, bakalářská práce, Brno, FIT VUT v Brně, 2012

Využití víceúrovňových asociačních pravidel v internetovém obchodu

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Vladimíra Bartíka, Ph.D.

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

.....

Michael Králík

12. 5. 2012

Poděkování

Na tomto místě bych rád poděkoval Ing. Vladimíru Bartíkovi za jeho odborné vedení, užitečné rady a připomínky při zpracování této bakalářské práce.

© Michael Králík, 2012

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

1.	Úvod.....	3
2.	Analýza požadavků na IS.....	4
2.1	Dolování užitečných dat.....	4
2.2	Vedení skladu zboží	4
2.3	Správa uživatelů a zaměstnaneckých rolí.....	4
2.4	Automatizace zpracování objednávek.....	4
2.5	Vedení pohledávek, uskutečněných plateb a daňová evidence	5
2.6	Backend statistiky prodeje	5
3.	Použité technologie.....	6
3.1	HTML a Latte.....	6
3.2	CSS.....	7
3.3	JavaScript	7
3.4	Skriptovací jazyk PHP	8
3.5	MySQL.....	8
3.6	Nette Framework.....	8
3.6.1	Model – View – Controller (MVC)	9
3.6.2	Model – View – Presenter (MVP)	10
3.7	Data mining.....	10
3.8	Dolování asociačních pravidel a frekventovaných množin.....	11
3.8.1	Jednoúrovňová asociační pravidla	11
3.8.2	Víceúrovňová asociační pravidla	12
3.8.3	Generování asociačních pravidel z frekventovaných množin.....	13
4.	Návrh systému	14
4.1	Diagram případů užití.....	14
4.2	Detaily případů užití.....	17
4.3	E-R diagram	19
4.4	Návrh databáze a její tabulky	21
5.	Rozhraní aplikace.....	22
5.1	Uživatelská část.....	22

5.1.1	Neregistrovaný zákazník.....	22
5.1.2	Registrovaný zákazník.....	23
5.2	Administrační část.....	26
5.2.1	Administrátor	27
5.2.2	Skladník, Expeditor, Reklamační oddělení.....	27
5.2.3	Účetní.....	28
6.	Implementace	29
6.1	Popis výsledného IS	29
6.2	Instalace systému.....	29
6.3	Dolování víceúrovňových asociačních pravidel.....	30
7.	Testování algoritmu	31
7.1	Test algoritmu pro kategorie zboží.....	31
7.2	Test algoritmu pro zboží	33
8.	Závěr	36

1 Úvod

Informační systémy jako takové jsou v dnešní době velice aktuálním tématem. Se statickými webovými prezentacemi, které návštěvníkovi pouze „ukazují“ jistou informaci a jejichž změna znamená zásah do zdrojového kódu, se setkáváme čím dál tím méně často. Do popředí se derou dynamické webové aplikace, které nabízí běžnému uživateli velké množství funkcí a prostoru pro jistou úroveň manipulace s daty, a firmám poskytují obrovské možnosti, jak zdokonalit a zefektivnit svou vlastní produkci a jak se prosadit na trhu a být v „něčem“ lepší než konkurence.

Práce se nezabývá pouze samotným informačním systémem, jeho rozhraním a funkcí. Hlavním úkolem bude využití víceúrovňových asociačních pravidel v prostředí implementovaného internetového obchodu.

Aplikace je navržena pro uživatele v co nejprívětivější formě tak, aby intuitivně vedla uživatele a co nejvíce mu ulehčila práci s informačním systémem. Administrační rozhraní sloužící pro modifikaci obsahu a sledování statistik je implementováno tak, aby při jeho používání nebylo vůbec nutné zasahovat do zdrojových kódů.

Bakalářská práce je rozdělena do několika důležitých kapitol a tématicky souvisejících podkapitol. První kapitolou je samozřejmě tato úvodní část.

Ve 2. kapitole se podrobněji zabývám analýzou požadavků na informační systém, kde bude čtenář seznámen se všemi funkcemi a možnostmi navrhované aplikace.

3. kapitola obsahuje popis technologií použitých při vývoji aplikace – pro implementaci jsem zvolil framework Nette a dále jsou zde také zmíněny technologie jako HTML, Latte, CSS, Javascript, PHP a MySQL.

Kapitola 4 analyzuje navržené řešení, obsahuje schéma návrhu databáze a další, pro návrh aplikace nezbytné, diagramy (Use Case a Entity Relationship diagram).

Uživatelské a administrační rozhraní jsou popsány a na praktických ukázkách zobrazeny v kapitole č. 5.

Šestá kapitola se zabývá samotnou implementací, 7. kapitola testováním a závěrečná kapitola je věnována vyhodnocení výsledků práce a možnostem jejího rozšíření.

2 Analýza požadavků na IS

Jak již bylo zmíněno – hlavním úkolem práce je využití víceúrovňových asociačních pravidel v prostředí elektronického obchodu. V této kapitole se zaměřím na výčet a popis požadavků, vlastností a funkcí, kterými bude tento systém disponovat.

2.1 Dolování užitečných dat

Nejzajímavější částí a funkcionalitou internetového obchodu bude získávání užitečných dat z uskutečněných objednávek pomocí víceúrovňových asociačních pravidel a jejich interpretace do jistého vzorce, který bude při objednávce určitého zboží automaticky nabízet uživateli zboží „přidružené“, tedy takové, které se s daným zbožím často objednává zároveň. Dolování dat a asociačním pravidlům se blíže věnuji ve 4. kapitole.

2.2 Vedení skladu zboží

U zboží, které bude uloženo na skladě, bude evidováno jeho přesné umístění a počet kusů, který právě ve skladu je. Zboží může být objednáno od několika různých dodavatelů, kteří zboží dodají v různých dlouhých dobách dodání – proto bude u zboží dále zaznamenán jeho stav – zda je nebo není na skladě, a za kolik dní bude naskladněno, pokud na skladě chybí. Tato aktualizace bude prováděna automaticky podle aktuálního počtu zboží na skladě a počtu dní, za který je dodavatel schopen zboží dodat.

2.3 Správa uživatelů a zaměstnaneckých rolí

Pro každého uživatele bude samozřejmostí možnost vytvoření objednávky, editování nákupního košíku a údajů pro odeslání objednávky. Registrovaný a přihlášený zákazník bude mít navíc k dispozici přehledné rozhraní, kde uvidí veškeré své uskutečněné objednávky a reklamace, jejich stav a obsah a bude mít možnost si vytisknout či uložit exportovanou fakturu nebo reklamační protokol ve formátu PDF.

Co se týče zaměstnanců, tak každému je v systému přidělena určitá role (administrátor, skladník, účetní,...) a podle této role bude mít přístupnou různou úroveň administrace – každý zaměstnanec bude mít právo dělat jen určité úkony (objednávat nové zboží, vyřizovat zaplacené objednávky, přidávat nové kategorie zboží,...).

2.4 Automatizace zpracování objednávek

Jedna z důležitých vlastností dobrého elektronického obchodu je automatizace celého procesu zpracování objednávek.

V následujících bodech budou shrnuty operace, které jsou potřebné pro plně automatický průběh vyřízení objednávky:

- Vygenerování e-mailu o provedené objednávce zákazníkovi a příslušnému zaměstnanci, který bude mít objednávku na starosti
- Vygenerování platebních údajů
- Vygenerování faktury do PDF
- Parsování e-mailu z banky (kontrola zaplacení objednávky)
- Automatické zrušení objednávky po nezaplacení do 30 dní
- Automatická změna stavu objednávky po zaplacení
- Automatická notifikace zákazníkovi, že bude jeho objednávka expedována
- Automatické upozornění příslušného zaměstnance o možnosti expedovat zásilku

2.5 Vedení pohledávek, uskutečněných plateb a daňová evidence

Systém bude uchovávat a evidovat pohledávky a uskutečněné platby za objednané zboží. Z těchto dat bude dále počítat užitečné statistiky a staré pohledávky (po 30ti dnech nezaplacené objednávky) bude automaticky stornovat a mazat. Součástí systému bude také daňová evidence ve své základní podobě – pro potřeby neplátce daně.

2.6 Backend statistiky prodeje

Další nedílnou součástí systému a užitečnou pomůckou pro provozovatele elektronického obchodu jsou různé statistiky vypovídající o úspěšnosti/neúspěšnosti firmy. Dalším přínosem těchto statistik může být přehodnocení vedení obchodu po jejich vyhodnocení. Implementovaný informační systém počítá následující statistiky:

- Tržby za určité období
- Nejprodávanější zboží
- Počet objednávek/prodaného zboží
- Počet reklamací
- Nejreklamovanější zboží
- Nejméně reklamovanější zboží
- Nejlepší dodavatel (podle počtu reklamací, objednávek, doby dodání)

3 Použité technologie

Implementovaný informační systém musí být dostupný všem uživatelům Internetu přes webový prohlížeč, a to bez jakékoliv další instalace nějakého softwaru. Jelikož bude uživatelský počítač zastávat pouze funkci zobrazovacího zařízení a veškeré činnosti budou vykonávány na straně serveru, tak pro implementaci bude použito skriptovacího jazyku komunikujícího s databázovým systémem.

Při implementaci byly použity následující technologie:

- Značkovací jazyk HTML/Latte v kombinaci s kaskádovými styly CSS
- Skriptovací jazyk JavaScript
- Skriptovací jazyk PHP – Nette Framework
- Databázový systém MySQL

3.1 HTML a Latte

HyperText Markup Language, označovaný zkratkou HTML, je programovací jazyk pro tvorbu webových stránek. Tyto stránky mohou být poté prohlíženy kýmkoliv, kdo je připojen k Internetu. Neustále prochází revizí a vývojem, aby splňoval požadavky na rostoucí návštěvnost internetu pod vedením W3C – organizací pověřené návrhem a vývojem jazyka.

