

Analýza a implementace moderních metod internetového obchodu

Modern Internet commerce method analysis and implementation

Tomáš Říha

Bakalářská práce
2013



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Tomáš ŘÍHA**
Osobní číslo: **A10057**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**
Forma studia: **prezenční**

Téma práce: **Analýza a implementace moderních metod
internetového obchodu**

Zásady pro vypracování:

1. Zpracujte literární rešerši na dané téma.
2. Popište možnosti tvorby internetového obchodu v aplikačním frameworku ASP.NET.
3. Analyzujte požadavky a vytvořte návrh internetového obchodu.
4. Vytvořte aplikaci internetového obchodu.
5. Demonstrujte výsledky a proveďte zhodnocení.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. NAGEL, Christian. Professional C 4 and .Net 4. Indianapolis, IN: Wiley Pub., c2010, 1474 s. ISBN 04-705-0225-8.
2. SPAANJAARS, Imar. Beginning ASP.NET 4 in C and VB. Indianapolis, IN: Wiley Pub., 2010, XXXVI, 803 s. Wrox beginning guides. ISBN 04-705-0221-5.
3. EVJEN, Bill, Scott HANSELMAN a Devin RADER. Professional ASP.NET 4 in C and VB. Indianapolis, IN: Wiley Pub., 2010, IVIII, 1477 s. ISBN 04-705-0220-7.
4. DORRANS, Barry. Beginning ASP.NET security. Chichester, U.K.: Wiley, 2010, XX, 412 s. ISBN 978-047-0743-652.
5. GOURLEY, David a Brian TOTTY. HTTP: The definitive guide. 1. Vyd. Sebastopol, CA: OReilly, 2002, XVIII, 635 s. ISBN 15-659-2509-2.

Vedoucí bakalářské práce:

Ing. Erik Král

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce:

24. února 2013

Termín odevzdání bakalářské práce:

14. června 2013

Ve Zlíně dne 24. února 2013

prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Bakalářská práce se zabývá analýzou a implementací moderních metod internetového obchodu. Pro vytvoření internetového obchodu byla zvolena technologie ASP.NET. Cílem práce je navrhnout internetový obchod umožňující administraci obsahu a analýzu výsledků prodeje.

Klíčová slova: ASP.NET, AJAX, e-shop, HTTP, HTTPS, MVC, MVP, MVVM, návrhový vzor, sociální síť, platební systém, SQL, zabezpečení, Web Mining, Web Usage Mining.

ABSTRACT

This thesis deals with the analysis and implementation of modern methods for an online store. ASP.NET technology was chosen for the creation of an online store. The aim of this thesis is to design an online store allowing content administration and sale results analysis.

Keywords: ASP.NET, AJAX, e-shop, HTTP, HTTPS, MVC, MVP, MVVM, software architecture pattern, social network, SQL, security, Web Mining, Web Usage Mining.

Rád bych poděkoval Ing. et Ing. Eriku Královi za vedení mé práce, vstřícný přístup a cenné rady, které mi poskytl během konzultací.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD.....	10
I TEORETICKÁ ČÁST	11
1 ASP.NET	12
1.1 ZÁKLADNÍ PRINCIPY A FUNKCE ASP.NET.....	12
2 ASP.NET AJAX.....	15
2.1 ZÁKLADNÍ PRINCIPY A FUNKCE ASP.NET AJAX.....	15
3 NÁVRHOVÉ VZORY	17
3.1 MVC.....	17
3.2 MVP	18
3.3 MVVM.....	19
3.4 SROVNÁNÍ NÁVRHOVÝCH VZORŮ	20
4 ZABEZPEČENÍ	21
4.1 DIGITÁLNÍ CERTIFIKÁTY	21
4.1.1 Certifikační autority v ČR	22
4.2 HTTPS.....	23
4.2.1 SSL Handshake	25
4.2.2 Serverové certifikáty	25
4.3 MOŽNOSTI ZABEZPEČENÍ ASP.NET APLIKACE.....	26
4.3.1 Autentizace založená na systému Windows.....	26
4.3.2 Autentizace založená na formulářích	27
4.4 APLIKAČNÍ SLUŽBY ASP.NET	28
4.5 OPEN WEB APPLICATION SECURITY PROJECT	29
5 PLATEBNÍ SYSTÉMY	32
5.1 PAYSEC.....	32
5.2 PAYU	33
5.3 PAYPAL.....	34
5.3.1 Express Checkout.....	34
5.3.2 PayPal Payments Standard	35
6 INTEGRACE SOCIÁLNÍCH SÍTÍ.....	37
6.1 FACEBOOK	37
6.1.1 Social Plugins.....	37
6.1.2 Login	42
6.1.3 Dialogs	43
6.1.4 Open Graph Protocol.....	44
6.2 GOOGLE+	45
6.2.1 Google + Plugins	45

6.3	TWITTER.....	46
6.3.1	Twitter Plugins	46
7	ANALÝZA A PREDIKCE CHOVÁNÍ UŽIVATELŮ	48
7.1	WEB MINING.....	48
7.1.1	Web Usage Mining.....	49
7.1.2	Sběr dat z úrovně serveru	49
7.1.3	Sběr dat z úrovně uživatele	50
II	PRAKTICKÁ ČÁST	52
8	ANALÝZA	53
8.1	DIAGRAM UŽITÍ	53
9	POPIS POUŽITÝCH NÁSTROJŮ.....	56
9.1	VISUAL WEB DEVELOPER 2010 EXPRESS.....	56
9.2	ADOBE PHOTOSHOP CS2.....	56
10	NÁVRH INTERNETOVÉHO OBCHODU	57
10.1	ROZMÍSTĚNÍ PRVKŮ.....	57
10.2	GRAFICKÝ NÁVRH	58
11	REALIZACE INTERNETOVÉHO OBCHODU.....	59
11.1	GRAFICKÝ NÁVRH	59
11.2	STRUKTURA DATABÁZE.....	60
11.3	UŽIVATELSKÉ ROZHRANÍ.....	62
11.4	ZABEZPEČENÍ APLIKACE.....	65
11.5	INTEGRACE SOCIÁLNÍCH SÍTÍ	69
11.6	ANALÝZA A PREDIKCE CHOVÁNÍ UŽIVATELŮ	69
12	POPIS INTERNETOVÉHO OBCHODU.....	71
12.1	KATALOG PRODUKTŮ	71
12.1.1	Detail produktu.....	71
12.2	NÁKUPNÍ KOŠÍK.....	72
12.3	ÚVODNÍ STRÁNKA	74
12.4	ČLÁNKY	76
12.4.1	Obsah článku	76
12.5	PROFIL.....	77
12.6	ADMINISTRACE.....	78
12.7	NÁMĚTY NA VYLEPŠENÍ.....	79
	ZÁVĚR	80
	CONCLUSION	82
	SEZNAM POUŽITÉ LITERATURY.....	84

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	87
SEZNAM OBRÁZKŮ	89
SEZNAM PŘÍLOH.....	91

ÚVOD

První kapitola je věnována základním principům technologie ASP.NET. Tato technologie umožňuje vytvářet dynamické webové stránky, které jsou díky kompilovatelnému kódu rychlejší a většinu chyb je možné odchytit již při vývoji webové stránky.

