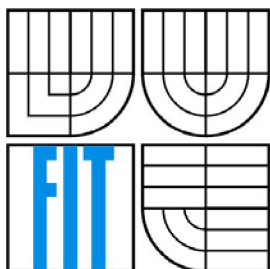


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INTERNETOVÝ OBCHOD S VYUŽITÍM
TECHNOLOGIE MACROMEDIA FLASH
INTERNET STORE USING TECHNOLOGY MACROMEDIA FLASH

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

Zbyněk Padělek

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. Jaroslav Zendulka, CSc.

BRNO 2007

Abstrakt

Tato bakalářská práce se zabývá tvorbou elektronického obchodu s využitím technologie Macromedia Flash. Cílem bylo vytvořit systém podle požadavků zákazníků na elektronické obchody.

Nejprve je představen pojem elektronický obchod a prostředí použité při vývoji. Následuje popis konkrétních požadavků na systém a jejich analýza. Práce pokračuje popisem technologií použitých při tvorbě systému. V závěrečné části se zabývá implementací systému jako webové aplikace s využitím Flash, XML a ColdFusion.

Klíčová slova

databáze, Flash, XML, ColdFusion, PHP, MySQL

Abstract

This bachelor work is concentrated on building an electronic shop with the use of Macromedia Flash technology. The aim was to build a system according to the purchasers' request of the electronic shop.

First this work explains the term of electronic shop and then the medium used in the develop. Then it is followed by the description of particular requests of the system and their analysis. The work continues describing the technologies used in building the system. The final part is concentrated on implementation of the system as a web application with the use of Flash, XML, and ColdFusion.

Keywords

database, Flash, XML, ColdFusion, PHP, MySQL

Citace

Zbyněk Padělek: **Internetový obchod s využitím technologie Macromedia Flash, bakalářská práce, Brno, FIT VUT v Brně, 2007**

Internetový obchod s využitím technologie Macromedia Flash

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením doc. Ing. Jaroslava Zendulky, CSc. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Zbyněk Padělek
10. května 2007

Poděkování

Především bych rád poděkoval vedoucímu své bakalářské práce panu doc. Ing. Jaroslavu Zendulkovi, CSc. za odborné vedení a čas věnovaný konzultaci této práce. Také bych chtěl poděkovat panu Ing. Miloši Eysseltovi za trpělivost a ochotu při obhajobách semestrální projektu.

© Zbyněk Padělek, 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 Elektronické obchodování	4
1.1 Nakupování na internetu	4
1.2 Moduly elektronického obchodu (funkčnost modulů)	5
2 Technologie Macromedia Flash	7
2.1 Charakteristika produktu Flash	7
2.2 Objektové rozhraní pro výtvarníky	7
2.2.1 Komponenty movie a scéna	7
2.2.2 Vizualní objekt MovieClip	8
2.3 Objektové rozhraní pro programátory	9
2.3.1 Programovací jazyk ActionScript	9
2.4 ColdFusion	9
2.5 Architektura Flash - databáze	10
2.6 Jaké jazyky v práci použiji	11
3 Požadavky na elektronický obchod	12
3.1 Obecné požadavky	12
3.1.1 Požadavky jednotlivých elementů obchodu	12
3.2 Analýza	13
3.2.1 UML	14
3.2.2 Modely systému	14
4 Návrh implementace	18
4.1 Volba implementačního prostředí	18
4.2 Blokový popis	18
4.3 Návrh implementačního rozhraní	19
4.4 Návrh struktury databáze	19
4.5 Popis souborů systému	20
4.5.1 Soubory obsluhované animací	20
4.6 Autentizace uživatele	21
4.7 Vývoj a testování	23
5 Závěr	24
5.1 Dosažené výsledky	24
5.2 Přínos práce	25
Literatura	26

Seznam příloh	27
Příloha 1:.....	28
Příloha 2.....	31

Úvod

Obchodování a nakupování se stále více orientuje na internet, který je pro zákazníky nejpohodlnější a pro prodávajícího nejlevnější. Správně vytvořený obchod je základem k úspěchu na poli elektronického obchodování. Co si vlastně máme představit pod pojem elektronické obchodování? Pro internetové obchody existuje několik různých pojmenování. Elektronický obchod, virtuální obchod, internetový obchod, on-line obchod nebo e-shop. V každém případě jde o obchod, který existuje v podobě internetových stránek.

Tato bakalářská práce se zabývá tvorbou elektronického obchodu s využitím technologie Macromedia Flash. Cílem bylo vytvořit systém podle požadavků zákazníků na elektronické obchody.

Tvorbu systému lze rozložit do několika dílčích úkolů. Nejprve je třeba seznámit se s požadavky kladenými na systém. Tyto získané požadavky pak podrobit analýze a vytvořit modely systému. Následuje volba implementačního prostředí. Teprve poté nastává etapa vlastní realizace systému. Nakonec se zhodnotí dosažené výsledky.

Práce je logicky členěna do několika kapitol. V první kapitole je čtenář obecně seznámen s pojmem a moduly elektronického obchodování. Ve druhé kapitole následuje popis zvolené technologie Macromedia Flash. Ve třetí kapitole je popsán rozbor požadavků na systém. Tyto požadavky jsou analyzovány s využitím modelování v jazyku UML, konkrétně byl vytvořen diagram případů použití (use-case diagram) a ER diagram. Čtvrtá kapitola popisuje vlastní implementaci systému a testování. Pátá kapitola obsahuje ukázkou použití systému. Závěr a možnosti dalšího rozvoje systému obsahuje poslední šestá kapitola.

Znalosti o databázích, jejich návrhu a o modelovacích technikách k tomu vhodných byly čerpány z [1]. Pro prohloubení znalostí o Flash, ActionScript a XML posloužilo [3, 4, 5]. Použití jazyka CFML bylo nastudováno z [2]. Nepostradatelné byly [5, 6]. Obecná fakta o použitých jazycích a jejich vývoji byla čerpána z [7].

1 Elektronické obchodování

Co to vlastně elektronický obchod vůbec je a co si pod tímto pojmem můžeme představit. Pro internetové obchody existuje několik různých pojmenování. Elektronický obchod, virtuální obchod, internetový obchod, on-line obchod nebo e-shop. Pod tímto pojmem se může skrývat vše od průzkumu trhu, vyhledávání zboží a nabídky zboží, přes samotný prodej či nákup, až po zákaznické služby. Jedná se tedy o „schopnost důvěryhodně provádět obchodní transakce pomocí komunikační sítě způsobem, který je účinnější a ekonomičtější v porovnání s jinými způsoby obchodu“.

Tato kapitola obsahuje klasifikaci elektronických obchodů, na které můžeme na internetu narazit.

1.1 Nakupování na internetu

Internetové obchody u nás fungují již několik let. Počet uživatelů Internetu (a tedy i potenciálních zákazníků) dosáhl potřebného kritického množství, padají psychické bariéry z nákupů ve „virtuálních obchodech“, vyřešeny jsou i některé otázky zabezpečení plateb a obchodního styku vůbec.

V České republice již existují firmy, které prostřednictvím elektronického obchodu realizují významnou, ne-li hlavní část svého obratu. U určitých druhů zboží již začíná klasický obchodní model zaostávat. Elektronický obchod nabízí zákazníkům mnohem širší nabídku než „kamenný obchod“ a navíc s přidanou hodnotou. Například dodání až do domu (kanceláře), výhodnější cena, větší dostupnost informací o zboží a podobně.

S rostoucím rozvojem elektronického bankovníctví si stále více zákazníků oblíbilo tento způsob nakupování. Výhody ze strany zákazníka:

- **Ušetřený čas** Do virtuálního obchodu může zákazník přijít třeba v sobotu o půlnoci. Elektronický obchod nikdy nezavírá!
- **Finanční úspora** Protože obecně platí, že elektronické obchody své zboží neprodávají za ceny vyšší než obchody kamenné (úspora při nákupu na internetu se v průměru pohybuje mezi deseti až patnácti procenty). Provozovatel virtuálního obchodu totiž nemusí platit za pronájem prostor a stejně tak ušetří i za mzdy nadbytečných zaměstnanců. Na nižších cenách zboží se projevuje i skutečnost, že se k existenci elektronického obchodu neváže nutnost vlastnit skladovací prostory v případě, kdy obchodník může zboží odebírat z velkoskladů, nebo dokonce přímo od výrobců.