Význam zkratky:

- Hypertext – je metoda, jakou se pohybujeme v Internetu, a to prostým kliknutím na odkaz (speciální text také zvaný hyperlink)
- Markup – HTML značkuje text dokumentu jako jistý typ textu (např. kurzíva)
- Language – jelikož má HTML kódová slova a syntaxi, tak se jedná o jazyk

HTML sestává z množiny značek (tzv. tagů) – tyto značky mají své atributy a jsou vkládány do textu dokumentu, tím určují jeho význam a také podobu. Jednotlivé značky se uzavírají do úhlových závorek `< a >` a ve většině případů jsou párové. Společně s textem tvoří značky element, jehož vlastnosti jsou upřesněny pomocí jeho atributů. Tagy jako takové se na prohlížečem zobrazené stránce neobjeví, ale projeví se na jejím vzhledu. Mezi nejjednodušší tagy patří ty, které formátují text – např.:

`<b>`Tyto slova budou tučně`</b>`, a tyto ne.

Poté je text uložen jako HTML soubor a jeho obsah může být zobrazen pomocí webového prohlížeče, mezi které patří Internet Explorer, Mozilla Firefox nebo Opera. Tento prohlížeč poté přečte soubor a převádí text do uživateli srozumitelné podoby. Vývoj samotného HTML byl ovlivněn také vývojem jednotlivých prohlížečů. [1]

Latte (Language for Transforming Text) je jednoduchý a výkonný jazyk pro vkládání značek do textových dokumentů. Nesmírnou výhodou tohoto jazyku oproti HTML je jednoduchost jeho syntaxe a snadná editace.

Sám vynálezce HTML řekl, že nebylo původně zamýšleno, aby tento jazyk byl spatřen lidmi, natož tak, aby ho sami používali. Původní záměr byl, že bude generován pomocí WYSIWYG editorů nebo při editování za pomoci jazyků vyšší úrovně – např. Latte.

Dokumenty psané v Latte mohou být převedeny do vysoce kvalitního HTML kódu, což z něj dělá lepší jazyk pro psaní World Wide Web dokumentů. [2]

3.2 CSS

Kaskádové styly (Cascading Style Sheets) se používají pro formátování vzhledu webových stránek. Jsou používány například pro definování stylu textu, velikosti tabulky, pozadí stránky a dalších aspektů, které mohly být dříve definovány pouze v kódu HTML stránky. Hlavním cílem je tedy umožnit návrhářům oddělit vzhled dokumentu od jeho struktury a obsahu. Zatím byly vydány dvě specifikace – a to CSS1 a CSS2, nyní se již pracuje na verzi CSS3.

CSS dovoluje vývojářům vytvořit jednotný vzhled a tuto šablonu použít pro více stránek. Namísto toho, abychom definovali styl každé tabulky nebo bloku textu v rámci HTML, tak použijeme nadefinovaný styl, který stačí nadefinovat pouze jednou v rámci CSS. Takto nadefinovaný styl může poté použít jakákoliv stránka odkazující se na něj. S tím souvisí další výhoda – v situaci, kdy potřebujeme změnit např. velikost písma na 50ti stránkách používajících stejný styl, nemusíme je upravovat jednu po druhé, ale stačí upravit jen jeden CSS soubor.

Kaskádové styly nejsou dobré pouze pro vytváření textových stylů, ale jsou také užitečné pro formátování dalších aspektů rozložení webové stránky. Například mohou být použity pro definování odsazení buněk tabulky, styl, tloušťku a barvu rámečku tabulky, obtékání obrázku a další. CSS poskytuje vyšší úroveň kontroly nad tím, jak bude stránka vypadat, než poskytovalo HTML.

Mezi další nesporné výhody můžeme začlenit možnost cachování stylů, kdy si prohlížeč načte soubor se styly do cache paměti, a tak může být dosaženo rychlejšího načtení stránky. Dále také můžeme CSS vlastnosti jednotlivých elementů dynamicky měnit pomocí Javascriptu.

V době, kdy můžeme na Internet přistupovat pomocí chytrých telefonů a dalších zařízení, nám kaskádové styly umožňují nadefinování rozdílného vzhledu stránky pro různá zařízení a tento styl na příslušném zařízení použít a tím zpříjemnit její prohlížení přizpůsobením např. velikosti písma vzhledem k velikosti displeje. [3]

3.3 JavaScript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk, používaný v prostředí Internetu, umožňující dynamické ovládání interaktivních prvků grafického uživatelského rozhraní (např. tlačítka) nebo vytvoření různých efektů a animací.

Jedná se o klientský skript, často vkládaný přímo do HTML kódu – jeho vykonání se tedy provádí až po tom, co je odeslán společně se stránkou na klientský počítač (do prohlížeče). Oproti tomu serverové skripty jsou vykonávány na serveru a na klienta se posílají už jen výsledky.

Javascript je často chybně zaměňován s Javou, což je samostatný programovací jazyk. Oba dva jazyky spojuje vedle podobného názvu už jen podobná syntaxe.

Charakteristika jazyka:

- interpretovaný
- objektový - využívá objektů prohlížeče a zabudovaných objektů
- závislý na prohlížeči
- syntaxí podobný jazykům C, Java a podobným [4]

3.4 Skriptovací jazyk PHP

PHP (oficiálně známý jako PHP: Hypertext Preprocessor, původně Personal Home Page) je skriptovací programovací jazyk, jehož kód je prováděn na straně serveru.

PHP kód může žádat záznamy z databáze, vytvářet obrázky, číst nebo zapisovat do souboru nebo také komunikovat se vzdálenými servery a přistupovat k většině databázovým systémům (např. MySQL, Oracle, ODBC,...). Výsledek PHP skriptu je poslán v kombinaci s HTML do klientského webového prohlížeče, aniž by uživatel poznal že webový server použil PHP – protože prohlížeč vidí pouze HTML.

Podpora Apache – jednoho z nejvíce používaných webových serverů na světě a také podpora MySQL – mocné, volně dostupné SQL databáze, pro kterou PHP poskytuje nepřehledné množství funkcí pro práci s ní, z něj dělá jeden z nejpopulárnějších volně dostupných jazyků. Pro kombinaci s MySQL a serverem Apache se často setkáváme se zkratkou LAMP – Linux, Apache, MySQL, PHP, Perl nebo Python.

Další výhodou je možnost použití PHP na nejznámějších operačních systémech (Microsoft Windows a Linux). Jelikož je PHP kód zapsán přímo do kódu HTML stránky, není pro jeho používání potřeba dalšího vývojového prostředí. Blok kódu začíná znaky `<?php` a končí `?>` (nebo také `<SCRIPT LANGUAGE="php"></SCRIPT>`). Syntaxe jazyku vychází především z programovacích jazyků C a Perl. Pro použití proměnných není nutná předešlá deklarace. [5]

3.5 MySQL

MySQL je relační databázový systém typu DBMS (Database Management System), vlastněný společností Oracle. Jedná se o multiplatformní databázi, která je tvořena z jedné nebo více tabulek – ty nesou jednotlivé řádky (záznamy) a sloupce mající jméno a uvozující datový typ jednotlivých polí záznamu.

Práce s daty v databázi se provádí pomocí příkazů, respektive dotazů vycházejících z deklarativního programovacího jazyka SQL (Structured Query Language).

Systém MySQL je šířen jako open source a lze jej instalovat na operační systémy Linux, MS Windows a další. Systém je využitelný v jazycích C, C++, Java, PHP, Python a další. [6]

3.6 Nette Framework

Nette Framework je výkonný framework využívající událostmi řízené programování pro tvorbu webových aplikací v PHP 5, který je založen z velké části na použití komponent. Nette Framework je svobodný software, který lze použít i v komerčních projektech. Jeho autorem je David Grudl. [7]

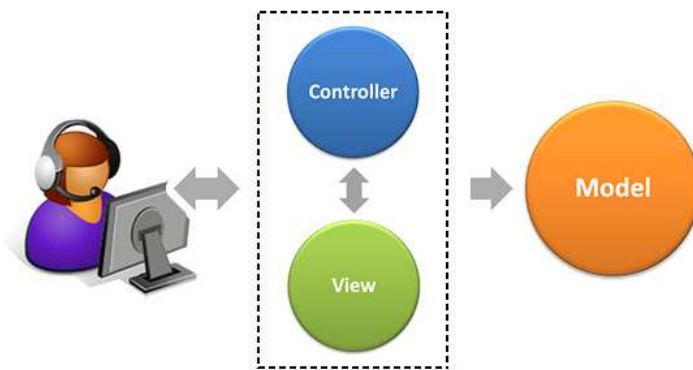
Hlavní přednosti:

- Dokonalé zabezpečení - používá revoluční technologii, která eliminuje výskyt bezpečnostních děr a jejich zneužití, jako je např. XSS, CSRF, session hijacking, session fixation atd.
- Vyzrálý objektový návrh - promyšlený a čistý objektový návrh využívající nových vlastností PHP 5, komponent a událostmi řízeného modelování.
- Open-source licence – framework lze zdarma používat i v komerčních projektech.
- Strmá křivka učení
- Práce v týmu - výborně rozděluje práci mezi více programátorů a HTML kodérů. [8]

3.6.1 Model – View – Controller (MVC)

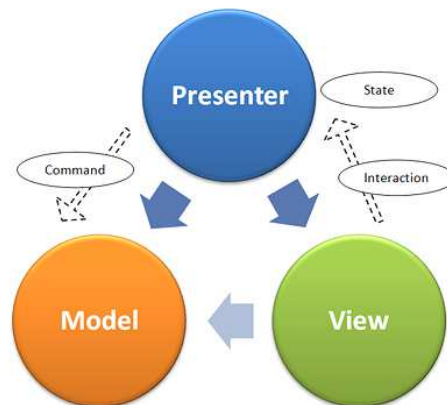
Architektura MVC je řešením pro aplikace, které byly v dřívějších dobách navrhovány tak, že bylo uživatelské rozhraní spojeno s logikou aplikace v jednotlivý celek. Dodnes tento přístup používá například Delphi.