Druhá kapitola se zabývá základními principy technologie ASP.NET AJAX. Tato technologie umožňuje obnovovat pouze části webové stránky bez nutnosti jejího úplného obnovení.

Ve třetí kapitole jsou popsány návrhové vzory, které jsou alternativou k tradiční technologii ASP.NET založené na webových formulářích.

Obsahem čtvrté kapitoly je problematika zabezpečení webové stránky a komunikace mezi klientem a webovým serverem. K zajištění důvěryhodnosti webové stránky slouží digitální certifikáty, které umožňují uživatelům posoudit, zda informace, které odesílají, odesílají správnému subjektu.

Pátá kapitola se věnuje dostupným platebním systémům. Platební systémy umožňují potencionálním zákazníkům zaplatit za nakoupené zboží okamžitě po jeho objednání a urychlit tak proces jeho dodání.

Šestá kapitola obsahuje popis nástrojů, které poskytují sociální sítě. Sociální sítě jsou důležitým marketingovým nástrojem. Umožňují firmám a jiným soukromým subjektům působit na cílovou skupinu zákazníků a poskytovat jim informace.

Sedmá kapitola popisuje techniky k analýze a predikci chování uživatelů. Analýzou informací o pohybu uživatelů v internetovém obchodu lze získat velké množství důležitých informací, které mohou sloužit např. k stanovení marketingových plánů. Na základě této analýzy můžeme přizpůsobit obsah a strukturu tak, aby uživatel našel hledané informace co nejpohodlněji a nejrychleji.

Praktická část je věnována analýze požadavků a realizaci internetového obchodu.

I. TEORETICKÁ ČÁST

1 ASP.NET

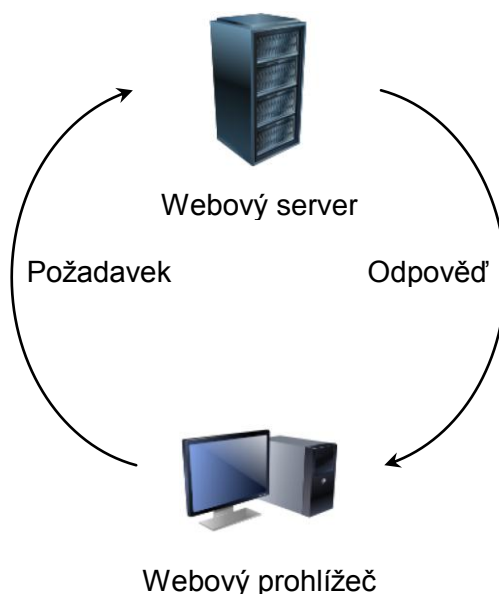
ASP.NET je technologií pro tvorbu webových aplikací. ASP.NET byl navržen společností Microsoft v roce 2002 jako součást .NET Frameworku¹. ASP.NET je nástupcem starší technologie ASP a nabízí vývojářům následující výhody:

- Výběr objektově orientovaného jazyka např. Visual Basic .NET, C# a další.
- Plně kompilovatelný kód.
- Oddělení prezentační vrstvy (HTML) od programového kódu.

[1]

1.1 Základní principy a funkce ASP.NET

Komunikace webového prohlížeče s webovým serverem probíhá prostřednictvím HTTP protokolu. Po zadání webové adresy do webového prohlížeče a jejím potvrzení, dojde k vytvoření požadavku. Požadavek je odeslán webovému serveru, který sídlí na zadané webové adrese. Pokud je webový server aktivní a požadavek je platný, webový server jej přijme, zpracuje a odešle odpověď webovému prohlížeči.



Obr. 1.1 Funkce HTTP protokolu

¹ Framework je softwarový balíček, obsahující nástroje pro tvorbu různých aplikací. [4]

Pokud se jedná o jednoduché statické soubory, jako jsou HTML stránky nebo obrázky, webový server je jednoduše přečte z místního pevného disku a odešle je do webového prohlížeče. Pro dynamické soubory jako jsou ASPX stránky, není toto zpracování dostačující. Pokud by webový server odeslal ASPX stránku obsahující programový kód, který se má zpracovat na webovém serveru, přímo do webového prohlížeče, nedošlo by k jeho zpracování. Proto namísto toho, aby webový server odeslal ASPX stránku přímo do webového prohlížeče, předává požadavek k dalšímu zpracování. Další zpracování požadavku probíhá prostřednictvím Application Mappingu² nebo Handler Mappingu³, kde jsou ASPX stránky mapovány do aplikace, která je schopna tento požadavek zpracovat. V případě ASPX stránek je požadavek zpracován prostřednictvím ASP.NET Runtime⁴, který je součástí .NET Frameworku.

Výsledný vzhled stránky, která se zobrazí ve webovém prohlížeči, je tak ovlivněn třemi hlavními prvky:

- *Statické soubory*: Statické soubory jakou jsou HTML stránky, soubory CSS nebo kód napsaný v JavaScriptu je odeslán přímo do webového prohlížeče.
- *Serverové prvky ASP.NET*: Serverové prvky uložené v ASPX stránkách jsou zpracovány, převedeny na HTML a znovu vloženy do stránky.
- *Programový kód*: Programový kód napsaný v Visual Basic .NET nebo C# může být vložený přímo do stránky, nebo může být umístěn v odděleném souboru tzv. Code Behind file. Tento programový kód může být zpracován automaticky při načtení stránky nebo na základě požadavku vyvolaným uživatelem. V každém případě, zpracování programového kódu může výrazně ovlivnit výsledný vzhled stránky, protože jeho prostřednictvím může dojít k přístupu do databáze, provedení výpočtů, zobrazení či skrytí ovládacích prvků atd.

² Application Mapping je komponentou IIS, která slouží k zjištění verze .NET Frameworku, kterou používá aplikace. [2]

³ Handler Mapping je komponentou IIS, která slouží k zpracovávání požadavků z různých zdrojů, které vrací uživateli informace. [3]

⁴ ASP.NET Runtime je komponentou odpovědnou za přijímání a zpracování požadavků ASPX stránek. [1]

Jakmile je zpracování stránky hotovo, výsledná HTML stránka je odeslána zpět do webového prohlížeče. Webový prohlížeč ji přečte, analyzuje a na konec ji zobrazí uživateli.