- **Možnost dohledání všech dostupných informací o zboží** Na rozdíl od kamenných obchodů, kdy je zákazník odkázán na informace od zaměstnanců (mnohdy nedostatečně proškolených či neochotných), je v elektronickém obchodě snadné o dané komoditě shromáždit požadované informace na jednom místě. A to včetně odkazu na výrobce nebo na recenzi daného zboží.

1.2 Moduly elektronického obchodu (funkčnost modulů)

Systém pro elektronický obchod lze rozdělit na tři části. Jsou to **informační báze**, která obsahuje nabízené produkty, další částí je **vyhledávací systém**, což je prostředek pro vyhledávání v informační bázi. V prostředí Internetu tuto funkci velice dobře splňují klasické webovské prohlížeče. Poslední částí je elektronický platební systém (EPS), který slouží k provádění obchodních transakcí.

Jednotlivé části jsou složené z několika dalších, které by každý elektronický obchod měl obsahovat. Pro jednoduchost je pojmenuji moduly:

- **Katalog produktů** - katalogově řazené produkty do kategorií. Univerzální řešení, možnost tvorby atypických produktových řad.
- **Vyhledávání** - vlastní volby vyhledávacích filtrů.
- **Novinky, press a mimořádné nabídky** - prodejní akce. Sdělení. Tiskové zprávy. Jednoduchý a logický marketingový nástroj.
- **Nákupní košík** - objednávka či poptávka, provázání s platebními mechanismy.
- **Základní informace o e-shopu** - publikování informací typu Společnost, obchodní podmínky, kontakt apod.
- **Registrace a Uživatelé** - podrobná databáze klientů, práce pod heslem (např. zasílání mailů).
- **Platby** - platební systém podporující SSL přenos dat (e-banka, OK Karta, YES Karta).
- **Redakční systém** - možnost vlastního okamžitého provádění změn jakékoliv části aplikace.
- **Mail Robot, Mail list** - pravidelné, automatické a hromadné zasílání informací registrovaným uživatelům.
- **Statistiky** - sledování statistik přístupů na stránky. Velmi přehledné, grafické výstupy.

Většina elektronických obchodů nabízených na českém trhu tyto moduly však obsahuje. Mnohé nabízejí jako svojí konkurenční výhodu i něco navíc. Pro příklad jmenujme relativně jednoduchou vizuální změnu vzhledu obchodu, individuální nastavení obchodu registrovaným zákazníkům (kupříkladu zákaznice pravidelně nakupující drogistické zboží si stránku „přebuduje“ dle svých představ a nejspíše nebude ochotna se „proklikat“ nabídkou venkovních barev na plot) nebo přímou podporu některého z vybraných účetních systémů.

2 Technologie Macromedia Flash

Macromedia Flash je vývojové prostředí, umožňující vytvářet animované interaktivní komponenty, jejichž chování se dá programovat. Vytvořené „flashové“ komponenty se dají přehrávat (pomocí vestavěného přehrávače, volně stažitelného z Internetu) přímo ve webových stránkách, nebo se dají využít jako ActiveX komponenty v jiných programech. V této kapitole představím zvolený produkt Macromedia Flash jako objektové rozhraní pro výtvarníky i pro programátory.

2.1 Charakteristika produktu Flash

Flash prošel poměrně dlouhým vývojem. Původně to byl především program pro pouhé vytváření animovaných obrázků. Postupně se rozšiřovala možnost vytvářené animace řídit pomocí skriptů, jejichž syntaxe se postupně vyvíjela a obohacovala. Poslední verze Macromedia Flash Professional 8 má již zabudován objektový řídicí jazyk (ActionScript, nyní ve verzi 3.0), syntaxí velmi podobný jazyku Java, který umožňuje poměrně pohodlně a sofistikovaně řídit chování vizuálních interaktivních elementů.

Velký úspěch vývojového prostředí Macromedia Flash je mimo jiné založen na tom, že tvůrcům se poměrně úspěšně podařilo definovat rozhraní pro výtvarníky (vytvářející základní animační prvky) a programátory, kteří těmto komponentám mohou pomocí výše zmíněného objektového jazyka vdechnout interaktivnost.

2.2 Objektové rozhraní pro výtvarníky

2.2.1 Komponenty movie a scéna

Základní komponentou aplikací ve Flashi je film (movie). Film je možné dělit na jednotlivé scény, které je možno postupně či programově (na přeskáčku) přehrávat. Scény jsou tvořené sekvencí jednotlivých snímků (frames). Filmy (movies) je možno libovolně řetězit - z každého filmu (movie) je možné programově zavolat zavedení dalšího filmu a pak spustit jeho přehrávání. To má výhodu zejména v internetových aplikacích, kdy po zavedení a spuštění první části animace se na pozadí z příslušného serveru stahuje její další část.

Ve vývojovém prostředí je možné jednotlivé filmy vytvořit (jak po grafické, tak i po programátorské stránce), otestovat a přeložit do mezijazyka (ve formě .swf souboru), který je možno interpretovat pomocí volně stažitelného interpretu (tzv. Flash Playeru) a přehrávat buď jako

samostatně spustitelnou animaci nebo ji prohlížet přímo v internetovém prohlížeči. Krom toho je možné vytvořený .swf soubor interpretovat pomocí speciální ActiveX komponenty, kterou je možno zabudovat do jiného programu. Důležité je, že tato komponenta si s aplikací může vyměňovat zprávy a tak můžeme jinou aplikací pohodlně řídit chování interaktivní animace. Aplikace zároveň může od interaktivní animace přijímat zprávy, referující o zásazích uživatele.

Ve Flashi je každá scéna tvořena několika vrstvami. Každé políčko filmu, resp. scény (frame) má tedy několik vrstev do nichž se ukládají jednotlivé obrazové prvky. Tyto vizuální prvky se mohou v každé vrstvě samostatně kreslit – Flash obsahuje poměrně výkonný nástroj pro vytváření vektorových obrázků. Jinou možností je vybrat si z knihovny příslušný obrázek a vytvořit jeho instanci.

2.2.2 Vizuální objekt MovieClip

Instancí však nemusí být jenom statický obrázek. Instancí může být i tzv. filmový klip (MovieClip), který je vlastně instancovanou třídou již dříve vytvořeného filmu. Speciálním druhem filmových klipů jsou tlačítka (Buttons), u nichž je možno definovat jak jejich vizuální vzhled (při přejetí kurzoru nad tlačítkem a při jeho stisku), tak i chování obslužením příslušné vyvolané události.

Vlastní MovieClip může mít i poměrně složitou hierarchickou strukturu – film, který ho vytváří může obsahovat instance dalších filmových klipů. Tak např. MovieClip auta může v sobě obsahovat MovieClipy jeho točících se kol. MovieClipy je možné (jako třídy) ukládat do knihoven a jako viditelné objekty umísťovat jejich instance na scéně (a to nejen "ručně" při tvorbě aplikace, ale i programově, pomocí volání příslušných konstruktorů).

Každá instance MovieClipu má své vlastnosti (koordináty umístění na scéně, velikost, přebarvení, průhlednost apod.), které je možné dynamicky měnit z programu. Krom toho má třída MovieClip celou řadu metod, které můžeme využívat (např. metodu pro detekci kolize s jinou instancí MovieClipu na scéně apod.). Při tvorbě MovieClipu můžeme také samozřejmě naprogramovat i vlastní specifické metody, které pak můžeme volat u všech jeho instancí. Tak je možné naprogramovat poměrně složité chování vizuálních komponent.

Konečně, poslední dvě verze Macromedia Flash umožňují poměrně jednoduše vytvářet speciální MovieClipy jako skutečné komponenty, a jejich vlastnosti pak nastavovat ve speciálním editoru komponent a za běhu volat jejich metody.

Komponenty nemusí být pouze vizuální – ActionScript ve verzi 3.0 je skutečný objektový jazyk (v němž je nyní možno definovat třídy, dědičnost, polymorfismus) což dává velké možnosti vytváření řady i nevizuálních komponent, které výtvarník může využít při tvorbě interaktivních animací (např. komponenta pro tvorbu stopy za kurzorem aj.).

2.3 Objektové rozhraní pro programátory

2.3.1 Programovací jazyk ActionScript

ActionScript je objektově orientovaný programovací jazyk pro aplikace vyvíjené pomocí Macromedia Flash, případně dalších vývojářských nástrojů, využívajících stejného datového formátu.