Tento přístup se však ukazuje jako neefektivní, když aplikace získává na složitosti – kvůli přílišné provázanosti je úprava kódu velice komplikovaná. Řešením je oddělení kódu aplikační logiky, označovaného jako **model**, od uživatelského rozhraní. Uživatelské rozhraní může být dále rozděleno na vrstvu zpracovávající vstupy od uživatele (**controller**) a na vrstvu obstarávající výstup na monitor (**view**). Ta prezentuje uživateli data získaná od modelu. Jakmile na ně uživatel zareaguje, controller tuto reakci zpracuje a doručí modelu.



Obr. 3.1: MVC [9]

3.6.2 Model – View – Presenter (MVP)



Obr. 3.2: MVP [9]

V Nette Frameworku se namísto controlleru setkáme s **presenterem**. Ten má podobnou roli – přímo komunikuje s modelem a předává získaná data do vybraného pohledu (view) a především zpracovává následující reakce uživatele:

- změna pohledu
- změna stavu
- příkaz modelu [9]

3.7 Data mining

Tato podkapitola čerpá z [10].

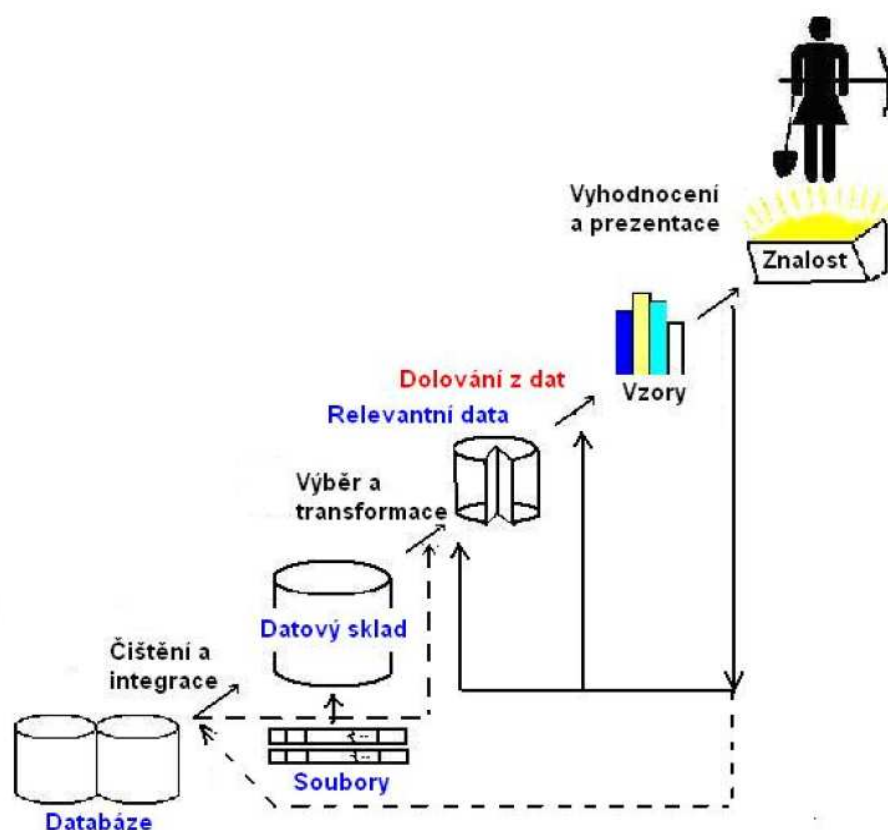
Data mining (angl. dolování z dat či získávání znalostí z databází) je metodologie extrakce netriviálních, skrytých a potenciálně užitečných informací a vzorů z dat. Netriviálnost a skrytost dat můžeme chápat jako informace, které nejsou na první pohled v datech vidět a musíme je získat jiným způsobem, než například nějakým jednoduchým SQL dotazem. Potenciální užitečnost můžeme chápat např. jako základ pro rozhodnutí, které zboží zákazníkovi ještě k jeho objednanému zboží nabídneme na základě toho, jaké produkty si často lidé kupují dohromady (např. tiskárna a příslušné náplně do ní).

Dolování dat se používá v mnoha oblastech. Ať už je to například při analýze trhu a marketingu, analýze nákupního košíku nebo při monitorování aktivit na internetu (zvláštěností je, že zde se snažíme najít takové vzory, které se zpravidla výrazně odlišují od častých vzorů – např. vyhledávání neobvyklých bankovních transakcí).

Získávání znalostí z databází můžeme chápat jako jistý proces tvořený řadou kroků, které se zpravidla v určitých iteracích opakují:

1. Čištění dat – vypořádání se s chybějícími daty, odstranění šumu.
2. Integrace dat – integrace dat z několika různých datových zdrojů.

3. Výběr dat – výsledkem jsou data, která jsou relevantní a užitečná pro řešení dané analytické úlohy.
4. Transformace dat – převod dat do konsolidované podoby – například sumarizací nebo agregací.
5. Dolování dat – v této fázi dochází k aplikaci určité metody a algoritmu s cílem extrahovat z dat vzory.
6. Hodnocení modelů a vzorů – identifikace užitečných vzorů a modelů.
7. Prezentace znalostí – reprezentace výsledků dolování uživateli.



Obr. 3.3: Proces dolování dat [10]

3.8 Dolování asociačních pravidel a frekventovaných množin

Tato podkapitola a zároveň její podkapitoly čerpají z [10].

Získáním asociačních pravidel získáme zajímavé asociace nad velkým množstvím datových položek. Internetový obchod je typickým prostředím, kde nalezení zajímavých asociací nad obchodními objednávkami může napomoci v mnoha aspektech obchodování.

Příkladem může být analýza nákupního košíku, kdy analyzujeme chování zákazníka, sledujeme jaké zboží nejčastěji nakupuje dohromady a získané znalosti uplatníme při návrhu akčních nabídek, katalogů nebo rozmístění produktů v obchodě.

3.8.1 Jednoúrovňová asociační pravidla

Jedná se o nejjednodušší variantu asociačních pravidel. I když smyslem této práce je využití víceúrovňových asociačních pravidel, zmínka o pravidlech jednoúrovňových je nezbytnou součástí, jelikož se z nich vychází a pro získání víceúrovňových pravidel je použito stejného algoritmu.

Tímto algoritmem je algoritmus Apriori, který slouží pro získávání frekventovaných množin. Jak název napovídá, algoritmus využívá znalostí o frekventovaných množinách získaných v předchozích iteracích algoritmu. Pro generování $(k+1)$ -množin jsou v každé iteraci použity předchozí frekventované k -množiny.

Pro zvýšení efektivity se využívá tzv. Apriori vlastnosti - každá podmnožina frekventované množiny musí být také frekventovaná.

Proces získání frekventovaných množin se skládá ze dvou kroků, spojovacího a vylučovacího:

- **Spojovací krok**

Spojením množin z L_{k-1} získáme kandidáty na frekventované množiny (C_k). Z těchto kandidátů algoritmus v dalším kroku odstraňuje množiny nefrekventované a získává L_k (všechny frekventované k -množiny). Předpokladem pro správnou funkčnost algoritmu je, že položky v množině jsou lexikograficky seřazeny. Pro spojení dvou $(k-1)$ -množin l_1 a l_2 se musí prvních $k-2$ prvků shodovat. Výsledná množina bude obsahovat prvky $l_1[1], l_1[2], \dots, l_1[k-1], l_2[k-1]$, kde $l_1[i]$ je i -tý prvek množiny l_1 a musí platit, že $l_1[k-1] < l_2[k-1]$.

- **Vylučovací krok**

V tomto kroku odstraňuje algoritmus z C_k všechny nefrekventované množiny. Jelikož by bylo zjišťování výskytů kandidáta v databázi neefektivní, využíváme Apriori vlastnost. Součástí frekventované k -množiny nemůže být nefrekventovaná $(k-1)$ -množina.

3.8.2 Víceúrovňová asociační pravidla

Jelikož může být u některých aplikací složité nalézt zajímavé asociace na nejnižší úrovni, mohou asociace na vyšší úrovni získané pomocí víceúrovňových asociačních pravidel představovat důležitou znalost.

Základním přístupem je procházení hierarchií shora dolů, kdy použijeme algoritmus Apriori pro každou úroveň. Jelikož by byl tento postup neefektivní, používají se následující variace:

- **Použití stejné minimální podpory pro všechny úrovně**

Jde o jednoduchou metodu, která má však i své nevýhody. Položky na nejnižší úrovni se v databázi nevyskytují tak často, jako položky na vyšších úrovních - dochází tedy ke ztrátě zajímavých asociací na nejnižší úrovni

- **Použití redukované minimální podpory na nižších úrovních**

Hodnota minimální podpory se pro každou úroveň liší. Pro zvýšení efektivity této varianty existují následující strategie:

- o **Nezávisle na úrovních** – testuje se každý uzel, bez ohledu na to, zda je jeho otcovských uzel frekventován.
- o **Filtrování úrovní jednou položkou** – Teprve pokud je otcovský uzel frekventován, potom je položka na i-té úrovni testována.
- o **Filtrování úrovní k-množinami** – Teprve pokud je odpovídající množina na vyšší úrovni frekventována, potom je množina na i-té úrovni testována.