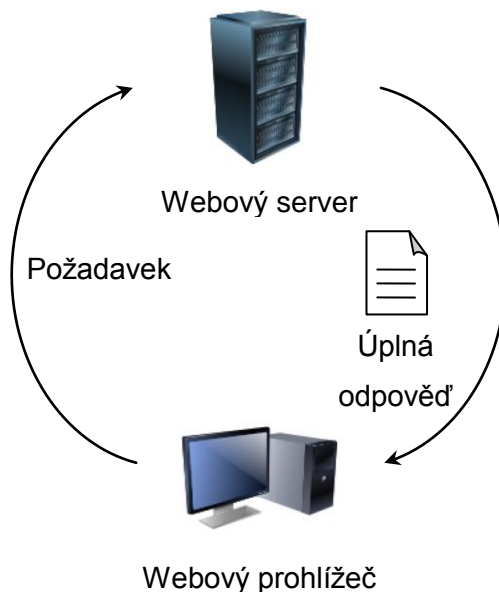
[1]

2 ASP.NET AJAX

ASP.NET AJAX je technologií pro tvorbu webových aplikací umožňující výměnu dat mezi webovým prohlížečem a webovým serverem prostřednictvím asynchronních volání. ASP.NET AJAX byl navržen společností Microsoft v roce 2005. Největší výhodou této technologie je tvorba stránek bez „prokliknutí“. [1]

2.1 Základní principy a funkce ASP.NET AJAX

V první kapitole bylo popsáno, jakým způsobem spolu komunikuje webový prohlížeč a webový server. Webový server zpracuje požadavek a odešle odpověď v podobě zpracované HTML stránky zpět do webového prohlížeče. Webový prohlížeč ji přečte, analyzuje a na konec ji zobrazí uživateli, případně stáhne další soubory, jako jsou CSS soubory nebo obrázky. Během interakce uživatele a stránky (např. kliknutím na tlačítko odeslat po vyplnění formuláře) je celá stránka odeslána na webový server a poté znovu načtena webovým prohlížečem.



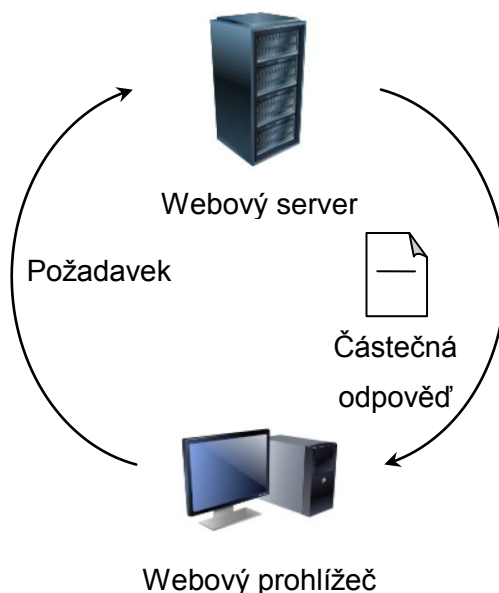
Obr. 2.1 Běžné zpracování stránky

Ačkoliv je tento model používán k zobrazování stránek po řadu let, obsahuje několik velkých nevýhod:

- Celá stránka musí být po odeslání na webový server znovu načtena webovým prohlížečem. Výsledná HTML stránka, přijatá webovým prohlížečem je tak zbytečně větší, než by bylo nutné. Protože, po vyplnění formuláře a kliknutí na

tlačítko odeslat, musí být znovu načteny i ty prvky, které se vůbec nezměnily, jako jsou např. ovládací prvky, hlavička stránky, patička stránky atd. Ideální by tedy bylo, kdyby byly přijaty webovým prohlížečem pouze ty prvky, u kterých došlo ke změně. V našem případě by webový server odeslal pouze formulář. Webový prohlížeč by tak obdržel pouze částečnou odpověď, která by byla použita k obnovení pouze těch prvků, které se změnily a ostatní prvky stránky by zůstaly nezměněny.

- Celá stránka musí být znovu vykreslena webovým prohlížečem. Po načtení webové stránky webovým prohlížečem, webový prohlížeč starou stránku zahodí a vykreslí novou stránku. Toto znovu vykreslení stránky způsobuje tzv. „prokliknutí“.



Obr. 2.2 AJAX zpracování stránky

ASP.NET AJAX tedy umožňuje:

- Vytváření stránek bez „prokliknutí“. Obnovují se pouze části stránky bez nutnosti obnovení celé stránky.
- Poskytnutí odezvy uživatelům během čekání na obnovení části stránky.
- Pravidelné aktualizace obsahu stránky a volání kódu na straně serveru pomocí časovače.

3 NÁVRHOVÉ VZORY

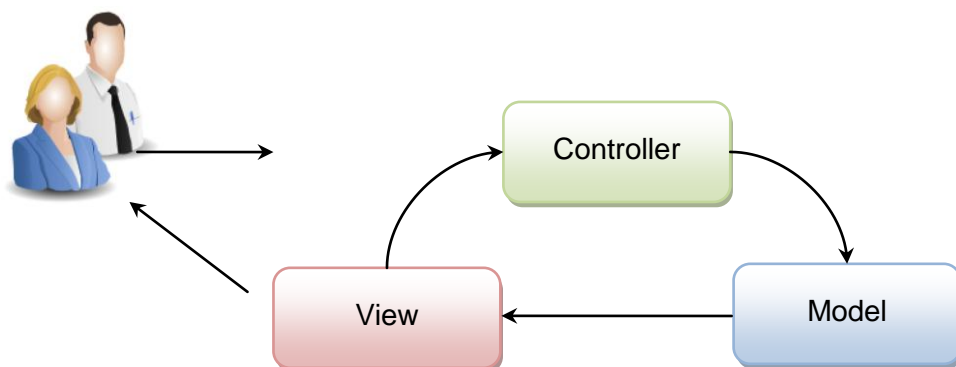
Návrhové vzory jsou alternativou k tradiční technologii ASP.NET založené na webových formulářích. Umožňují oddělit prezentační, aplikační a datovou vrstvu v aplikaci. Aplikace je obvykle rozdělena do třech částí tak, aby je bylo možné samostatně upravovat a aby byl dopad změn na ostatní části co nejmenší. [6]

3.1 MVC

Model View Controller (dále jen MVC) je návrhovým vzorem, který rozděluje aplikaci do třech logických částí:

- *Controller*: Controller představuje sadu tříd, které se starají o celkové provázání funkčnosti aplikace.
- *Model*: Model představuje sadu tříd, které popisují jak data, tak i doménovou logiku⁵.
- *View*: View definuje, jakým způsobem se Model zobrazuje uživateli.

[6]



Obr. 3.1 Architektura MVC

⁵ Doménová logika představuje metody pro práci s daty. [6]

Interakci aplikace s uživatelem je možné popsat v několika krocích:

1. Uživatel vykoná nějakou akci na uživatelském rozhraní.
2. Akce je zachycena Controllrem.
3. Controller rozhodne, jak na akci zareagovat a změní nějaké hodnoty v Modelu.
4. Model předá informace o změně View, které zobrazí změny uživateli.