Poslední verze Flashe přinesla rozšíření programovacího jazyka ActionScript o možnosti pohodlného vytváření tříd (zavádí se elementy jazyka: class, extends, implements, interface, dynamic, static, public, private, get, set, import aj.) a komponent, takže programování se stává konečně jednodušší a přehlednější. Tím nechci říci, že v minulé verzi jazyka nešlo vytvářet vlastní třídy a objekty, ale syntaxe jejich tvorby pro programátory Java, C++, C# byla poněkud nestandardní.

Vše vyplývalo z toho, že původní skriptovací jazyk měl umožnit měnit vizuální složky vytvářených animací spíše pomocí vestavěných metod vizuálních elementů, než složitějším vytvářením hierarchie objektů a tříd. Jak se postupně Flash rozvíjel, byly do jazyka postupně zabudovávány další možnosti, které byly nakonec završeny novou verzí jazyka ActionScript 3.0.

Nová verze také přinesla zpracování výjimek, rozlišování identifikátorů i podle velikosti písmen a další vylepšení.

Přesto se však Flash od klasických Java jazyků trochu liší - Flash např. pracuje s jediným numerickým typem (Numeric) a dále má základní typy Boolean, Date a String, typ Character je nahrazen jednoznakovým Stringem, zato však pracuje výhradně se 64 bitovými znaky, takže odpadají problémy se znaky různých abeced. Kořenem všech typů je typ Object. ActionScript zná objekt Error, takže zpracování výjimek je nyní pohodlné. Typ Array má řadu vlastností nejen polí ale i seznamů.

ActionScript má řadu vestavěných tříd (Math, System atd.). Řada vestavěných tříd souvisí s tvorbou vizuálních objektů – třídy MovieClip, Stage (scéna), Sound, Video, Microphone, Mouse, Key, PrintJob, Color, Accessibility aj. Další vestavěné třídy souvisejí s internetovým propojením a výměnou dat (třídy XML, XML_Node, XML_Source).

Velkou výhodou programů ve Flashi je platformová nezávislost vytvářených aplikací – což je podporováno volně stažitelným interpretem přeložených souborů Flash. Flash podporuje dobrou spolupráci s Javou a prostřednictvím ActiveX komponent i s prostředím operačních systémů od Microsoftu.

2.4 ColdFusion

ColdFusion je serverové řešení pro rychlý vývoj a provoz dynamických webových aplikací. Díky robustnímu, ale lehce pochopitelnému skriptovacímu jazyku, jednoduchému připojení na lokální i podnikové databáze, vestavěnému vyhledávacímu modulu a doplňku pro tvorbu grafů mohou

vývojáři velmi rychle vytvářet komplexní dynamicky generované webové stránky, publikační systémy a e-commerce aplikace.

Zdálo by se, že jde o klasický preprocesorový nástroj, jakých dodnes vznikla celá řada. Se svou první verzí vydanou v polovině roku 1995 jde ale nepochybně o jeden z prvních nástrojů svého druhu - bohužel ve své době málo rozšířený. Stránky pro ColdFusion se píšou v CFML, což je nadmnožina jazyka HTML rozšířená o speciální tagy formátu <cf...>. Narozdíl od ostatních technologií zde programování probíhá neprocedurálně pomocí specializovaných tagů (např. cfquery pro spojení s databází nebo cfform pro tvorbu formuláře apod.) spíše než psaním komplexních skriptů. Dnes patří do rozsáhlé rodiny vývojových nástrojů Macromedia MX.

ColdFusion MX pracuje na systémech Windows, Linux a Solaris a podporuje standardy .NET a Java. Přímé propojení s Macromedia Flash umožňuje výměnu dat mezi Flash aplikací a databází v reálném čase.

2.5 Architektura Flash - databáze

Flash můžeme používat s mnoha rozličnými prostředími na straně serveru, ať už se jedná od PHP, ColdFusion, ASP.NET a Java, ale můžeme jej používat i v jiných prostředích, včetně Perl, CGI nebo s kódem ISAPI/NSAPI, prostě se vším, čemu se říká middleware.

Uvnitř každého serverového prostředí můžeme své systémy integrovat s programem Flash pomocí jakékoli kombinace následujících technik:

- Předávání hodnot tam a zpět ve stylu CGI jako dvojice název/hodnota. Konceptně je to obdobné tomu, jak se dnes pracuje s formuláři a s parametry URL, ale bez toho aby se stránka připojovala na server.
- Předávání hodnot tam a zpět mezi Flashem a naším serverem ve formě XML. To umožňuje vyměňovat se serverem více fasetová, libovolně složitá data.
- Necháme na svém serveru Flash připojený k Flash Remoting Gateway. To je obvykle nejjednodušší cesta, která také mívá nejméně řádků kódu. Pracujeme-li s ColdFusion nebo s JRun, už máme přístup k Flash Remoting. Je v nich zahrnut. V ostatních případech si potřebujeme obstarat licenci na Flash Remoting a nainstalovat bránu na našem serveru.

2.6 Jaké jazyky v práci použiji

Ve své práci budu používat nejen programovací či skriptovací jazyky, ale také značkovací a mluvím i o vizuálních prvcích, které budu přidávat do Flashe. Které jazyky tedy v práci použiji:

- Jazyk na serveru, který jsem si zvolil. CFML, při použití ColdFusion.PHP při použití Apache.
- ActionScript. Moderní, flexibilní skriptovací jazyk. Pomocí ActionScriptu se ve Flashi dělají takové věci, jako interakce se serverem, změna informací zobrazujících se uživateli, nebo pohybování věcmi po obrazovce. Je založen na stejném standartu jako Javascript.
- Prvky vizuálního jazyka. Jedním ze zajímavých aspektů práce ve Flashi je, že lze vizuální prvky (obrázky, tlačítka, snímky, animace) chápat jako programovací objekty, které mají své vlastní funkce, skriptovatelné vlastnosti atd. Tedy vizuální prvky lze opravdu chápat jako jisté stavební bloky programovacího jazyka (nebo aspoň jako části vizuálního slovníku).
- XML. To samozřejmě není programovací jazyk, ale jazyk to je. Pomocí XML se dají velice snadno vyměňovat strukturovaná data a instrukce mezi naším serverem a Flashem.
- SQL. Za předpokladu, že jsou některá z informací, které chcete prezentovat v rozhraní posílené Flashem, uložené v nějaké databázi, budete i nadále své tabulky aktualizovat a získávat z nich informace pomocí SQL.

3 Požadavky na elektronický obchod

Prvním krokem při tvorbě nového systému je sběr požadavků na tento systém. Požadavky je třeba důkladně a podrobně analyzovat a na základě této analýzy systém navrhnout. Fáze analýzy je velmi důležitá. Pokud zde nastane chyba a ta se odhalí až ve fázi implementace, je mnohdy velmi náročné chybu napravit.

3.1 Obecné požadavky

Systém je vytvářen podle reálných požadavků, které jako častý aktivní uživatel elektronického obchodování považují za stěžejní. Požadavky byly získány při nakupování v elektronických obchodech z České republiky i zahraničí.

3.1.1 Požadavky jednotlivých elementů obchodu

3.1.1.1 Rozhraní obchodu - katalog

Je velmi důležitý element elektronického obchodování. Je to místo kde se prohlížením výrobků rozhodnou zákazníci nakupovat. Zákazníci chtějí výrobky vidět, přečíst si o nich a my jim musíme poskytnout vše co potřebují vědět, aby se rozhodli nakupit. Důležitou prvkem je, že chtějí vědět cenu. Mnoho lidí se rozhoduje podle ceny, množství výrobků, nebo kvality. Velice důležité je rozvržení obchodu – chceme přece zákazníka povzbudit nakoupit u nás. V katalogu jde o tyto požadavky:

- Možnost procházení kategorií výrobků.
- Možnost detailních informací u zvoleného výrobku.
- Hodnocení zboží.
- Kontaktní formulář.

3.1.1.2 Nákupní košík, karta

Jakmile si zákazník vybere výrobek, nebo se o něj zajímá, chce si jej vzít sebou, dokud pokračuje v prohlížení sortimentu. Nechceme je nutit nakupovat ihned po vybrání. Dokud nakupují chtějí se podívat na výrobky v košíku, kvůli srovnání. Tohle vše ovlivňuje zákaznickovu mysl, takže je důležité udělat to pro ně jednoduché. Nakonec se rozhodnout zaplatit, nebo vrátit výrobek zpět. To vše musíme umožnit, nechceme v žádném případě děsit naše zákazníky. Musíme jim pomoci a vést je.