3.8.3 Generování asociačních pravidel z frekventovaných množin

Pro generování asociačních pravidel se využívá rovnice pro výpočet spolehlivosti:

$$conf(A \Rightarrow B) = P(B | A) = \frac{s(A \cup B)}{s(A)} \quad (3.1)$$

Generování se provádí v následujících krocích (podmínka minimální podpory není v krocích zohledněna, jelikož je automaticky splněna tím, že jsou pravidla generována z frekventovaných množin):

- Vygeneruj všechny neprázdné podmnožiny s frekventované množiny l .
- Pro všechny podmnožiny vygeneruj pravidlo $s \Rightarrow (l-s)$ a pomocí rovnice zjisti jeho spolehlivost. Pravidlo je silné, pokud je spolehlivost vyšší než minimální.

4 Návrh systému

Poté, co nám analýza požadavků dala určitou představu o tom, jak by měl systém vypadat, přicházíme před samotnou implementací k návrhu – asi jedné z nejdůležitějších částí práce. Jelikož špatně navržený systém nelze dobře a kvalitně implementovat, měla by se mu při vývoji věnovat zvýšená pozornost.

4.1 Diagram případů užití

Diagramy případů užití (anglicky use case diagrams) jsou diagramy chování popisující sadu operací (možností užití), které nějaký systém nebo systémy (subjekty) mohou ve spolupráci s jeho externími uživateli (účastníky) vykonat – popisují chování systému. Každý případ užití by měl uživateli poskytnout smysluplné a hodnotné výsledky.

Specifikace UML 2.4 popisuje use case diagramy také jako specializaci diagramů tříd, které jsou diagramy strukturovanými.

Diagramy užití se používají ke specifikaci externích požadavků na systém a jeho používání, ke specifikaci funkcionality nabízené systémem a k definování spolupráce prostředí se subjektem tak, aby byl schopen plnit své služby.

Elementy případu užití:

- Aktér/účastník – externí objekt, který je v interakci se systémem. Je značen figurou a může to být např. zákazník, administrátor,...
- Subjekt/systém – např. společnost, softwarový nebo fyzický systém nebo nějaké zařízení
- Příklad užití – popisuje funkcionalitu, kterou systém nabízí účastníkům (např. vytvořit objednávku, zobrazit reklamace,...). Ta je v diagramu značena oválně a vždy musí být dokončena, aby byl případ považován za hotový – to znamená, že po jeho provedení již nebude očekávat žádné vstupy a že může být znovu vykonán nebo bude v chybovém stavu. [11]

Účastníky tohoto modelu jsou:

- Neregistrovaný zákazník – logicky bude tomuto zákazníkovi nabídnuta omezená funkčnost. Bude moci prohlížet nabízené zboží, vytvořit objednávku a editovat nákupní košík. Pro dokončení objednávky se bude muset zákazník zaregistrovat.
- Registrovaný zákazník – samozřejmě je stejná funkčnost jako je nabízená neregistrovaným zákazníkům. Dále může registrovaný a přihlášený uživatel prohlížet provedené objednávky nebo reklamace a stornovat doposud nezaplacenou objednávku. Dále si má zákazník možnost vytisknout fakturu nebo reklamační protokol, který je skriptem vygenerován do PDF souboru. Samozřejmě je také možnost změny hesla a osobních údajů zadaných při registraci.
- Registrovaný zákazník má také tu výhodu, že pokud si do nákupního košíku vloží zboží, které by chtěl objednat a poté se na stránky obchodu vrátí například až za týden, položky v košíku budou stále uchovány.

- Administrátor – účastník s největšími pravomocemi – spravuje databázi uživatelů, zaměstnanců, zboží, kategorií, dodavatelů...
- Skladník – zaměstnanci zastávajícímu tuto roli bude umožněno prohlížet vytvořené objednávky a reklamace a upravovat doposud nevyřízené objednávky. Další funkcí skladníka je objednávání zboží, jeho přidávání do databáze a správa skladovacích prostor.
- Expeditor – aktér modelu, který se stará o odeslání objednávek a vyřízených reklamací
- Reklamační – člověk starající se o vytvoření a vyřízení reklamací a umožnění jejich expedice
- Účetní – ač je tento aktér poměrně v pozadí, tak je velice důležitou osobou pro chod obchodu. Jeho náplní je kontrola plateb, pohledávek a celkově daňové evidence

Všem zaměstnancům je samozřejmě povoleno si vypsat veškeré údaje uložené v databázi, prohlížet statistiky a vyřízené objednávky a reklamace. Jejich editování a mazání je již povoleno jen zaměstnancům na určitých pozicích.



Obr. 4.1: Diagram případů užití

4.2 Detaily případů užití

Po vytvoření diagramu případů užití, který je velmi stručný, je nutné provést jejich specifikaci. Každá tato specifikace má své vstupní a spouštějící podmínky, výčet účastníků, tok událostí, který popisuje jednotlivé kroky a následné podmínky omezující stav systému po skončení případu užití. Pro příklad a názornost uvádím tři detaily:

Identifiká-	UC01		
Název	Prohlížení objednávek/reklamací		
Popis	Zákazník Z si nechá zobrazit nim provedené objednávky a reklamace		
Priorita	I= nízká	Frekvence	Obvykle jednou měsíčně
Vstupní podmínky	Zákazník Z je přihlášen do systému.		
Výstupní podmínky	Systém zobrazil objednávky a reklamace provedené daným uživatelem.		
Uživatelé	Zákazník Z		
Základní posloupnost	Krok	Činnost	
	1	Případ užití začíná volbou „Moje objednávky/Moje reklamace“.	
	2a	Systém automaticky vyřadí objednávky a reklamace, které neprovedl Z.	
	2b	Když Z neprovedl žádné objednávky ani reklamace: 2b.1 Systém upozorní Z na tuto skutečnost. 2b.2 Případ užití končí.	
	2c	Systém zobrazí seznam objednávek/reklamací.	
	3	Z si může zobrazit detaily určité objednávky/reklamace	
Alternativní posloupnost	Krok	Činnost	
	2 až 3	Z se může kdykoliv vrátit do hlavní nabídky programu/opustit systém.	
Poznámky			

Obr. 4.2: Detail případů užití 1

Identifikátor	UC02		
Název	Editace/prohlížení košíku		
Popis	Zákazník Z si zobrazí obsah nákupního košíku/upraví jeho obsah		
Priorita	3= vysoká	Frekvence	Několikrát denně
Vstupní podmínky	Zákazník Z je přihlášen do systému nebo vystupuje v roli zákazníka guest .		
Výstupní podmínky	Systém zobrazil/změnil obsah nákupního košíku		
Uživatelé	Zákazník Z		
Základní posloupnost	Krok	Činnost	
	1	Případ užití začíná volbou „Zobrazit obsah košíku“.	
	2a	Když je košík prázdný: 2.1 Systém oznámí Z tuto skutečnost 2.2 Případ užití končí	
	2b	Systém zobrazí seznam zboží v nákupním košíku Z.	
	3	Z může zboží z košíku odebrat, nebo změnit počet objednaných kusů a aktualizovat obsah košíku	
	4	Byl-li obsah nákupního košíku v kroku 3 změněn, systém si tuto změnu uloží.	
	5	Systém zobrazí aktualizovaný obsah košíku.	
Alternativní posloupnost	Krok	Činnost	
	2 až 5	Z se může kdykoliv vrátit do hlavní nabídky programu/opustit systém.	
Poznámky			

Obr. 4.3: Detail případů užití 2

Identifikátor	UC03		
Název	Editovat Zboží		
Popis	Zaměstnanec Z edituje určité zboží z databáze.		
Priorita	2=střední	Frekvence	Několikrát týdně
Vstupní podmínky	Zaměstnanec Z je přihlášen do systému.		
Výstupní podmínky	Vybrané zboží je upraveno a změny jsou uloženy v databázi.		
Uživatelé	Zaměstnanec Z		
Základní posloupnost	Krok	Činnost	
	1	Případ užití začíná volbou „Editovat zboží“.	
	2a	Systém zjistí zda má Zaměstnanec Z práva k tomuto úkonu.	
	2b	Když není Zaměstnanec Z v požadované roli: 2b.1 Systém mu oznámí, že k tomuto úkonu nemá oprávnění. 2b.2 Případ užití končí	
	2c	Zaměstnanec Z může upravit informace o zboží, změnit dodavatele, uskladnění, nebo nahradit nový obrázek zboží.	
	3	Po volbě „Uložit“ systém uloží změny do databáze a ohlásí uživateli, že bylo zboží upraveno.	
Alternativní posloupnost	Krok	Činnost	
	2	Z se může vrátit do hlavního menu programu/opustit systém, aniž by zboží upravil.	
Poznámky			