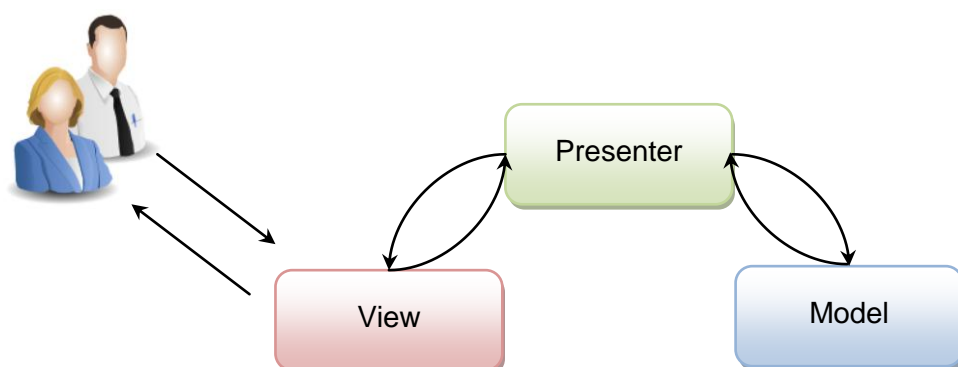
[7]

3.2 MVP

Model View Presenter (dále jen MVP) je návrhový vzorec, který rozděluje aplikaci do třech logických částí:

- *Model*: Model představuje sadu tříd, které popisují jak data, tak i doménovou logiku.
- *Presenter*: Presenter představuje sadu tříd, které popisují aplikační a prezentační logiku. Jinými slovy manipuluje s Modelem, což pomocí systému notifikací zajistí aktualizaci View.
- *View*: View zpracovává uživatelský vstup (volá metody Presenteru) a definuje, jakým způsobem se Model zobrazí uživateli.

[6]



Obr. 3.2 Architektura MVP

Interakci aplikace s uživatelem je možné popsat v několika krocích:

1. Uživatel vykoná nějakou akci na uživatelském rozhraní.
2. Akce je zachycena pomocí View a převedena na událost, která je předána Presenteru.
3. Presenter rozhodne, jak na akci zareagovat a změní nějaké hodnoty v Modelu.
4. Změněný Model předá informace o změně Presenteru.
5. Presenter předá informace o změně View, které zobrazí změny uživateli.

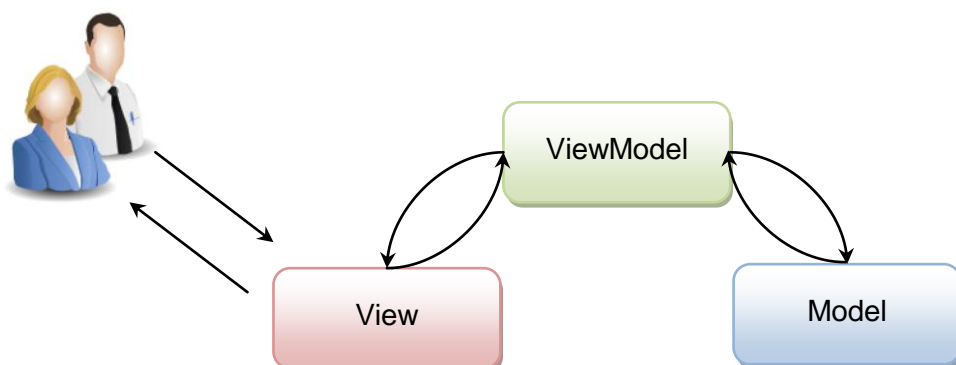
[7]

3.3 MVVM

Model View ViewModel (dále jen MVVM) je návrhovým vzorem, který rozděluje aplikaci do třech logických částí:

- *Model*: Model představuje sadu tříd, které popisují jak data, tak i doménovou logiku.
- *View*: View zpracovává uživatelský vstup (volá metody ViewModelu) a definuje, jakým způsobem se Model zobrazí uživateli.
- *ViewModel*: ViewModel je kombinací logických částí Model a Presenter.

[6]



Obr. 3.3 Architektura MVVM

Z funkčního hlediska je MVVM velice podobné MVP. Interakci aplikace s uživatelem je možné popsat v několika krocích:

1. Uživatel vykoná nějakou akci na uživatelském rozhraní.
2. Akce je zachycena pomocí View a převedena na událost, která je předána ViewModelu.
3. ViewModel rozhodne, jak na akci zareagovat a změní nějaké hodnoty v Modelu.
4. Změněný model předá informace o změně ViewModelu.
5. ViewModel notifikuje View o změnách.
6. ViewModel předá informace o změně View, které zobrazí změny uživateli.

[7]

3.4 Srovnání návrhových vzorů

Základní myšlenkou návrhového vzoru MVC je oddělení doménové logiky a grafického rozhraní na logické části Model a View. Dalším prvkem tohoto návrhového vzoru je Controller, který se stará o celkové provázání funkčnosti aplikace. Tento návrhový vzor je vhodný pro vývoj webových aplikací. View neobsahuje žádný programový kód, obsahuje pouze uživatelské rozhraní, zatímco Controller zpracovává akce vyvolané uživatelem. Společnost Microsoft vyvíjí ASP.NET MVC Framework pro tvorbu webových aplikací využívajících návrhového vzoru MVC.

Základní myšlenkou návrhového vzoru MVP je přidat logické části View více funkcí. Dalším prvkem tohoto návrhového vzoru je Presenter, který obsahuje události a stav View. Tento návrhový vzor je vhodný pro vývoj desktopových aplikací.

Základní myšlenkou návrhového vzoru MVVM je, aby View neobsahovalo žádný programový kód, pouze uživatelské rozhraní napsané v XAML. View tak může zobrazovat aktualizace a data aniž by obsahovalo nějaký programový kód. Tento návrhový vzor je vhodné pro vývoj desktopových aplikací využívající WPF. [8]

4 ZABEZPEČENÍ

4.1 Digitální certifikáty

Digitální certifikáty obsahují informace o subjektech např. firmách nebo uživateli, za které se zaručila důvěryhodná organizace (certifikační autorita). Běžné digitální certifikáty obsahují:

- Jméno subjektu např. firma, organizace, uživatel atd.
- Datum expirace.
- Certifikační autorita.
- Digitální podpis certifikační autority.

Navíc digitální certifikáty často obsahují veřejný klíč subjektu, informace o subjektu a informace o použitém kódovacím algoritmu.

Neexistuje žádný univerzální standart pro digitální certifikáty. Existuje mnoho rozdílných druhů digitálních certifikátů. Mnoho dnešních certifikátů, používá k ukládání informací standardní formu X.509 v3. X.509 v3 certifikáty umožňují ukládat strukturované informace o certifikátech do zpracovatelných polí. Existuje několik variant certifikátů založených na standardu X.509:

- Certifikáty certifikačních autorit.
- Certifikáty pro e-mailové klienty.
- Serverové certifikáty.
- Softwarové certifikáty
- Webové certifikáty.

Webové prohlížeče automaticky načítají digitální certifikáty serverů, ke kterým se připojují. Bezpečné spojení je navazováno prostřednictvím HTTPS. Pokud server nemá žádný certifikát, zabezpečené spojení se nevytvoří.

Když webový prohlížeč obdrží certifikát, tak nejdříve kontroluje jméno certifikační autority. Pokud se jedná o známou certifikační autoritu, prohlížeč je již vybaven jejím veřejným klíčem (webové prohlížeče obsahují certifikáty mnoha certifikačních autorit). Po

kontrole certifikační autority je kontrolován digitální podpis. Pokud se jedná o neznámou certifikační autoritu, prohlížeč si není jistý, zda může důvěřovat certifikační autoritě a obvykle zobrazí uživateli dialogové okno s oznámením, zda důvěřuje certifikační autoritě. Certifikační autoritou může být např. dodavatel softwaru, fyzická osoba nebo IT oddělení.