- Uživatelské účty.
- Opětovný nákup bez nutnosti vyplňovat všechny údaje.
- Sledování stavu nákupu, zasilání informací o stavu na e-mailovou adresu.
- Historie nákupů.

3.1.1.3 Platba

Platba je velice důležitým prvkem v online nakupování. V kamenném obchodě si zákazník vybere a u pokladny zaplatí v hotovosti, ale v online nakupování platí cenu včetně daně a ceny za dopravu, opět jim musíme snadnou cestou ukázat, co platí, kolik stojí doprava. Jestli si můžou zboží vyzvednout, nebo nechat doručit domů.

- Detailní výpis platby
- Pokladna s možností registrace.

3.1.1.4 Distribuce

Podle druhu produktu a území, které hodláme pokrýt, můžeme buď distribuci převzít sami, použít některé z existujících kurýrních služeb, nebo staré dobré pošty. Samozřejmě nám tato starost odpadá, když se pustíme do prodeje produktů a služeb, které lze distribuovat přímo po síti. Zákazník si tedy po výběru zboží zvolí, zdali si zboží chce vyzvednout osobně, nebo si jej nechá zaslat pomocí pošty či zásilkové služby.

- Objednávky a průběh transakcí.
- Zajištění dopravy k zákazníkovi.

3.2 Analýza

V systému bude uloženo poměrně velké množství dat. V těchto datech je třeba se vyhledávat a zobrazovat výsledky. Systém musí umožňovat současnou práci více uživatelů, je tedy třeba řešit konflikty při práci s daty, které mohou následkem toho vzniknout. Navíc je potřeba data zálohovat. Ze všech těchto požadavků vyplývá, že pro ukládání dat bude, nejvhodnější použít některý z existujících databázových systémů. Databáze bude umístěna na serveru, ke kterému se budou přes počítačovou síť připojovat klienti. Tito klienti pak budou s daty pracovat.

Dalším krokem je analýza způsobu používání systému, uchovávaných dat a jejich struktury. Tento krok podpořilo modelování systému, kdy jsem využil modelovací jazyk UML. Z vytvořených modelů jsem následně vycházel při návrhu systému a jeho následné implementaci.

3.2.1 UML

UML (Unified Modeling Language) je grafický jazyk používaný pro vizualizaci, specifikaci, navrhování a dokumentaci programových systémů. Podporuje objektově orientovaný přístup. Díky svým možnostem se používá nejen v softwarovém inženýrství. Uplatňuje se také při modelování hardwaru, business procesů, organizačních struktur a v dalších oblastech.

Historie UML se začíná v polovině 90. let. V té době bylo k dispozici mnoho různých metod objektové analýzy a návrhu. Situace v tomto odvětví byla dosti nepřehledná. Nastala potřeba vytvoření jednotného jazyka. V roce 1994 začali na takovém jazyce pracovat Grady Booch a Jim Rumbaugh, zaměstnanci Rational Software. Vycházeli ze svých metodik Booch a OMT (Object Modeling Technique) s cílem použít jejich společné vlastnosti. V roce 1995 se do vývoje UML zapojil Ivar Jacobson se svou metodou OOSE (Object-Oriented Software Engineering). UML se ve verzi 1.1 stává standardem v roce 1997. V současné době se pracuje na verzi 2.1

3.2.2 Modely systému

Modely systému jsou trojího druhu:

Funkcionální model popisuje funkcionalitu systému z pohledu jeho uživatele. Sem se řadí diagram případů použití.

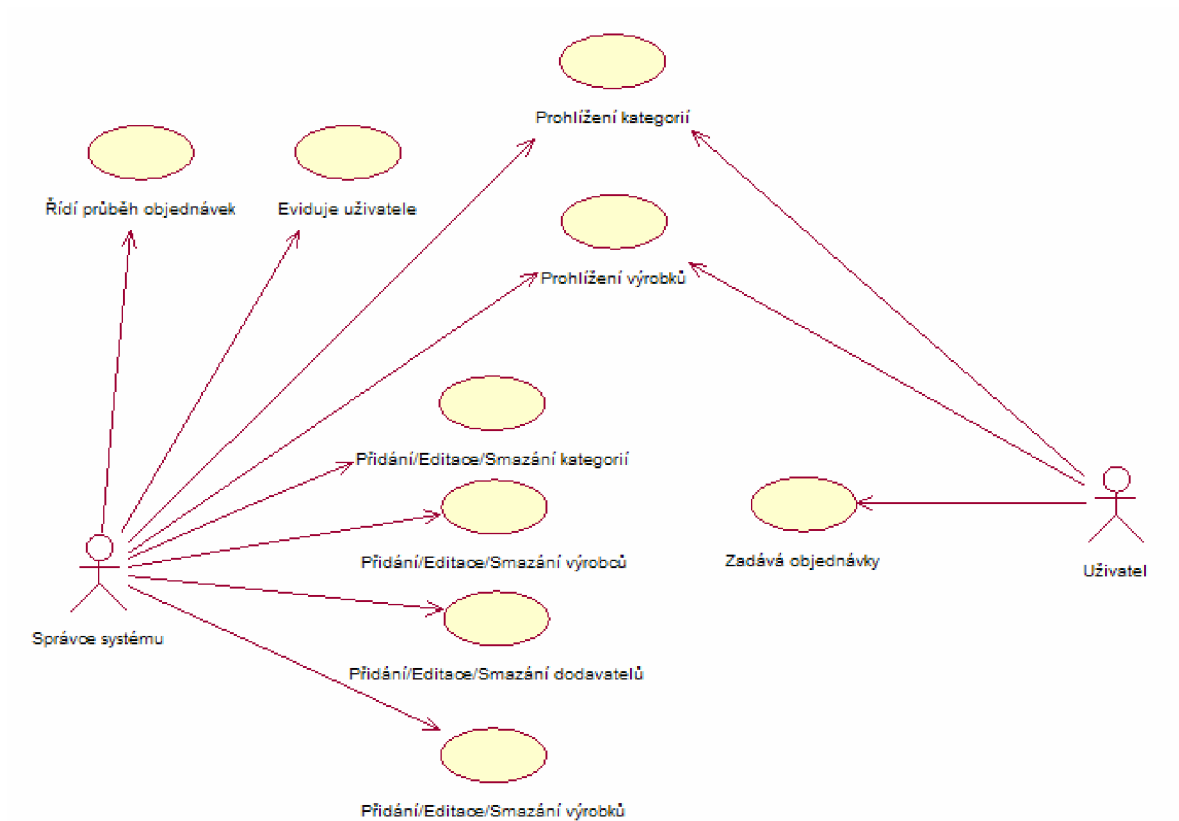
Objektový model nebo model dat popisuje strukturu systému, objekty uvnitř systému a jejich vlastnosti a asociace. Sem patří diagram tříd. Notace, kterou používá, je vhodná i pro tvorbu ER diagramů, kde se využijí prvky pro popis datové části systému.

Dynamický model nebo model chování popisuje vnitřní chování systému.

Diagram je určitá grafická reprezentace daného modelu systému.

Diagram případů použití

Diagram případů použití (use-case diagram) znázorňuje činnosti, které vykonávají dané typy uživatelů. Zachycuje činnosti systému z pohledu uživatele, interakci uživatele se systémem. Obsahuje všechny role uživatelů, kteří budou se systémem pracovat, a všechny případy použití, které jim systém nabízí.