Obr. 4.4: Detail případů užití 3

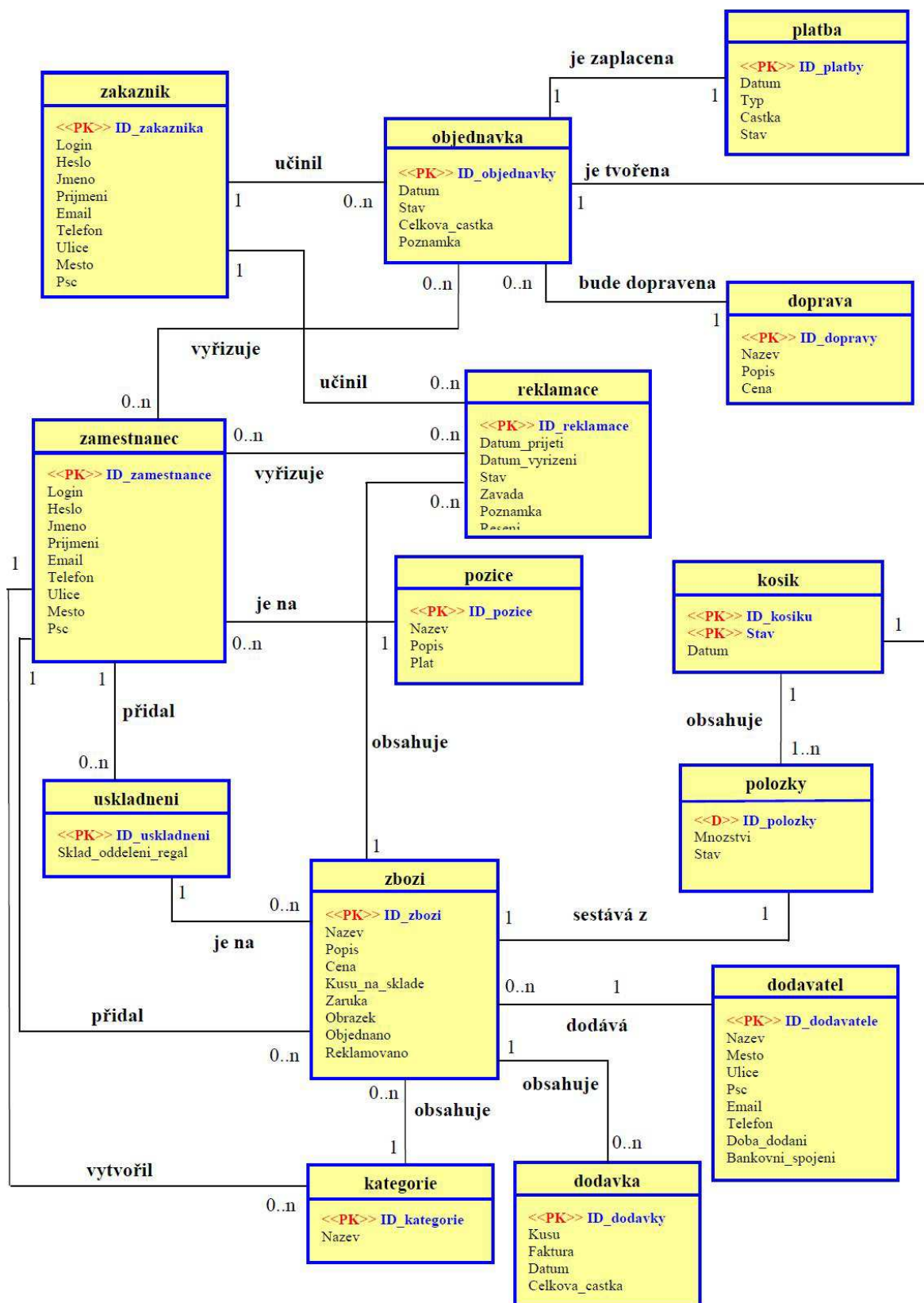
4.3 E-R diagram

Jedním ze základních modelů strukturované analýzy je datové modelování, které zobrazuje entity aplikační domény zpracovávané systémem a statické vztahy mezi nimi (jedná se typicky o perzistentní data uložená v databázi). Toto modelování má zásadní význam pro návrh databáze a jeho výstupem je jeden z typů konceptuálních schémat, a to diagram entit a vztahů (angl. Entity Relationship Diagram – ERD).

Jak již bylo řečeno, ER model slouží k modelování dat a jejich vztahů „v klidu“. Řeší otázky jako: Která data potřebujeme v systému uchovávat? Jaké jsou mezi nimi vztahy? Jedná se o síťový model popisující návrh uložených dat v systému na vyšší úrovni abstrakce.

Mezi základní pojmy datového modelování patří:

- Entita – věc reálného světa (objekt), např. zákazník e-shopu s přihlašovacím jménem xkrali15
- Entitní množina – množina entit stejného typu, např. Zákazník
- Atribut – vlastnost entity, např. jméno a příjmení zákazníka
- Vztah – příslušný vztah mezi entitami, např. zákazník Z si objednal zboží s identifikačním číslem 33
- Vztahová množina – množina vztahů téhož typu, např. Zákazník si objednal Zboží [12]



Obr. 4.5: E-R diagram

4.4 Návrh databáze a její tabulky

Samotný návrh vychází z příloženého E-R diagramu. Tabulky navíc budou obsahovat také cizí klíče z tabulek, se kterými jsou ve vztahu. Pro přehled uvádím výpis všech tabulek společně s datovými typy jednotlivých sloupců:

zakaznik ID_zakaznika: Int(10) Login: Varchar(20) Heslo: Varchar(200) Jmeno: Varchar(20) Prijmeni: Varchar(20) Email: Varchar(50) Telefon: Int(9) Ulice: Varchar(50) Mesto: Varchar(30) Psc: Int(5)	zamestnanec ID_zamestnanec: Int(10) ID_pozice: Int(10) Login: Varchar(20) Heslo: Varchar(200) Jmeno: Varchar(20) Prijmeni: Varchar(20) Email: Varchar(50) Telefon: Int(9) Ulice: Varchar(50) Mesto: Varchar(30) Psc: Int(5)	zbozi ID_zbozi: Int(10) ID_kategorie: Int(10) ID_dodavatele: Int(10) ID_uskladneni: Int(10) Nazev: Varchar(50) Popis: Varchar(500) Cena: Int(6) Kusu_na_sklade: Int(5) Zaruka: Int(2) Obrazek: Varchar(100) Objednано: Int(9) Reklamovano: Int(9)
pozice ID_pozice: Int(10) Nazev: Varchar(30) Popis: Varchar(200) Plat: Int(9)	kosik ID_kosiku: Varchar(200) Stav: Int(1) ID_zakaznika: Int(10) Datum: Date	dodavatel ID_dodavatele: Int(10) Nazev: Varchar(30) Mesto: Varchar(30) Ulice: Varchar(50) Psc: Int(5) Email: Varchar(50) Telefon: Int(9) Doba_dodani: Int(2) Bankovni_spojeni: Varchar(20)
kategorie ID_kategorie: Int(10) Nazev: Varchar(30)	polozky ID_polozky: Int(10) ID_kosiku: Varchar(200) ID_zbozi: Int(10) Mnozstvi: Int(3) Stav: Varchar(40)	
doprava ID_dopravy: Int(10) Nazev: Varchar(50) Popis: Varchar(500) Cena: Int(3)	objednavka ID_objednavky: Int(10) ID_zakaznika: Int(10) ID_zamestnanec: Int(10) ID_platby: Int(10) ID_dopravy: Int(10) ID_kosiku: Varchar(200) Datum: Date Stav: Varchar(50) Celkova_castka: Int(9) Poznamka: Varchar(200)	platba ID_platby: Int(10) Datum: Date Typ: Varchar(20) Cena: Int(9) Stav: Varchar(20)
dodavka ID_dodavky: Int(10) ID_zbozi: Int(10) Kusu: Int(3) Faktura: Varchar(20) Datum: Date Celkova_castka: Int(9)	reklamace ID_reklamace: Int(10) ID_zakaznika: Int(10) ID_zamestnanec: Int(10) ID_zbozi: Int(10) Datum_prijeti: Date Datum_vyřízení: Date Stav: Varchar(40) Zavada: Varchar(500) Poznamka: Varchar(200) Reseni: Varchar(100)	uskladneni ID_uskladneni: Int(10) Sklad_oddeleni_regal: Varchar(100)

Obr. 4.6: Návrh databáze a její tabulky

5 Rozhraní aplikace

V této kapitole detailněji popíší funkce informačního systému nabízené zákazníkům a administrátorům (viz kapitola 4.1 Diagram případů užití). Pro znázornění doplním teoretický výklad obrázky zobrazujícími uživatelské a administrační rozhraní a znázorňující funkce nabízené systémem.

5.1 Uživatelská část

Jak již bylo jednou řečeno a jak z diagramu případů užití vyplývá, neregistrovaný zákazník má omezené možnosti používání informačního systému, zatímco registrovaný zákazník má k těmto omezeným funkcím k dispozici vlastní uživatelské rozhraní, kde může sledovat své objednávky a reklamacie.

Systém je implementován tak, že pokud není uživatel registrován a přihlášen, vystupuje pod rolí hosta (angl. guest). Aplikace poté v dané chvíli zjišťuje, v jaké je zákazník roli, a podle toho mu zpřístupňuje jednotlivé sekce systému.

5.1.1 Neregistrovaný zákazník

Tento uživatel je omezen pouze na sekce a funkce informačního systému, které nepotřebují jeho osobní údaje.

Při navštívení je uživateli zobrazena uvítací obrazovka se sekci „Aktuality“, dále může zaslat přes kontaktní formulář internetovému obchodu dotaz nebo nějakou připomínku, zobrazit si obchodní podmínky a zboží nabízené obchodem. Má na výběr zobrazit si veškeré zboží nebo jej filtrovat pomocí kategorií a jednotlivých výrobců. Tento výpis zboží může dále filtrovat podle abecedy nebo ceny, a to od nejlevnějšího nebo nejdražšího zboží.



Obr. 5.1: Uživatelské rozhraní – výpis zboží

K dispozici má neregistrovaný návštěvník také detailní výpis zboží zahrnující jeho popis, délku záruky, počet kusů na skladě a cenu. Po volbě „koupit“ je položka zboží vložena do nákupního košíku. Vypsání jeho obsahu a možnost jej editovat je neregistrovanému zákazníkovi také povoleno. Může ho úplně vyprázdnit, mazat jednotlivé položky, nebo upravovat počet kusů již vloženého zboží.