[9]

4.1.1 Certifikační autority v ČR

Seznam kvalifikovaných poskytovatelů certifikačních služeb je spravován Ministerstvem vnitra České republiky.

Seznam kvalifikovaných poskytovatelů certifikačních služeb:

- Česká pošta, s. p.
- eIdentity, a. s.
- První certifikační autorita, a. s.

[10]

Všechny výše uvedené certifikační autority poskytují následující služby:

- Vydávání kvalifikovaných certifikátů.
- Vydávání komerčních certifikátů.
- Vydávání kvalifikovaných časových razítek.

[11] [12] [13]

Podle [11] jsou výše uvedené služby popsány jako:

- *Kvalifikované certifikáty*: Osobní kvalifikovaný certifikát se využívá zejména při komunikaci s úřady státní správy.

Systémový kvalifikovaný certifikát není přímo vázán na konkrétní osobu. Je určen zejména pro organizace, které potřebují zautomatizovat elektronické podepisování např. odesílaných odpovědí na obdržené zprávy či výpisů z různých registrů.

- *Komerční certifikáty:* Komerční certifikáty mají širší oblast použití než kvalifikované certifikáty, které lze ze zákona použít jen pro první zmiňovaný účel.

Na rozdíl od kvalifikovaných certifikátů nejsou však komerční certifikáty automaticky uznávány. Obě komunikující strany se musí dohodnout např. smluvně, že budou důvěřovat komerční certifikační autoritě. Komerční certifikáty mohou být vydávány osobám i technologickým komponentám např. aplikace, zařízení, servery atd.

Komerční certifikáty lze využít například pro následující účely:

- Ověření elektronických podpisů.
 - Zajištění šifrované komunikace např. aktivace šifrovaného SSL spojení.
 - Autentizace uživatelů (přihlášení do aplikace pomocí certifikátu).
- *Kvalifikované časové razítko:* Jedná se o službu, která důvěryhodným způsobem spojuje data v elektronické podobě s časovým okamžikem a zaručuje, že uvedená data v elektronické podobě existovala v daný časový okamžik. Časové razítko důvěryhodným způsobem „razítkuje“ elektronický dokument v daném čase a je tedy vhodným doplňkem k elektronickému podpisu.

4.2 HTTPS

HTTPS je obvykle implementováno ve všech dostupných webových prohlížečích a serverech. Kombinuje HTTP protokol se sadou výkonných asymetrických, symetrických a na certifikátech založených kryptografických technik, které dělají HTTPS bezpečné, flexibilní a snadno upravovatelné.

HTTPS je v podstatě HTTP odesílané přes zabezpečenou transportní vrstvu. Namísto odesílání nezabezpečených HTTP zpráv do transportní vrstvy a potom do Internetu, HTTPS pošle HTTP zprávu do zabezpečující vrstvy, která ji zašifruje před odesláním do transportní vrstvy a potom do Internetu.



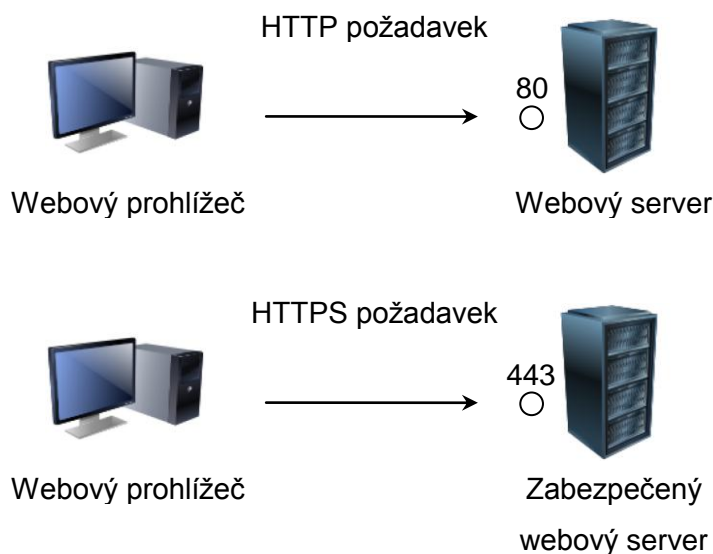
Obr. 4.1 Vrstvy HTTP a HTTPS

Dnes je zabezpečující vrstva HTTP implementována prostřednictvím SSL a nebo jeho nástupcem TLS. Volba zabezpečené verze HTTP respektive HTTPS je volitelná. Při vytváření požadavku pro webový server, musíme webovému serveru oznámit, že budeme používat HTTPS. Oznámení se provádí prostřednictvím webové adresy. Pokud používáme HTTP, webová adresa obsahuje předponu `http://`. Pokud používáme HTTPS, webová adresa obsahuje předponu `https://`. Když je uživatel požádán o provedení transakce na webové stránce dojde k vyhodnocení webové adresy:

- Pokud má webová adres předponu `http://`, je otevřeno spojení se serverem na portu⁶ 80 a jsou odeslány HTTP příkazy.
- Pokud má webová adres předponu `https://`, je otevřeno spojení se serverem na portu 443 a dojde k handshaku se serverem. Prostřednictvím handshaku dojde k výměně zabezpečujících parametrů SSL a HTTP příkazů v binárním formátu.

Protože je SSL binárním protokolem, zcela odlišným od HTTP, je komunikace prováděna na různých portech. Pokud jsou oba protokoly prováděny na portu 80, většina webových severů interpretuje binární tvar SSL jako chybné HTTP a ukončí komunikaci. [9]

⁶ Port je číslo označující specifickou aplikaci nebo proces při komunikaci. [9]



Obr. 4.2 Komunikace HTTP a HTTPS

4.2.1 SSL Handshake

Před odesláním zabezpečeného HTTP požadavku je nutné provést handshake. Handshake slouží k:

- Výměně čísel verzí protokolů.
- Výběru šifry, kterou znají obě strany.
- Ověření identity každé strany.
- Vygenerování dočasného klíče k šifrovanému kanálu.

Dočasný klíč k šifrovanému kanálu je vytvořený na základě zvoleného šifrovacího algoritmu. [9]

4.2.2 Serverové certifikáty

SSL podporuje vzájemné vyměňování certifikátů mezi uživatelem a serverem a serverem a uživatelem. Dnes se uživatelské certifikáty k prohlížení webových stránek obvykle nepoužívají, protože většina uživatelů certifikát nevlastní. Webový server může po uživateli požadovat certifikát, ale tato situace se vyskytuje jen zřídka.

Na druhou stranu, HTTPS transakce vždy vyžadují serverové certifikáty. Při provádění zabezpečené transakce na webovém serveru, jako je např. odesílání informací o kreditní kartě, chtějí uživatelé vědět, že informace, které odesílají, odesílají správnému subjektu.