Obr. 3.1: Diagram případů použití

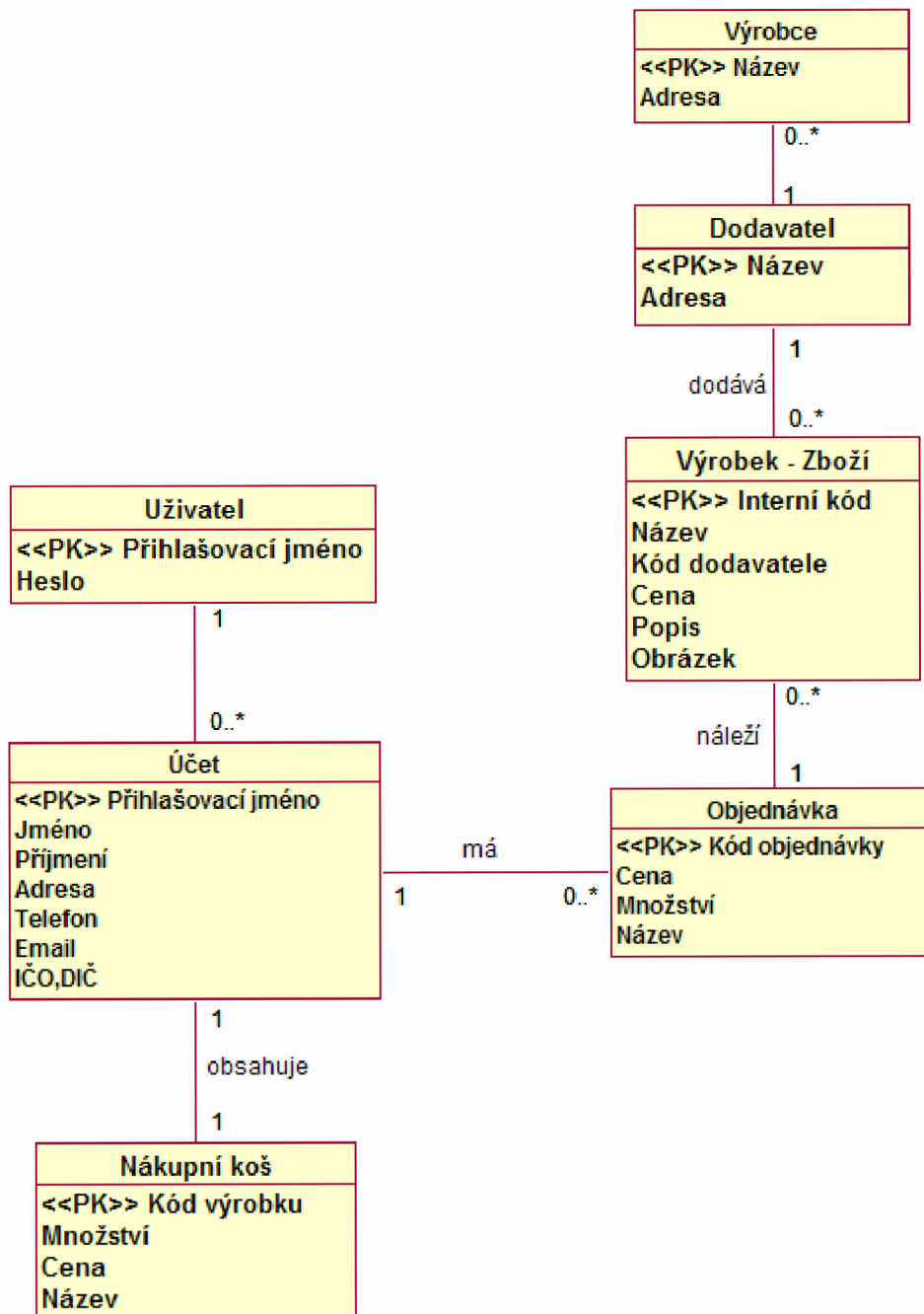
Diagram případů použití navrhovaného systému vidíme na obrázku 3.1. Vy stupují v něm dvě role uživatelů. Libovolný *uživatel* má dovoleno prohlížet si kategorie výrobků a do nich zařazené jednotlivé výrobky. Dále může zadávat objednávky.

Pojem „*správce systému*“ v sobě zahrnuje vytváření, mazání a úpravu kategorií, výrobků, výrobců a dodavatelů. Jednotlivé kategorie a výrobky budou zobrazovány v předem vytvořeném prostředí, odpovídající zvolenému druhu prodávaného sortimentu. U jednotlivých výrobců budou uvedeny odkazy na jejich detailní zobrazení. Pro zadávání objednávek bude sloužit tlačítko „Buy“. Které přidá vybrané zboží do nákupního košíku. Pro registraci nových uživatelů, nebo změnu stávajících údajů bude sloužit formulář, který se zobrazí po stisknutí příslušného tlačítka nebo odkazu. Pro efektivní práci se systémem je třeba v záznamech vyhledávat. Výsledkem vyhledávání bude přehledová tabulka zobrazující pouze záznamy vyhovující parametrům hledání. *Správce systému* dále řídí průběh objednávek. Navíc spravuje uživatelské účty v systému.

ER diagram

ER diagram znázorňuje svět jako množina objektů – entit a vztahů mezi nimi. Popisuje data v klidu, nezobrazuje operace, kterých se data účastní. Uzly znázorňují entitní množiny (data), hrany pak vztahy mezi nimi. Entitní množina je množina entit téhož typu. Každá entita má svůj název a

zpravidla několik atributů – vlastností. Jedním z atributů je identifikátor (primární klíč), který jednoznačně označuje danou entitu v rámci její entitní množiny. Vztah mezi entitami má název, kardinalitu a může mít atributy. Kardinalita určuje maximální počet vztahů, ve kterých se může vyskytovat jedna entita.



Obr. 3.2: ER diagram

Na obrázku 3.2 vidíme ER diagram vytvářeného systému. Základní entitou v systému je *Účet*. Obsahuje jak údaje potřebné pro identifikaci uživatele v rámci systému (přihlašovací jméno – login, zda je uživateli povoleno přihlášení do systému), tak údaje popisující uživatele v reálném světě (jméno, příjmení, adresu, telefon, email atd.). Každému účtu náleží entita *uživatel* pod kterou se návštěvník přihlašuje do systému se svým přihlašovacím jménem a heslem. Entita *účet* obsahuje *nákupní koš*, ve kterém jsou uloženy jednotlivé výrobky, jejich množství a cena.

Další důležitou entitou v systému je *výrobek – zboží*. Na základě výrobku, který má interní kód se provádějí objednávky. Výrobky mají název, cenu, obrázek, může obsahovat také delší popis. Výrobky dodávají entity *dodavatel* a *výrobce*, které mají svoje adresy a interní kódy.

Uživatelé systému zadávají objednávky, ty reprezentuje entita *objednávka*. Uchovává informace o množství, ceně a názvu vybraných výrobků. Jednotlivé objednávky identifikuje kód objednávky.

4 Návrh implementace

4.1 Volba implementačního prostředí

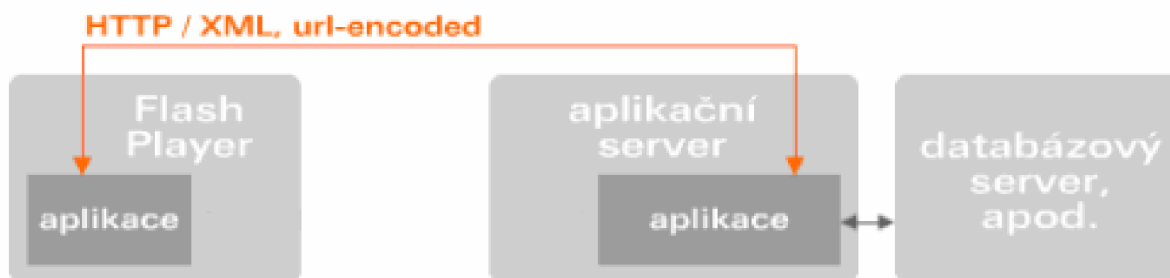
Analýzou požadavků na systém jsem dospěl k závěru, že data budou uložena v databázi na serveru, ke kterému se budou přes počítačovou síť připojovat klienti. Dále je požadována nenáročnost na výpočetní výkon a softwarové vybavení počítačů. Z těchto důvodů jsem se rozhodl implementovat systém jako webovou aplikaci, vytvořenou v prostředí Flash, která umožňuje využít řadu pokročilých grafických efektů. Toto řešení neklade na klientské počítače prakticky žádné požadavky, vyžaduje pouze přítomnost internetového prohlížeče a nainstalovaný Flash přehrávač. Provoz systému není na dnešních počítačích žádný problém.

Aplikace komunikuje s databází vytvořenou v prostředí Microsoft Office Access 2003. Jako server posloužil Macromedia ColdFusion MX7.

4.2 Blokový popis

Realizovaný systém elektronického obchodu můžeme rozdělit do několika funkčních bloků, jejichž popis bude následovat.