ACER Aspire 5742G-374G75Mnkk

Popis:
Intel Core i3 370M, 15.6" LED 1366x768 lesklý, RAM 4GB, NVIDIA GeForce GT540M 1GB, HDD 750GB, DVD, WiFi, BlueTooth, Webkamera, Windows 7 Home Premium 64-bit
Záruka: 2 roky
Kusů na skladě: 4 - (Na skladě do 3 dnů)

Cena: 12490,- Kč

[Koupit](#)

Obr. 5.2: Detail zboží

Zboží	Množství	Odebrat kus	Přidat kus	Cena	Smazat
Kingston DataTraveler HyperX 64GB	1 ks			3000 Kč	
ACER Aspire 5742G-374G75Mnkk	2 ks			24980 Kč	
				Celkem:	27980 Kč

[<<Zboží](#)
[Vyprázdnit](#)
[Objednat>>](#)

Obr. 5.3: Výpis nákupního košíku

K tomu, aby mohl uživatel zboží objednat, se musí registrovat nebo přihlásit – tato část je tedy mezníkem funkčnosti pro jednotlivé druhy zákazníků. Systém je samozřejmě navržen tak, aby uchovával jednotlivé položky vložené do nákupního košíku po celou dobu nákupu a v případě, kdy chce zákazník zboží objednat a je nucen se registrovat nebo přihlásit, tak poté co tento požadavek splní, je obsah košíku převeden na nového uživatele a je uchováván pod jeho novým ID.



Login:
 Heslo:
 Heslo pro kontrolu:
 Jméno:
 Příjmení:
 E-mail:
 Telefon: +420
 Ulice:
 Město:
 PSČ:
[Registrovat](#) [Cancel](#)

Obr. 5.4: Registrační formulář

5.1.2 Registrovaný zákazník

Po přihlášení je uživateli zpřístupněno jeho vlastní uživatelské rozhraní, pomocí kterého si může změnit své osobní údaje, které uvedl u registrace, nebo změnit heslo pro přístup do systému.



Obr. 5.5: Profil uživatele

Toto rozhraní slouží také ke sledování objednávek a reklamací daného uživatele – jejich stavu, podrobnému výpisu nebo možnosti si vygenerovat fakturu vyřízené objednávky nebo reklamační protokol podané reklamace do PDF souboru a následně je vytisknout.

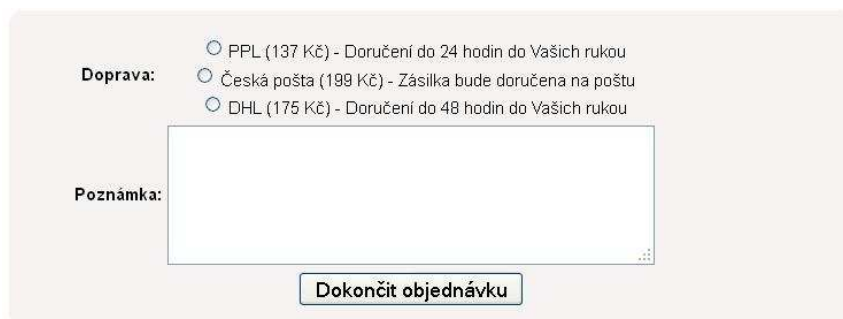
Vaše objednávky:			
Objednávka	Stav	Náhled	Tisk
1	Zpracovává se		
3	Vyřízeno		
4	Vyřízeno		
10	Vyřízeno		
12	Vyřízeno		

Obr. 5.6: Přehled objednávek

Detail objednávky:		č. faktury: 3	
Dodavatel: e-shop a.s. Krátká 25 Brno 61200		Odběratel: Michael Králík Na Sídlišti 691 Zlaté Hory 79376	
Datum objednávky:		2011-11-13	
Stav objednávky:		Vyřízeno	
Objednané zboží:			
Název	Množství	Cena/ks	
24" HP ZR2440w	1	9700 Kč	
Způsob dopravy:	PPL	137 Kč	
		Celkem: 77870 Kč	
Poznámka k objednávce:		Prosím o rychlé vyřízení	
Stav platby:		Zaplaceno	
Datum platby:		2012-01-31	
Děkujeme, že u nás nakupujete.			

Obr. 5.7: Detail objednávky

Jakmile dojde k výběru druhu dopravy zásilky s objednaným zbožím a odsouhlasení finálního výpisu objednávky, kterým uživatel potvrdí obsah objednávky a adresu, na kterou bude dodána, je objednávka uložena do databáze. Od této chvíle může být sledován její stav nebo může být také stornována přes uživatelské rozhraní.



The image shows a user interface for selecting a shipping method. It features a light beige background with a rounded rectangle containing the form elements. On the left, the label "Doprava:" is positioned above three radio button options. The first option is "PPL (137 Kč) - Doručení do 24 hodin do Vašich rukou". The second option is "Česká pošta (199 Kč) - Zásilka bude doručena na poštu". The third option is "DHL (175 Kč) - Doručení do 48 hodin do Vašich rukou". Below these options is a text area labeled "Poznámka:". At the bottom center of the form is a button labeled "Dokončit objednávku".

Doprava:

- ☐ PPL (137 Kč) - Doručení do 24 hodin do Vašich rukou
- ☐ Česká pošta (199 Kč) - Zásilka bude doručena na poštu
- ☐ DHL (175 Kč) - Doručení do 48 hodin do Vašich rukou

Poznámka:

Dokončit objednávku

Obr. 5.8: Výběr dopravce

5.2 Administrační část

Pro aktualizování nabídky obchodu, přidání nových zaměstnanců či zákazníků, vytvoření nové reklamace nebo správu jednotlivých objednávek a reklamací se musí zaměstnanec přihlásit do administrační části.

Tato sekce je přístupná všem zaměstnancům. Každý vystupuje po přihlášení pod určitou rolí (administrátor, skladník, účetní,...) a na základě této role je mu umožněno, resp. zakázáno provádět jednotlivé úkony. Jako příklad uvedu situaci, kdy se bude skladník pokoušet vytvořit novou reklamaci – tento úkon může provádět pouze zaměstnanec pracující na reklamačním oddělení – proto systém zaměstnanci oznámí, že k tomuto úkonu nemá oprávnění a skladníkovi nezbude nic jiného než kontaktovat někoho z příslušného oddělení.

Každý zaměstnanec má samozřejmě právo si po přihlášení změnit své osobní údaje nebo přihlašovací heslo a zobrazovat statistiky. Před úpravou zboží nabízeného obchodem si může zaměstnanec zvolit, zda si nechá vypsát veškeré zboží nebo jen určitou kategorii.

Statistiky **Daňová evidence** **Zkontrolovat došlé platby**

Menu

- ▶ Dodavatelé
- ▶ Doprava
- ▶ Kategorie zboží
- ▶ Pracovní pozice
- ▶ Ukládání
- ▶ Zákazníci
- ▶ Zaměstnanci
- ▶ Zboží
- ▶ Vyřízené objednávky
- ▶ Nevýřízené objednávky
- ▶ Vytvořit reklamaci
- ▶ Vyřízené reklamace
- ▶ Nevýřízené reklamace
- ▶ Dodávky zboží

Zboží

1 | 2 | 3 | ...Další

Zboží	Kusů na skladě	Editovat	Objednat	Smazat
22" ASUS VE228H LED	9			
22" iiyama ProLite B2274HDS černý	7			
23" HP x2301 LED	2			
24" HP ZR2440w	11			
24" Samsung S24A300H černý	3			
A4tech XL-747H Gaming motiv pavouk	11			
ACER Aspire 5742G-374G75Mnkk	4			
ACER Aspire 5750ZG-B958G75Mnrr	0			
Acer Aspire M3900	6			
Acer Veriton M4610G	5			
ASUS K53SV-SX966V	6			
ASUS N75SF-TZ205V	10			
CANON PIXMA MP250	2			
Canon PIXMA MX870	5			
EPSON Stylus SX130	3			

Přihlášen zaměstnanec: admin

Upravit profil
Změna hesla
Odtáhnout

Zboží podle kategorie

- ▶ Flashdisky
- ▶ Klávesnice
- ▶ LCD monitory
- ▶ Myši
- ▶ Notebooky
- ▶ Osobní počítače
- ▶ Tiskárny a skenery

Přidat zboží

Kategorie: Flashdisky
Dodavatel: Express
Ukládání: A/Notebooky/11
Název:
Popis:
Cena(Kč):
Kusů na skladě:
Záruka:
Obrázek: Procházet...
Uložit Cancel

Obr. 5.9: Administrace zboží

5.2.1 Administrátor

Jak je již z názvu této role zřejmé, tento zaměstnanec bude mít nejvyšší úroveň oprávnění. Nejčastějším úkonem administrátora bude přidávání, editování a mazání informací souvisejících se zbožím uloženým v databázi – jako je například zboží samotné, jeho kategorie, dodavatel a uskladnění.

Jako jediný zaměstnanec může vytvářet nové pracovní pozice a vložit do systému nového zákazníka nebo zaměstnance a editovat jejich osobní údaje.



Obr. 5.10: Přidání zákazníka

5.2.2 Skladník, Expeditor, Reklamační oddělení

Tyto role byly definovány pro správu objednávek a reklamací. Zatímco reklamační oddělení může pouze vytvářet a editovat jednotlivé reklamace, skladníková funkce spočívá nejen ve správě objednávek, ale také může editovat a přidávat skladovací prostory a zboží, které bude prostřednictvím administrativní části také objednávat od dodavatelů přes jednoduchý objednávkový formulář.

Ihned po vytvoření objednávky nebo reklamace je zákazník informován e-mailem a při změně jejich stavu je okamžitě znovu informován. To stejné platí pro zaměstnance – po vytvoření objednávky je notifikován jeden ze skladníků, který zboží přichystá k expedici. Jakmile dojde k zaplacení objednávky, je o této skutečnosti informován expeditor a může dojít k jejímu vyřízení a odeslání.