Serverové certifikáty podepsané známými certifikačními autoritami, pomáhají uživatelům posoudit, jak moc mohou věřit subjektu, kterému odesílají informace.

Serverové certifikáty, využívající standardní formu X.503 v3 obsahují informace o jménu subjektu, adrese, doménovém jménu atd. [9]

4.3 Možnosti zabezpečení ASP.NET aplikace

Technologie ASP.NET disponuje několika základními druhy autentizačních metod jako je:

- Autentizace založená na systému Windows.
- Autentizace založená na formulářích.

[4]

Problematika týkající se zabezpečení aplikace je podrobně popsána v následujících publikacích [14] a [15].

4.3.1 Autentizace založená na systému Windows

Využívání autentizačních metod založených na systému Windows je podmíněno používáním platformy Windows.

- *Integrovaná Windows autentizace:* Integrovaná Windows autentizace vyžaduje k autentizaci uživatele uživatelské jméno a heslo. Uživatelské jméno a heslo je před odesláním na server převedeno na hash⁷. Tato metoda se využívá v soukromých sítích a uživatelé musí používat webový prohlížeč Internet Explorer.
- *Základní autentizace:* Základní autentizace vyžaduje k autentizaci uživatele uživatelské jméno a heslo. Uživatelské jméno a heslo není před odesláním na server zašifrováno. K zabezpečení uživatelského jména a hesla je možné použít SSL. Tato metoda je součástí HTTP protokolu a je podporována většinou prohlížečů.

⁷ Hash je výstupem hashovací funkce (jednosměrný šifrovací algoritmus). [9]

- *Autentizace pomocí metody digest:* Autentizace pomocí metody digest vyžaduje k autentizaci uživatele uživatelské jméno a heslo. Uživatelské jméno a heslo je před odesláním na server zašifrováno. Abychom mohli používat tuto metodu autentizace, musíme mít Windows Domain Controller⁸ a webové prohlížeče musí podporovat HTTP 1.1.

[4]

4.3.2 Autentizace založená na formulářích

Využívání autentizačních metod založených na formulářích je podporováno na různých platformách.

Autentizace pomocí formulářů je jednou z nejoblíbenějších metod autentizace, pomocí které můžeme uživatelům poskytnout přístup k celé webové stránce nebo jen k jejím částem. Tato metoda používá k přihlašování klasický přihlašovací formulář, do kterého uživatel zadává uživatelské jméno a heslo. Uživatelské jméno a heslo není před odesláním na server zašifrováno. K zabezpečení uživatelského jména a hesla je možné použít SSL nebo TLS. [4]

K bezpečnému ukládání hesel do databáze lze použít hashovací algoritmus. Hashovací algoritmus je kryptografickou funkcí, která je používána k vytváření bezpečného otisku dat.

Vlastnosti hashe:

- *Stejná délka:* Délka hashe se mění v závislosti na použitém hashovacím algoritmu. Výsledná délka hashe je nezávislá na délce vstupních dat.
- *Deterministický algoritmus:* Výsledný hash je vždy stejný pro stejná vstupní data.
- *Jednosměrnost:* Data nemohou být rekonstruována z hashe.
- *Jednoznačnost:* Hashovací algoritmus by měl produkovat naprosto jiný výsledný hash i přesto, že se vstupní data změnila v jediném bitu.

[5]

⁸ Windows Domain Controller je server, který odpovídá na požadavky o autentizaci. [4]

4.4 Aplikační služby ASP.NET

V případě, že nám žádná z výše uvedených autentizačních metod nevyhovuje, je možné vytvořit vlastní autentizační metodu, ale z bezpečnostních důvodů se to nedoporučuje.

Autentizací je myšleno poskytování údajů o své identitě. V případě webových stránek se k autentizaci uživatele nejčastěji používá e-mail a heslo případně uživatelské jméno a heslo. V závislosti na naší identitě nám pak webové stránky poskytují různá práva, jak přistupová, tak k provádění specifických operací.

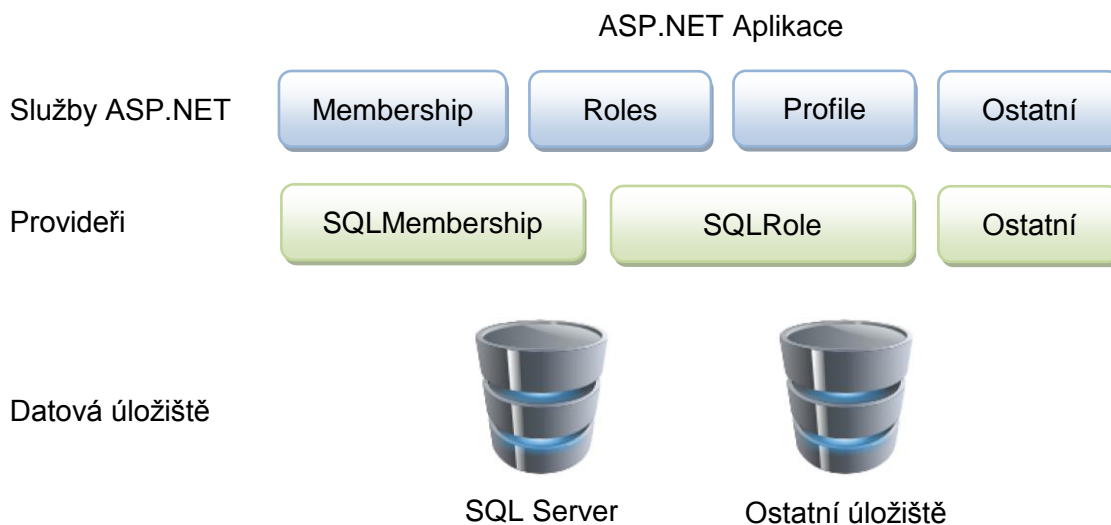
Aby webové stránky mohli rozhodnout, jaká práva mají udělit uživateli, musí znát oprávnění vztahující se na daného uživatele. Jednotlivá oprávnění mohou být založena na uživatelském jménu nebo skupině uživatelů, do které je uživatel zařazen. Obsah webových stránek pak může být blokován/zpřístupněn pro specifického uživatele nebo skupinu uživatelů.

Technologie ASP.NET poskytuje spoustu aplikačních služeb, mezi ty nejzákladnější patří:

- *Membership*: Umožňuje spravovat uživatelské účty.
- *Roles*: Umožňuje spravovat skupiny uživatelů.
- *Profile*: Umožňuje spravovat specifické informace o uživatelích.

Výše zmíněné aplikační služby nekomunikují přímo s příslušným zdrojem dat, ale s nakonfigurovaným providerem⁹. Například SQL Server Membership provider je navržen pro práci s uživateli jako je např. přihlašování, vytváření uživatelů nebo resetování hesla. SQL Server Role provider je navržen pro práci se skupinami uživatelů. V jedné aplikaci je možné nakonfigurovat více providerů v závislosti na našich potřebách. Výhodou používání různých providerů je možnost jejich výměny bez nutnosti provádění změn v programovém kódu.