Pod pojmem "aplikační server" si představme libovolné programovací prostředí, ve kterém vytváříme aplikace na straně serveru, například PHP, J2EE, .NET. V mém případě jde o programovací prostředí ColdFusion, který zpracovává údaje z databáze a umožňuje jejich zobrazování v námi vytvořeném systému. Pro zjednodušení schématu považuji za součást aplikačního serveru i WWW server. Samotná aplikace se přehrává v bloku Flash Player ve formě .swf souboru. Celé blokové schéma je patrné na Obr. 4.1: Blokové schéma internetového obchodu



Obr. 4.1: Blokové schéma internetového obchodu

4.3 Návrh implementačního rozhraní

Uživatelé kromě nabízených funkcí zajímá také vzhled a způsob ovládání aplikace. Návrh uživatelského rozhraní je tedy důležitou součástí tvorby aplikace. Pokud je rozhraní špatně vytvořené a nepřehledné, znesnadňuje používání aplikace a uživatele znechucuje. Naopak přehledné a přívětivé uživatelské rozhraní práci s aplikací zjednodušuje, zrychluje a zkracuje čas potřebný pro naučení se ovládání aplikace. Uživatelské rozhraní by mělo uživatelům připadat známé, protože uživatelé se neradi učí nové věci.

Při tvorbě uživatelského rozhraní jsem vycházel ze zavedených zvyklostí. Stránka je rozdělena na jednotlivé funkční bloky. V záhlaví je menu s informacemi o obchodě. Levá a pravá strana jsou rozděleny do lišt. V pravé liště je menu, služby nabízené obchodem a prodávané značky. V pravé liště se nachází košík, který je umístěn vedle horního menu. Dále produktové speciality a nejprodávanější produkty. Stránka je zakončena patičkou. Pro zvýšení přehlednosti je většina funkčních bloků obchodu je odlišena barevným pozadím. Barvy jsem volil světlé, aby uživatele nerozptylovaly. Klád jsem důraz na dosažení dostatečného kontrastu barvy pozadí a textu, aby byl zobrazovaný text snadno čitelný.

V příloze 1 je zobrazena ukázka systému v následující podobě:

- Procházení kategoriemi výrobků – přehledová tabulka s výrobky (Obr. 0.1)
- Detail uživatelem vybraného výrobku (Obr. 0.2)
- Vložení výrobků do uživatelova košíku (Obr. 0.3)

4.4 Návrh struktury databáze

Prvním krokem implementace je návrh schématu databáze, stanovení její struktury a vytvoření tabulek. Při tomto procesu vycházíme z ER diagramu (obr. 3.2) vzniklého ve fázi analýzy. Nejprve vytvoříme logické schéma databáze. Na jeho základě potom v databázi vytvoříme příslušné tabulky.

Při tvorbě schématu databáze je třeba nejdříve odstranit složené a vícehodnotové atributy entit. Složený atribut se nahrazuje více atomickými (dále dělitelnými) atributy. Například atributy *jméno*, *příjmení* entity *účet* se rozloží na atributy tituly před jménem, jméno, příjmení a tituly za jménem. Vícehodnotové atributy se buď přetvoří na další entitní množinu nebo se nahradí pevným počtem opakování.

Dále je třeba zpracovat vztahy mezi entitami. Vztahy jsou realizovány na základě cizích klíčů. V praxi se řeší tak, že se do tabulky v databázi přidá sloupec obsahující hodnoty identifikátorů z jiné tabulky. Tento sloupec nazýváme cizím klíčem. Máme-li entitní vztah 1:1, můžeme sloupec cizí klíč

vložit do libovolné tabulky reprezentující dané entity. V případě vztahu 1:N obsahuje sloupec cizí klíč tabulka vyjadřující tu entitu, u které je konec vztahu označen N. Tabulka reprezentující entitu u konce vztahu s označením kardinality 1 je beze změn. Existuje-li mezi entitami vztah s kardinalitou N:N, je třeba vytvořit novou vazební tabulku. Tato pak obsahuje dva sloupce cizích klíčů, pro každou propojovanou entitu jeden. Vazební tabulka obsahuje také sloupce uchovávající atributy vztahu, pokud je třeba.

Databáze obsahuje následující tabulky:

- *Users* (tabulka s informacemi o uživatelských jménech, heslech a právech)
- *Account* (každý záznam reprezentuje osobní informace uživatele)
- *Cart* (kód, množství a cena vybraných výrobků)
- *Order* (záznamy s kódem objednávky, cenou a množstvím)
- *Product* (záznamy s nabízenými výrobky)
- *Suppliers* (informace o dodavatelích)
- *Maker* (informace o výrobcích)

4.5 Popis souborů systému

Kód systému je většinou umístěn v hlavní animaci. Kód pro komunikaci se serverem a kód ActionScript v samostatných třídách je rozčleněn do několika souborů. Především jsou to soubory pracující s jednotlivými tabulkami v databázi. Každou tabulku obsluhuje jeden soubor, není to však pravidlem. Tyto soubory obsahují kód v jazyce CFML. Dále třídy v jazyce Actionscript vkládané do animace. Vzhled stránek je popsán v animaci tvořené programem Flash.

4.5.1 Soubory obsluhované animací

Soubory s příponou *.**fla**, obsahující jednotlivé části animace, do kterých se načítají data z databáze a která se načítají do hlavního souboru **Gourvita.swf**.

LoginManager.cfc Třída *LoginManager*, která je implementována jako komponenta ColdFusion (CFC). Metody v ní vystavené, prověřují, zda má jméno uživatele a heslo odpovídající protějšky v databázi.

LoginXmlSource.cfm Obslužná stránka XML. Volá metody interně ze třídy *Loginmanager.cfc* a vystavuje výsledky jako pakety <response>, které podrobněji proberu v kapitole 4.6.

LoginMovieClip.as Hlavní animace obsahuje napojení na třídu *LoginMovieClip*. Třída kontroluje textová pole pro zadávání údajů o uživateli a kontroluje komponenty symbolu *LoginClip* a *LoginFormClip* pro řízení animace po přihlášení.

UserInfo.as Třída *UserInfo* je navržena tak, aby byla na straně klienta protějškem třídy datové struktury *LoginInfoObject*, která se na serveru používá ke sledování přihlašovacího stavu uživatele. její konstruktor přebírá jediný argument, o kterém se předpokládá, že je to odkaz na výsledek metody paketu `<response>` XML serveru. Vlastnosti `isLoggedIn`, `idUser`, `loggedInSince` a `sessionID` se kopírují z XML do nové instance třídy.

mailto.php Obslužná stránka, která odesílá ze stránky kontakt dotazy, na emailový účet administrátora.

4.6 Autentizace uživatele

Důležitým prvkem systému je autentizace uživatele, neboli ověření jejich identity. Uživatel svou identitu prokazuje zadáním přihlašovacího jména (loginu) a hesla.

Proces autentizace využívá proměnných sezení na serveru. To je prostředek umožňující animacím vytvořeným ve Flashi předávat data mezi jednotlivými úseky animace. Proces autentizace se skládá z kódu umístěným na serveru a kódu v animaci běžícím u klienta. U kódu na serveru moje verze implementuje:

Webovou stránku, která si vyměňuje data s XML s Flashem. Stránka přijímá v URL jediný parametr s názvem *action*, který lze nastavit na *getUserInfo*, *attemptLogin*, nebo na *logout*. Stránka vrátí XML, který obsahuje jediný prvek `<response>`. Flash pomocí informace z prvku `<response>` rozhodne, zda se má uživatel vyzvat, aby zadal své jméno a heslo, případně jaký obsah mu má zobrazit. Tato stránka se jmenuje *LoginXmlSource.cfm*.

Třidu *LoginManager*. Implementace třídy vystavuje tyto metody: *getUserInfo()*, *attemptLogin()* a *logout* (viz. tabulka 4.1). Úkolem těchto metod je prověřit, zda mají jméno uživatele a heslo odpovídající protějšky v databázi. ukládají také stav každého přihlášení uživatele do oboru sezení serveru. Třída je implementovaná v souboru *loginManager.cfc*.

Třidu *LoginInfoObject*. Jedná se o prostý objekt datové struktury s pojmenovanými vlastnostmi vztahujícími se k stavu přihlášení uživatele (*isLoggedIn*, *idLogin*, *sessionID*, *rightList* a *loggedInSince*). Tato pomocná třída je implementovaná v souboru *LoginInfoObject.cfc*.