Vyřízené objednávky

1 | 2 | 3 | ...Další

Objednávka	Datum	Stav	Zobrazit	Tisk	Smazat
3	2011-11-13	Vyřízeno			
4	2011-11-13	Vyřízeno			
10	2012-02-12	Vyřízeno			
12	2012-02-12	Vyřízeno			
16	2012-02-20	Vyřízeno			

Nevyřízené reklamace

Reklamační	Datum příjmu	Stav	Editovat	Smazat
2	2012-01-03	Připraveno k odeslání		
5	2012-01-23	Zpracovává se		

Obr. 5.11: Administrace objednávek/reklamací

5.2.3 Účetní

I když má zaměstnanec s touto rolí v podstatě nejnižší oprávnění v administraci obchodu, zastává nesmírně důležitou funkci – má na starosti finanční stránku provozování obchodu.

Jako jediný zaměstnanec má přístup do sekce s daňovou evidencí a přehledem nezaplacených objednávek(pohledávek) společně s výpisem zaplacených objednávek za období od poslední kontroly zaplacení pohledávek.

6 Implementace

K vytvoření informačního systému jsem použil framework Nette společně s databázovou knihovnou Dibi. Generování faktur a reklamačních protokolů je implementováno za pomoci knihovny FPDF. Knihovna lightbox zase umožňuje zobrazovat náhled fotek nabízeného zboží

6.1 Popis výsledného IS

Systém je rozdělen do dvou modulů, a to modul pro zákazníky a modul pro zaměstnance. Každý modul sestává z formulářů (přihlašovací, kontaktní, objednávkový,...), z výchozí šablony obsahující jednotlivá menu, hlavičku a patičku stránky, do které jsou vkládány bloky šablon pro jednotlivé stránky (výpis zboží, objednávek, statistik,...). Vykreslení a naplnění těchto šablon obstarávají presen-tery, které pro získání dat využívají jednotlivých modelů.

6.2 Instalace systému

Jelikož výsledkem práce není aplikace, která by se nainstalovala jedním spustitelným souborem .exe, musí být před začátkem používání splněny následující body:

- **K dispozici musí být webový server s podporou technologie PHP 5.3 s povoleným modulem MySQL pro práci s databází**
- **Vytvoření databáze**
Po vytvoření databáze použijte pro vytvoření jednotlivých tabulek soubor *databaze.sql*, který je k dispozici mezi zdrojovými soubory. SQL skript také naplní jednotlivé tabulky testovacími daty a vytvoří 3 testovací zákazníky a zaměstnance na každé z uvedených pozic.
- **Nastavení konfiguračního souboru pro připojení k databázi**
Tento krok spočívá ve vyplnění údajů pro připojení k používané databázi v souboru *config.neon*. těmito údaji jsou – driver (mysql, mssql,...), host, uživatelské jméno, heslo, jméno databáze a znaková sada.
- **Nakopírování veškerých souborů a php skriptů**
Na webový server je potřeba umístit všechny zdrojové soubory a zachovat adresářovou strukturu pro správnou funkčnost systému

6.3 Dolování víceúrovňových asociačních pravidel

Pro získání víceúrovňových asociačních pravidel jsem v informačním systému uplatnil přístup použití redukované minimální podpory na nižších úrovních společně s filtrováním úrovní k -množinami.

Dolování probíhá nad dvěmi úrovněmi (*Kategorie, Zboží*), kdy je pro každou úroveň použit algoritmus Apriori. V každém jeho cyklu se zvlášť získávají frekventované množiny pro jednotlivé úrovně, pro které je nastavena zvlášť hodnota minimální podpory – k tomu je použito funkcí *apriori_kat_gen*, resp. *apriori_zbozi_gen*. Součástí obou funkcí jsou oba kroky algoritmu Apriori – spojovací a vylučovací. Funkce *apriori_zbozi_gen* se liší v tom, že frekventovaná množina jednotlivých výrobků (zboží) je testována pouze v případě, že odpovídající množina na úrovni kategorií je také frekventovaná – jinými slovy, množina kategorií daných výrobků z frekventované množiny musí být také frekventovaná.

Zmíněné dvě funkce jsou společně s dalšími funkcemi (ať už těmi, které využívá samotný algoritmus Apriori – např. *kombinace_bez_opakovani*, což je funkce pro získání všech $(k-1)$ -množin z (k) -množiny bez opakování; nebo funkcemi pro získání/nastavení/změnu identifikátoru nákupu nebo vygenerování nového hesla) umístěny v souboru *funkce.php*.

Poté, co získáme konečné frekventované množiny pro každou z úrovní, dochází k vygenerování silných asociačních pravidel. Stejně jako minimální podpora použita v algoritmu Apriori, tak i spolehlivost je nastavitelná proměnná, kterou lze ve zdrojovém kódu podle potřeby upravit (jedná se o funkci *renderpridatDoKosiku* v souboru *HomepagePresenter.php*, který je součástí Modulu zákazníka – jedná se o proměnné (v závorce je uvedena výchozí hodnota proměnných) *\$min_pod(4%)*, *\$min_pod_kat(7%)* a *\$spolehlivost(40%)*.

Jak již bylo zmíněno v kapitole *Generování asociačních pravidel z frekventovaných množin*, pro každou frekventovanou množinu l , skript generuje všechny její neprázdné podmnožiny a pro každou podmnožinu s , vygeneruje pravidlo $s \Rightarrow (l-s)$ a vypočte jeho spolehlivost. Pokud jeho spolehlivost je vyšší než minimální, pak je pravidlo silné. U každého takového pravidla dochází nakonec ke kontrole, zda zboží, resp. kategorie obsažené v podmnožině s , jsou obsaženy také v nákupním košíku – pokud ano, tak je dané zboží, resp. kategorie zařazena do výpisu zboží/kategorií často se vyskytující v objednávkách s právě objednávaným produktem.

Při vkládání zboží do košíku, se poté zákazníkovi objeví „doporučené“ zboží, související s právě objednávaným produktem.

7 Testování algoritmu

Po dohodě s vedoucím práce jsem pro testování vytvořil v systému 50 objednávek různých výrobků, přičemž jsem vytvořil několik opakujících se objednávek, obsahujících buď stejné zboží, nebo zboží ze stejných kategorií. Poté jsem při experimentování upravoval hodnoty minimální podpory a spolehlivosti, které přímo ovlivňují množství objevených asociačních pravidel. Pro úroveň kategorií jsem zvolil minimální podporu vyšší, jelikož k objednání PC a monitoru dojde častěji, než k objednání určitého modelu PC společně s určitým modelem monitoru - proto je minimální podpora pro kategorie vyšší - aby odstranila "nezajímavé", ne tak časté asociace. A naopak minimální podpora pro zboží samotné je nižší. Jako výchozí hodnoty jsem pro minimální podporu kategorií zvolil 7%, zboží 4% a spolehlivost 40%. Tyto hodnoty je možné měnit a s narůstajícím počtem objednávek to bude i nezbytný úkon pro to, aby dolovací algoritmus objevil zajímavé a vypovídající asociace.

U takového vzorku dat, jakým je 50 objednávek, jsem byl schopen ověřit funkčnost dolovacího algoritmu tak, že jsem porovnal výpis frekventovaných množin z jeho jednotlivých cyklů a výsledná asociační pravidla s frekventovanými množinami a asociačními pravidly, ke kterým jsem došel manuální cestou.

7.1 Test algoritmu pro kategorie zboží

1. Zjištění kandidátů z DB a porovnání s minimální podporou (algoritmus prochází všechna objednaná zboží, zjišťuje jejich kategorii a počítá četnost objednání zboží z dané kategorie – ta se poté porovnává s minimální podporou a odstraní se kategorie, které podmínku minimální podpory nesplňují)

L_1	
<u>Množina</u>	<u>Počet výskytů</u>
{1}	15
{2}	13
{3}	21
{4}	11
{5}	8
{6}	7
{7}	14

Tabulka 7.1: Frekventované množiny kategorií první iterace algoritmu Apriori

2. Generování kandidátů z L_1 . Algoritmus kontroluje, zda každý z kandidátů splňuje podmínku minimální podpory. Poté prochází veškeré objednávky a zjišťuje počet výskytů všech kandidátů z množiny v dané objednávce – pokud množina splňuje podmínku minimální podpory je zařazena do L_2 .

L_2	
<u>Množina</u>	<u>Počet výskytů</u>
{1,2}	7
{1,7}	8
{2,7}	4
{3,6}	4
{3,5}	4
{3,4}	11
{5,6}	7
{4,6}	4
{4,5}	4

Tabulka 7.2: Frekventované množiny kategorií druhé iterace algoritmu Apriori

3. Opakování 2. kroku dokud L_i není prázdnou množinou nebo neobsahuje pouze množiny, které neobsahují ID kategorie právě objednávaného zboží (v tomto testu se jedná o tiskárnu – ID kategorie = 2)

L_3	
<u>Množina</u>	<u>Počet výskytů</u>
{3,5,6}	4
{3,4,6}	4
{3,4,5}	4
{4,5,6}	4

Tabulka 7.3: Frekventované množiny kategorií třetí iterace algoritmu Apriori

4. Jelikož množiny z L_3 neobsahují ID kategorie 2, výslednými frekventovanými množinami jsou množiny z L_2 , které obsahují ID kategorie 2 $\Rightarrow$ {1,2} a {2,7}
5. Vygenerování asociačních pravidel z frekventovaných množin.
- Neprázdnými podmnožinami první množiny jsou {1}, {2}. Budou vygenerována následující pravidla:

$$1 \Rightarrow 2; c = 7/15 = 46,67\%$$

$$2 \Rightarrow 1; c = 7/13 = 53,85\%$$
 - Neprázdnými podmnožinami druhé množiny jsou {2}, {7}. Budou vygenerována následující pravidla:

$$2 \Rightarrow 7; c = 4/13 = 30,77\%$$

$$7 \Rightarrow 2; c = 4/14 = 28,57\%$$

6. K tomu, aby byla nějaká kategorie označena jako „příbuzná“ – ta, která se s kategorií č. 2 objevuje často v objednávkách, musíme najít pravidlo $A \Rightarrow B$, jehož spolehlivost $\geq 40\%$ a množina A je již obsažena v nákupním košíku.