⁹ Provider je poskytovatel služeb. [1]



Obr. 4.3 Schéma webové stránky

Různé režimy autentizace se nastavují prostřednictvím web.config souboru¹⁰. [1]

Webové stránky mohou využívat celou řadu dalších providerů, které je možné stáhnou a nainstalovat prostřednictvím vývojového prostředí. Například součástí HTML Agility Pack je HtmlAgilityPackSanitizerProvider, který zabraňuje vkládání potenciálně nebezpečných programových kódů do webové stránky.

4.5 Open Web Application Security Project

Open Web Application Security Project (dále jen OWASP) je nezisková organizace zaměřená na zabezpečení webových aplikací. [14]

Jedním z projektů OWASP je OWASP Top Ten. Tento projekt popisuje deset nejzávažnějších bezpečnostních hrozeb:

- *Injection*: Injekce škodlivého kódu spočívá v odeslání nebezpečného programového kódu prostřednictvím nezabezpečeného webového formuláře nebo databázového dotazu. Útočník tak může provádět nežádoucí databázové dotazy nebo získat přístup k soukromým údajům.

¹⁰ Web.config soubor je konfiguračním souborem založený na XML, který slouží ke konfiguraci webové aplikace. [1]

- *Cross-Site Scripting (XSS)*: Vkládání skriptů do stránek spočívá v odeslání nebezpečného programového kódu prostřednictvím nezabezpečeného webového formuláře nebo URL. Útočník tak může spouštět nebezpečné skripty v prohlížeči, napadat uživatelské relace, poškozovat webové stránky nebo přesměřovat uživatele na nebezpečné stránky.
- *Broken Authentication and Session Management*: Narušení autentizace a relace spočívá v napadání session¹¹ vytvořené mezi uživatelem a serverem. Session se vytváří automaticky např. při přihlášení uživatele do internetového obchodu. K jejímu uložení se používá cookies nebo session ID, které se vloží jako parametr do URL. Útočník tak může získat přístup k cizím účtům a vydávat se za jiné uživatele.
- *Insecure Direct Object References*: Nezabezpečený přímý odkaz na objekt spočívá v napadení URL, které používá k odkazování názvy adresářů nebo primární klíče z databáze. Útočník tak může získat přístup k adresářům, soborům nebo informacím bez nutné autentizace.
- *Cross-Site Request Forgery (CSRF)*: Podvržení požadavku webové aplikace spočívá ve vložení neviditelného prvku, nejčastěji obrázku do webové aplikace, kam mají přístup pouze registrovaní uživatelé. Neviditelný prvek obsahuje programový kód, který může vykonat nějaký požadavek. Když uživatel vstoupí na tuto stránku, dojde k automatickému vykonání požadavku.
- *Security Misconfiguration*: Bezpečnost webové aplikace závisí na konfiguraci a zabezpečení webového serveru, databázového serveru, případně dalších služeb. Konfiguraci a zabezpečení provádějí obvykle poskytovatelé webhostingu. Vývojáři webových aplikací mohou zvýšit bezpečnost své webové aplikace používáním nejnovějších verzí .NET Frameworku.

¹¹ Session je označení pro relaci mezi uživatelem a serverem. Session cookie se smaže, pokud uživatel vypne webový prohlížeč. [9]

- *Insecure Cryptographic Storage*: Uživatelé přistupují ke svým účtům prostřednictvím uživatelského jména a hesla. Uživatelské jméno a heslo se ukládá do databáze. Pokud není heslo dostatečně zabezpečeno, v případě úspěšného SQL injection útoku by útočník získal uživatelská jména a hesla ke všem účtům. K bezpečnému ukládání hesel do databáze se používá hashovací algoritmus. Hashovací algoritmus zajistí, že do databáze není uloženo heslo v textové podobě, ale ukládá se jeho otisk.
- *Failure to Restrict URL Access*: Chybné zamezení přístupu ke konkrétním URL spočívá v nedokonalé kontrole přístupu k adresářům/souborům/stránkám určité skupiny uživatelů. Útočník tak může přistupovat k adresářům/souborům/stránkám a může provádět neoprávněné změny.
- *Insufficient Transport Layer Protection*: Webové aplikace, které nepoužívají zabezpečenou komunikaci mezi uživatelem a webovým serverem, vystavují data odesílané uživatelem nebezpečí jejich zneužití. Útočník tak může sledovat síťový provoz např. na bezdrátové síti a odchytnout citlivé údaje nebo session cookie a může přebrat relaci uživatele.
- *Unvalidated Redirects and Forwards*: Webové aplikace často přesměrovávají uživatele na jiné webové stránky pomocí nedůvěryhodných dat k určení cílové webové stránky. Útočník tak může bez řádného ověření přesměrovat uživatele na malware¹² nebo phishing¹³ stránky.

[15] [16]

¹² Malware je škodlivý program, který může poškodit počítač uživatele, nebo odcizovat uživatelské údaje. [5]

¹³ Phishing je technika k získávání uživatelských údajů, která využívá kopií webových stránek různých společností. [5]

5 PLATEBNÍ SYSTÉMY

5.1 PaySec

PaySec je platební brána zprostředkovávající platby na Internetu. Platební brána umožňuje zákazníkovi výběr mezi více platebními metodami:

- Platba z PaySec konta zákazníka na PaySec konto obchodníka.
- Platba z běžného účtu klienta u ČSOB na PaySec konto obchodníka.
- Platba z běžného účtu klienta u Poštovní spořitelny/Era na PaySec konto obchodníka.
- Platby platebními kartami asociací MasterCard, VISA a Diners Club na PaySec konto obchodníka.
- Platba převodem z běžného účtu v libovolné bance nebo složenkou na PaySec konto obchodníka.

Pokud zákazník použije k zaplacení platební bránu PaySec, internetový obchod přesměruje zákazníka na platební bránu a předá jí informace o požadované platbě. Přesměrování musí být provedeno tak, aby se rozcestník platebních metod PaySec zobrazil přes celé okno prohlížeče a v URL bylo uvedeno URL platební brány PaySec. Zákazník si na platební bráně vybere platební metodu. V závislosti na vybraném způsobu platby je zákazníkovi zobrazen formulář pro přihlášení k PaySec kontu, nebo je přesměrován do internetového bankovníctví. V rámci přesměrování jsou předány všechny informace k provedení platebního příkazu. Po provedení platby je uživatel přesměrován zpět na adresu, která byla předána platební bráně jako jeden z parametrů. Platební brána PaySec zašle zpět identifikaci provedené platby nebo informaci o zamítnutí platby zákazníkem. Pokud bylo vráceno číslo provedené platby, internetový obchod si prostřednictvím webové služby ověří, že byla transakce skutečně provedena. Tento krok je nezbytný k bezpečnému ověření skutečného stavu provedení transakce. Ověření transakce musí být provedeno ihned po zavolání návratového URL internetového obchodu a to tak rychle, aby internetový obchod mohl na stránce, kterou klientovi zobrazí po jeho návratu z platební brány PaySec, zobrazit odpověď, kterou získal pomocí tohoto ověření transakce. Obvykle během jednotek sekund.