Metoda	Popis
<i>attemptLogin(username, password)</i>	<i>Pokusí se uživatele přihlásit na základě zadaného jména a hesla. Jestliže se v tabulce Users najde souhlasný záznam, uloží se informace o uživateli do instance LoginInfoObject (v oboru sezení serveru), aby se zapamatovalo, že je uživatel přihlášený. Vrátí Booleovskou hodnotu, která vyjadřuje, zda bylo přihlášení úspěšné.</i>
<i>getUserInfo()</i>	<i>Vrátí objekt LoginInfoObject obsahující čtyři vlastnosti o aktuálním stavu přihlašovacího sezení: isLoggedIn, idUser, rightList a LoggedInSince. Návrátová hodnota odráží stav LoginInfoObject v oboru sezení serveru.</i>
<i>logout()</i>	<i>Resetuje informace v oboru sezení serveru, čímž aktuálního uživatele odhlásí.</i>

Tabulka 4.1: Metody, které vystavuje třída LoginManager

Implementace metod na serveru pracují s tabulkou *Users*. Jak fungují obslužné stránky XML

Stránka obsluhující data XML o aktuálním přihlašovacím stavu uživatele se jmenuje *LoginXMLSource.cfg*. Interně volá stránka metody ze třídy *LoginManager* (viz tabulka 4.1), které obstarávají mechanismus přihlášení a odhlášení uživatele.

Jestliže se stránka *LoginXMLSource* zavolá s parametrem *action=attemptLogin* v URL, očekává stránka, že jí bylo dodáno – ve formě XML – jako součást přicházejícího požadavku HTTP platné uživatelské jméno a heslo. Stránka očekává, že se jí doklady dodají ve formě velmi jednoduchého dokumentu XML, který obsahuje jen jeden prvek:

- `<credentials username="admin" password="flash" />`

Pomocí funkcionality týkající se rozkladu XML, kterou poskytuje server, bude stránka z balíčku `<credentials>` extrahovat uživatelské jméno a heslo. Pak zavolá metodu *LoginManager.attemptLogin()* uvedenou v tabulce 4.1. Jeli přihlášení úspěšné – tj. uživatelské jméno a heslo souhlasí s nějakým záznamem v databázi – odpoví stránka jiným jednoduchým balíčkem XML ve tvaru:

- `<response isLoggedIn="true">` nebo `<response isLoggedIn="false">`

Jestliže se stránka *LoginXmlSource* zavolá s parametrem *action=logout* v URL, zavolá interně metodu *LoginManager.logout()* a odpoví stejným balíčkem `isLoggedIn="false"`, jaký doprovází neúspěšné přihlášení.

Jestliže se stránka *LoginXmlSource* zavolá s parametrem *action=getUserInfo* v URL, odpoví poněkud delším paketem s informacemi o přihlašovacím stavu uživatele.

Kód u klienta:

Vlastní animace obsahuje několik textových polí, v nichž se zobrazují informace o stavu přihlášení uživatele. Jmenují se *txtUserID*, *txtRightList*, *txtSessionID* a *txtLoggedInSince*. Kód je přihlášení je připojený k události *onLoad*, vykoná se, když se SWF poprvé načte do přehrávače Flashe. Knihovna dokumentu obsahuje symbol animačního klipu s názvem *LoginClip*. Klip obsahuje dvě komponenty *XMLConnector* s názvy *xcGetUserInfo* a *xcAttemptLogin*. Obě jsou nakonfigurovány tak, aby kontaktovaly obslužné stránky XML.

Po přihlášení uživatele se vytvoří pomocí klíčového slova *new* nový objekt *UserInfo*. Výsledkem je objekt, v němž se budou udržovat data o uživateli obsažená uvnitř prvku *<response>*, který vrátil server. Pokud není uživatel aktuálně přihlášen, vykoná se větev *else*, kde se zviditelní prvky uživatelského rozhraní. Uživatel nyní může zadat své uživatelské jméno a heslo. Když pak klikne na tlačítko přihlásit, spustí se konektor *xcAttempt Login*, což způsobí, že se uživatelské jméno a heslo odešlou k ověření na server. Server odpoví prvkem *<response>*, který bude obsahovat buď *true*, jestliže se uživatel přihlásil úspěšně, nebo *false*, jestliže ne.

Když server dodá svou odpověď, vykoná se funkce *attemptLogin_Result()*. Která bude obsahovat booleovskou hodnotu vrácenou serverem. Jestliže uživatelské jméno a heslo nebyly platné, zobrazí se pomocí třídy *mx.controls.Alert* uživateli zpráva *Login Failed*.

4.7 Vývoj a testování

Systém byl vyvíjen v prostředí Macromedia Flash Professional 8. Jako server posloužil Macromedia ColdFusion MX7 a Microsoft Office Access 2003. Dále jsem testoval na HTTP serveru Apache 2.0.59 s PHP 5.2.1 a MySQL 5.0.37 běžícím na počítači s operačním systémem Windows XP. Jako internetové prohlížeče pro test posloužily Opera 9.20 a Mozilla Firefox 1.5.0.2.

5 Závěr

5.1 Dosažené výsledky

Cílem této bakalářské práce bylo navrhnout internetový obchod s možností procházení kategorií, výrobků a zadávání objednávek. Cíl této práce byl dosažen. Přístup uživatelů do systémů není podmíněn jejich přihlášením, přičemž jsou rozlišovány 2 typy uživatelů. Každý smí provádět jiné operace. Uživatel systému má právo prohlížet kategorie, výrobky a zadávat objednávky. Správce systému spravuje uživatelské účty a aktualizuje jednotlivé kategorie a databázi výrobků. Kromě toho smí vykonávat stejné činnosti jako uživatel.

Díky realizaci v podobě webové aplikace je systém nenáročný na výpočetní výkon a softwarové vybavení počítačů – klientů. Nutná je přítomnost přehrávače. Zvolené implementační prostředí umožňuje nasazení systému na většinu webových serverů.

Systémy elektronických obchodů jsou převážně vyvíjeny po důkladné analýze bodů uvedených v kapitole 3. Existuje pár univerzálních obchodů, ale převážná většina je stavěna na konkrétní výrobky a cílové zákazníky. Jako možná rozšíření systému vidím tato:

Uživatelsky měnitelný vzhled Implementace možnosti vybrat si jiný přednastavený vzhled aplikace. Uživatelé by si mohli vybrat různé barevné provedení aplikace a rozmístění ovládacích prvků. Konkrétně s prostředím Flashe se nabízí možnost plně nastavitelného vzhledu, kdy si uživatel uspořádá jednotlivé panely aplikace podle svého vkusu. Nastavení by se ukládalo do databáze ke každému uživateli a bylo by přístupné i po opětovném přihlášení do aplikace.

Možnost volby jazyka Lokalizace uživatelského prostředí ve více jazycích je dnes poměrně častým jevem. Flash nám nabízí úžasnou funkci, ve které jen přiřadíme každému textu vloženému do aplikace rozšíření a další jazyk, který můžeme poté lehce načíst např. z XML dokumentu a nakonec nesmíme zapomenout upravit i příslušnou měnu. Problém nastává s grafickými prvky, v tom smyslu, že musíme mít všechny dostupné s vybraným jazykem.

Statistiky zboží Podrobnější statistiky s přehledy nejhledanějších, nejprodávanějších a nejnavštěvovanějších produktů v obchodě

Security Implementace SSL (Secure Socket Layer). Obchodování na internetu se neobejde bez přítomnosti citlivých informací (čísla kreditních karet, jména, hesla atd.). Ať už chceme chránit nás, jako provozovatele, nebo zákazníka od zneužití citlivých informací, je na místě využít šifrování.

5.2 Přínos práce

Tvorba tohoto systému pro mě měla značný význam. Využil jsem v praxi své dosavadní znalosti z oblasti návrhu databází, jazyka SQL a tvorby webových stránek, animací v prostředí Macromedia Flash. Naučil jsem se pracovat s databází Access a MySQL v prostředí ColdFusion. Systém jsem vytvářel podle reálných požadavků. Cenné jsou pro mne nabyté zkušenosti s vývojem systému takového rozsahu.