Takovým pravidlem je pouze pravidlo $2 \Rightarrow 1$; $c = 7/13 = 53,85\%$. Proto jsou doporučenou kategorií při objednání tiskárny *Notebooky* (ID kategorie 1) – viz obr. 7.1.

7.2 Test algoritmu pro zboží

1. Zjištění kandidátů z DB a porovnání s minimální podporou (SQL dotaz vybírá z databáze pouze ty produkty, jejichž počet objednávek $\geq$ minimální podpora) – při počtu 50 objednávek a minimální podpoře 4% musí být zboží objednáno minimálně 2x, aby splnilo tuto podmínku

L_1					
<u>Množina</u>	<u>Počet výskytů</u>	<u>Množina</u>	<u>Počet výskytů</u>	<u>Množina</u>	<u>Počet výskytů</u>
{2}	3	{14}	7	{31}	6
{3}	2	{23}	2	{35}	5
{4}	5	{24}	2	{36}	3
{8}	2	{25}	2	{39}	3
{11}	3	{28}	3	{40}	6
{12}	4	{29}	4	{41}	8
{13}	3	{30}	6		

Tabulka 7.4: Frekventované množiny zboží první iterace algoritmu Apriori

2. Generování kandidátů z L_1 . Algoritmus kontroluje, zda každý z kandidátů splňuje podmínku minimální podpory. Poté prochází veškeré objednávky a zjišťuje počet výskytů všech kandidátů z množiny – pokud množina splňuje podmínku minimální podpory je zařazena do L_2 .

L_2	
<u>Množina</u>	<u>Počet výskytů</u>
{4,12}	2
{4,41}	2
{12,41}	2
{30,36}	3
{31,35}	3

Tabulka 7.5: Frekventované množiny zboží druhé iterace algoritmu Apriori

3. Opakování 2. kroku dokud L_i není prázdnou množinou nebo neobsahuje pouze množiny, které neobsahují ID právě objednávaného zboží (v tomto testu se jedná o tiskárnu EPSON Stylus SX130 – ID zboží = 12)

L_3	
<u>Množina</u>	<u>Počet výskytů</u>
$\emptyset$	

Tabulka 7.6: Frekventované množiny zboží třetí iterace algoritmu Apriori

4. Jelikož L_3 je prázdná množina, výslednými frekventovanými množinami jsou množiny z L_2 , které obsahují ID zboží 12 $\Rightarrow \{4,12\}$ a $\{12,41\}$
5. Vygenerování asociačních pravidel z frekventovaných množin.
- a. Neprázdnými podmnožinami první množiny jsou $\{4\}$, $\{12\}$. Budou vygenerována následující pravidla:

$$4 \Rightarrow 12; c = 2/5 = 40\%$$

$$12 \Rightarrow 4; c = 2/4 = 50\%$$

- b. Neprázdnými podmnožinami druhé množiny jsou $\{12\}$, $\{41\}$. Budou vygenerována následující pravidla:

$$12 \Rightarrow 41; c = 2/4 = 50\%$$

$$41 \Rightarrow 12; c = 2/8 = 25\%$$

6. K tomu, aby byl nějaký produkt označen jako „příbuzný“ – ten, který se společně se zbožím č. 12 objevuje často v objednávkách, musíme najít pravidlo $A \Rightarrow B$, jehož spolehlivost je $\geq 40\%$ a množina A je již obsažena v nákupním košíku.

Tuto podmínku splňují pravidla **$12 \Rightarrow 4; c = 2/4 = 50\%$** a **$12 \Rightarrow 41; c = 2/4 = 50\%$** .

Proto jsou mezi doporučeným zbožím při objednání EPSON Stylus SX130 – notebook ACER Aspire 5750ZG (ID zboží 4) a flashdisk Kingston Data Traveler SE9 (ID zboží 41) – viz obr. 7.1.

Kategorie

- Flashdisky
- Klávesnice
- LCD monitory
- Myši
- Notebooky
- Osobní počítače
- Tiskárny a skenery

Výrobci

- HP
- Asus
- Acer
- Canon
- Epson
- A-data
- Kingston
- Logitech



Zboží bylo přidáno do košíku.
EPSON Stylus SX130

<<Zboží
Košík>>

Nákupní košík

Položky: 1
Celkem: 1199 Kč



Novinka

Kingston DataTraveler SE9 16GB



300 Kč

Speciální nabídka

HP Pavilion dv6-6b85ec Dark Umber



23999 Kč

ZÁKAZNÍKŮ KUPUJÍCÍCH TENTO PRODUKT ZAKOUPILI TAKÉ

ACER Aspire 5750ZG-B958G75Mnrr



13690 Kč

Koupit
Více

Kingston DataTraveler SE9 16GB



300 Kč

Koupit
Více

PŘÍBUZNÉ KATEGORIE

Notebooky

Obr. 7.1: Zobrazení doporučeného zboží/kategorií získaných z asociačních pravidel

8 Závěr

Navržený a implementovaný systém splňuje všechny požadavky stanovené v zadání. Při jeho návrhu jsem kladl důraz na snadné a intuitivní ovládání systému takovým způsobem, aby uživateli ušetřil co nejvíce času a aby s ním mohl pracovat i méně zkušený uživatel.

Úvodními požadavky byla analýza požadavků na internetový obchod a seznámení se s problematikou dolování dat se zaměřením na víceúrovňová pravidla, dále potom na základě získaných znalostí vybrat vhodný algoritmus pro získání asociačních pravidel. Veškeré detaily, s tímto spojené, mi objasnil vedoucí práce - pan Ing. Vladimír Bartík, se kterým jsem konzultoval především implementaci dolovacího algoritmu.

Při návrhu databáze a internetového obchodu jsem vycházel ze všech doposud získaných informací a poté návrh implementoval. Součástí informačního systému (internetového obchodu) je vedení skladu, administrace, automatické zpracování objednávek, doporučení produktu, který je často objednáván(asociován) s právě objednávaným produktem, vedení plateb, daňové evidence a statistik.

Aplikaci jsem průběžně testoval během vývojové fáze, čímž se mi povedlo většinu chyb a nedostatků odhalit a opravit průběžně. S tím samozřejmě souvisí i drobné změny v návrhu systému, které vyplynuly při implementaci. Po dokončení jsem finální verzi opět testoval v provozu, přičemž dalším krokem zadání bylo otestování funkčnosti implementovaného řešení dolování asociačních pravidel na vhodně zvoleném vzorku dat – toto testování popisuje a shrnuje předešlá kapitola.

Dalším možným rozšířením systému by mohly být následující funkce:

- Věrnostní program – čím více zákazník nakupuje, tím větší získá výhody/slevy
- Zasílání novinek na e-mail registrovaného zákazníka
- Jazykové rozšíření – anglický jazyk
- Daňová evidence pro plátce daně

Literatura

- [1] **SHANNON, R.** *What is HTML?* [Online]. [cit. 2011-09-29]. Dostupné na: <<http://www.yourhtmlsource.com/starthere/whatishtml.html>>.
- [2] *Latte* [online]. [cit. 2011-09-29]. Dostupné na: <<http://www.latte.org/>>.
- [3] *Cascading Style Sheet Definition* [online]. [cit. 2011-09-29]. Dostupné na: <<http://www.techterms.com/definition/css>>.
- [4] **JANOVSKÝ, D.** *Úvod do JavaScriptu* [online]. [cit. 2011-09-29]. Dostupné na: <<http://www.jakpsatweb.cz/javascript/javascript-uvod.html>>.
- [5] *What is PHP?* [online]. [cit. 2011-09-29]. Dostupné na: <<http://www.softwareprojects.org/php-what-is-01.htm>>.
- [6] *MySQL databáze - český manuál* [online]. [cit. 2011-09-29]. Dostupné na: <<http://www.junext.net/mysql/>>.
- [7] *Wikipedie, otevřená encyklopedie - Nette Framework* [online]. [cit. 2011-10-04]. Dostupné na: <http://cs.wikipedia.org/wiki/Nette_Framework>.
- [8] *Nette Framework – Hlavní přednosti* [online]. [cit. 2011-10-04]. Dostupné na: <<http://nette.org/cs/hlavni-prednosti>>.
- [9] **GRUDL, D.** *Nette Framework - MVC & MVP* [online]. [cit. 2012-01-09]. Dostupné na: <<http://zdrojak.root.cz/clanky/nette-framework-mvc--mvp/>>.
- [10] **ZENDULKA, J; BARTÍK, V; LUKÁŠ, R; RUDOLFOVÁ, I.** *Získávání znalostí z databází, studijní opora* [online]. [cit. 2012-02-22]. Dostupné na: <<http://www.fit.vutbr.cz/~bartik/ZZN.pdf>>.
- [11] *Use Case Diagrams* [online]. [cit. 2011-10-05]. Dostupné na: <<http://www.uml-diagrams.org/use-case-diagrams.html>>.
- [12] **KŘENA, B; KOČÍ, R.** *Úvod do softwarového inženýrství, studijní opora* [online]. [cit. 2011-10-05]. Dostupné na: <https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/IUS-IT/texts/IUS_opora.pdf?cid=6313>.

Příloha A

Přihlašovací údaje zaměstnanců/zákazníků

Zaměstnanci:

Pozice	Login	Heslo
Administrátor	admin	admin
Skladník	skladnik	skladnik
Expeditor	expeditor	expeditor
Účetní	ucetni	ucetni
Reklamační	reklamacni	reklamacni

Tabulka A.1: Přihlašovací údaje zaměstnanců

Zákazníci:

Jméno	Login	Heslo
Michael Králík	samic	samic
Petra Smitalová	petra	petra
Luděk Koutný	spune	spune

Tabulka A.2: Přihlašovací údaje zákazníků