Literatura

- [1] Zendulka, J.: Slajdy k předmětu IDS. 2006.
- [2] Weiss, N.: Flash MX PROFESSIONAL 2004 pro vývojáře webových aplikací. Brno, Zoner software 2004.
- [3] Sudacki, L.: Tipy a triky pro Macromedia Flash. Praha, Computer Press 2002.
- [4] Swann C., Cannes G.: Flash s využitím XML. Praha, Grada Publishing 2003.
- [5] Kosek, J.: XML pro každého. Praha, Grada Publishing 2000.
- [6] Cramer, L.: Building an Online Shopping Cart. Článek dostupný na [www: http://movielibrary.lynda.com/html/modPage.asp?ID=265](http://movielibrary.lynda.com/html/modPage.asp?ID=265)
- [7] Kofránek J., Stodulka P., Andrlík M., Kripner T.: Macromedia Flash – objektový nástroj pro tvorbu "inteligentních" vizualizačních komponent. Dokument dostupný na [www: http://objekty.pef.czu.cz/2004/sbornik/33_Kofranek.pdf](http://objekty.pef.czu.cz/2004/sbornik/33_Kofranek.pdf)
- [8] Jeffrey F. H.: Saving Movie Clip positions with PHP and MySQL. Článek dostupný na URL: [www.flash-db.com/Ex1/SaveMovieTutorial.php?](http://www.flash-db.com/Ex1/SaveMovieTutorial.php)

Seznam příloh

Příloha 1. Ukázka použití systému

Příloha 2. Paměťové médium - CD

Příloha 1:

GOURVITA GOURMET-SHOP

Our products

- Coffee & Espresso
 - Coffee-Nation
 - Pure Origins
 - Blends
 - Espresso
 - Flavored coffee
 - Limited Edition
 - Test-Sets

Services

- Corporate Service
- Gift Services
- Fun & Downloads

Brands

- 100% Espresso
- Cilles Seit 1873
- FIRST LIVING
- juta
- VISTAVINA
- MÖVENPICK
- COFFEE-PADS&E
- J.J. DARBOVEN
- 100% Espresso
- Alfredo ESPRESSO

About us | Contact | T&C | Customer service | Your Account

luxury class COFFEE-NATION

PURE ORIGIN

No coffee country is too far, no plantation's too small. We have found 'the Jewels'.

Come with us on a special coffee journey with lots of different flavors.

Pure Origins

<p>BRAZILIAN MILDAIR MILD ORIGINS</p> <p>Gourmet - Coffee Nicaragua Paramo € 13,90</p>	<p>NICARAGUA PARAMO STARK ORIGINS</p> <p>Gourmet - Coffee Guatemala Maya € 13,90</p>	<p>GUATEMALA MAJA SANFT ORIGINS</p> <p>Gourmet - Coffee Guatemala Maya € 13,90</p>	<p>HONDURAS ARABICA WEICH ORIGINS</p> <p>Gourmet - Coffee Honduras Arabica € 13,90</p>
<p>COSTA RICA TARAZZU REICH ORIGINS</p> <p>Gourmet - Coffee Costa Rica Tarazzu € 13,90</p>	<p>KENYA BLUE MOUNTAIN RASSIG ORIGINS</p> <p>Gourmet - Coffee Kenya Blue € 13,90</p>	<p>JAMAICA BLUE MOUNTAIN REIN ORIGINS</p> <p>Gourmet - Coffee Jamaica Blue € 43,90</p>	<p>COLOMBIA SUPREMO RASSIG ORIGINS</p> <p>Gourmet - Coffee Colombia Supremo € 13,90</p>
<p>HAWAII KONA EXTRA FANCY SANFT ORIGINS</p> <p>Gourmet - Coffee Hawaii Kona € 39,90</p>	<p>PAPUA NEW GUINEA SANFT ORIGINS</p> <p>Gourmet - Coffee Papua New Guinea € 13,90</p>	<p>TANZANIA MOSHI MILD ORIGINS</p> <p>Gourmet - Coffee Tanzania Moshi € 13,90</p>	<p>EL SALVADOR TIGRE SANFT ORIGINS</p> <p>Gourmet - Coffee El Salvador Tigre € 13,90</p>
<p>YEMEN MOKKA STARK ORIGINS</p> <p>Gourmet - Coffee Yemen Mokka € 13,90</p>	<p>MEXICO OAXACA MILD ORIGINS</p> <p>Gourmet - Coffee Mexico Oaxaca € 13,90</p>		

Your shopping cart today

Positions: 0
Total: €0,00
[order](#)

Your Products Specials

Chai
HOT, SPICY, CHAI
ADD SPICE TO YOUR LIFE

kaffee, präsent, werbung
FÜR FIRMEN

World of Coffee

[Show all Specials](#)

Bestseller

Bodum Pavina Espresso glass
€ 13,90

Mövenpick Espresso
€ 22,90


Gourmet - Espresso Grande Delizioso
€ 14,90

Jamie Oliver Pestle & Mortar
€ 39,90

Copyright 2007 Gourvita All Rights Reserved

Obr. 0.1: Procházení kategoriemi výrobků – přehledová tabulka s výrobky

About us
Contact
T&C
Customer service
Your Account



Our products



- ☐ Coffee & Espresso
 - ☐ Coffee-Nation
 - ☐ Pure Origins
 - ☐ Blends
 - ☐ Espresso
 - ☐ Flavored coffee
 - ☐ Limited Edition
 - ☐ Test-Sets

Services

- ☐ Corporate Service
- ☐ Gift Services
- ☐ Fun & Downloads

Brands









Show all products: [Pure Origins](#)

Gourmet - Coffee Jamaica Blue Mountain
 Model: XXX/COF/037001

from €43,90

[Any questions?](#)

250g ground	€43,90	1	Stk.		BUY
250g beans	€43,90	1	Stk.		BUY

Jamaica Blue Mountain is the most expensive coffee in the world - but is it the best coffee? The climate on the Jamaican island is unique in the world: humidity, sun hours, soil and winds give a perfect surrounding for the coffee trees. There is no doubt that the aroma of Jamaican coffee cannot be compared to others - so there is reason why people acquires the reputation of being the 'Rolls Royce' of coffee. At Coffee-Nation we have all the opinion that this is truly the best coffee world-wide.


We at Coffee-Nation love coffee - a passion you can taste. All our products are hand-selected and roasted at lowest temperatures for improved taste. Compared to industrial coffee, this process can take up to four times longer... but therefore it also tastes four times better! We focus on outstanding quality and we are convinced that every delivered package of Coffee-Nation helps to let the community of coffee lovers grow a little bigger. Our special aroma bag preserves the freshness for 12 months. So it is easy to store always a little amount of coffee or espresso at home without making compromises on taste.

Your shopping cart today

 Positions: 0
 Total: €0,00

[order](#)

Your Products Specials



kaffee, präsentе, werbung
FÜR FIRME



[Show all Specials](#)

Bestseller



Bodum Pavina Espresso glass
€13,90



Mövenpick Espresso
€22,90



Gourmet - Espresso Grande Delizioso
€14,90



Jamie Oliver Pestle & Mortar
€39,90

Copyright 2007 Gourvita All Rights Reserved

Obr. 0.2: Detail uživatelem vybraného výrobku

About us
Contact
T&C
Customer service
Your Account




Our products

- ☐ Coffee & Espresso
 - ☐ Coffee-Nation
 - ☐ Pure Origins
 - ☐ Blends
 - ☐ Espresso
 - ☐ Flavored coffee
 - ☐ Limited Edition
 - ☐ Test-Sets

Services

- ☐ Corporate Service
- ☐ Gift Services
- ☐ Fun & Downloads

Brands









Share delicious moments. It's cheaper for you.

We lowered shipping costs for orders exceeding 100 € value. Take advantage and share orders with your friends. Detailed shipping fees will be shown in your shopping cart when check-out.

Lower costs!

Image	delete	Quantity	Product	Price	Total
	<input type="checkbox"/>	1	Jamaica Blue Mountain 250g ground	€43,90	€43,90
	<input type="checkbox"/>	1	Jamaica Blue Mountain 250g beans	€43,90	€43,90
Subtotal: €87,80					

UPDATE CART
SHOP MORE
PROCEED NOW TO CHECKOUT

Your shopping cart today

Positions: 0
Total: €0,00

[order](#)

Your Products Specials



Chai
HOT SPICY CHAI
ADD SPICE TO YOUR LIFE!

kafee, präsenete, werbung
FÜR FIRMAN

World of
Coffee

[Show all Specials](#)

Bestseller



Bodum Pavina Espresso glass
€13,90



Mövenpick Espresso
€22,90



Gourmet - Espresso Grande Delizioso
€14,90



Jamie Oliver Pestle & Mortar
€39,90

Copyright 2007 Gourvita All Rights Reserved

Obr. 0.3: Vložení výrobků do uživatelského košíku

Příloha 2

Paměťové médium - CD