Zákazníkovi je zobrazena informace o úspěšném nebo neúspěšném provedení transakce a zaplacení zboží. [17]

5.2 PayU

PayU je platební brána zprostředkovávající platby na Internetu. Platební brána umožňuje zákazníkovi výběr mezi více platebními metodami:

- Platba z běžného účtu klienta u České spořitelny.
- Platba z běžného účtu klienta u Fio banky.
- Platba z běžného účtu klienta u GE Money Bank.
- Platba z běžného účtu klienta u Komerční banky.
- Platba z běžného účtu klienta u mBank.
- Platba z běžného účtu klienta u Raiffeisenbank.
- Platba z běžného účtu klienta u Volksbank.
- Platby platebními kartami asociací MasterCard, VISA a Diners Club.
- Platby na terminálech Sazka a přepážkách České pošty.
- Běžný bankovní převod.
- Platba poštovní poukázkou.
- Mobilní platby Mobito.

Pokud zákazník použije k zaplacení platební bránu PayU internetový obchod přesměruje zákazníka na platební bránu a předá jí informace o požadované platbě. Zákazník si na platební bráně vybere platební metodu. V závislosti na vybraném způsobu platby je zákazník přesměrován do internetového bankovníctví nebo do jiného platebního systému. V rámci přesměrování jsou předány všechny informace k provedení platebního příkazu. Potvrzená platba od zákazníka je následně převedena prostřednictvím platebního kanálu na účet určený pro službu PayU. Nyní může obchodník platbu od zákazníka přijmout nebo odmítnout. Po provedení platby je uživatel přesměrován zpět na adresu, která byla předána platební bráně jako jeden z parametrů. [18]

5.3 PayPal

PayPal je platební systém zprostředkovávající platby a převody peněz na Internetu. K přijímání a zpracovávání plateb z internetového obchodu můžeme použít:

- Express Checkout
- PayPal Payments Standard

[19]

5.3.1 Express Checkout

Express Checkout umožňuje prodejcům minimalizovat počet nutných kroků k dokončení platby. Výhody Express Checkout:

- Přijímání plateb z jakéhokoliv PayPal účtu.
- Eliminace nutnosti zadávání osobních údajů nebo platebních informací na straně internetového obchodu.
- Přesměrování zákazníka na stránky prodejce po dokončení transakce.

Pokud zákazník použije k zaplacení objednávky Express Checkout, internetový obchod vykoná API¹⁴ volání, prostřednictvím kterého předá PayPalu podrobnosti o platbě. Zákazník je přesměrován na stránky PayPalu, kde je vyzván k přihlášení. Po přihlášení vybere způsob platby, vyplní dodací adresu a potvrdí platbu. PayPal potom přesměruje zákazníka do internetového obchodu, aby zkontroloval a potvrdil objednávku. Pokud je objednávka potvrzena, internetový obchod vykoná další API volání, aby získal další informace o objednavce včetně dodací adresy, e-mailové adresy a dalších informací o zákazníkovi. Prostřednictvím dalšího API volání dojde k vytvoření požadavku na úhradu objednávky a zákazníkovi je e-mailem odesláno potvrzení o zaplacení.

¹⁴ API představuje rozhraní pro programování aplikací. V tomto kontextu dojde k zavolání některé z metod, které nabízí API.

Další možnosti Express Checkout:

- *Platba prostřednictvím mobilních zařízení:* Express Checkout je optimalizován i pro platby prostřednictvím mobilních zařízení.
- *Přizpůsobení vzhledu PayPal stránky:* PayPal stránka, na kterou je uživatel přesměrován, může být přizpůsobena vzhledu internetového obchodu.
- *Správa prodeje:* Prodejci mohou nastavovat, jak jsou předvyplňována různá pole, sledovat statistiku prodeje, ukládat objednávky do databáze, zpracovávat opakující se platby, vyhledávat platby atd.

[19]

5.3.2 PayPal Payments Standard

PayPal Payment Standard umožňuje prodejcům přidávat na svoje stránky různá tlačítka k platbě nebo k přidávání zboží do nákupního košíku atd. Výhody PayPal Payment Standard:

- Vytvoření různých druhů tlačítek.
- Přidávání tlačítek na webové stránky prostřednictvím HTML kódu.
- Přijímání plateb z kreditních nebo debetních karet, případně z PayPal účtu.

Tlačítka PayPal Payment Standard se vkládají do stránky prostřednictvím HTML kódu. Jednotlivá tlačítka odkazují zákazníky přímo na stránky PayPalu. Na stránkách PayPalu je zákazník vyzván k přihlášení se na svůj PayPal účet, kde dokončí platbu. Po dokončení platby je zákazníkovi odesláno e-mailem potvrzení o zaplacení objednávky.

Další možnosti PayPal Payment Standard:

- *Doprava:* Tisk štítků s adresou, nastavení pojištění zásilek, sledování zásilek, potvrzení o doručení zásilky atd.
- *Přesměrování zákazníků do internetového obchodu:* Po dokončení platby může být zákazník opět přesměrován do internetového obchodu.
- *Přizpůsobení vzhledu PayPal stránky:* PayPal stránka, na kterou je uživatel přesměrován, může být přizpůsobena vzhledu internetového obchodu.

- *Řízení zásob*: Vytváření upozornění v závislosti na stavu zásob.
- *Správa prodeje*: Prodejci mohou nastavovat, jak jsou předvyplněna různá pole, sledovat statistiku prodeje, ukládat objednávky do databáze, zpracovávat opakující se platby, vyhledávat platby atd.

[19]

6 INTEGRACE SOCIÁLNÍCH SÍTÍ

6.1 Facebook

Sociální síť Facebook nabízí k integraci svých služeb softwarové prostředí Facebook Platform. Základní součásti Facebook Platform pro webové stránky jsou:

- Social Plugins
- Login
- Dialogs
- Open Graph Protocol

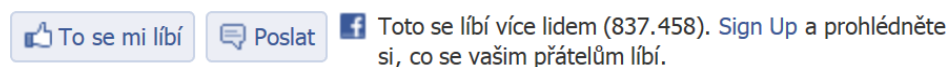
[20]

6.1.1 Social Plugins

Social Plugins umožňují přidávání prvků prostřednictvím IFRAME tagů a JavaScriptu. Výhodou tohoto řešení je snadná integrace kódu do webové stránky. Kód každého prvku je vygenerován prostřednictvím konfiguratoru.

Součástí Social Plugins je:

- *Like Button*: Like Button umožňuje uživatelům sdílet obsah s přáteli. Po kliknutí na tlačítko Like Button, umístěné na webové stránce, se v kanálu novinek uživatele automaticky zobrazí zpráva s odkazem na webovou stránku.



Obr. 6.1 Like Button [21]

- *Send Button*: Send Button umožňuje uživatelům posílat obsah jejich přátelům. Uživatel má možnost odeslat odkaz na webovou stránku svým přátelům prostřednictvím zprávy, zprávy na „zed“ skupiny nebo jako e-mail na libovolnou e-mailovou adresu. Zatímco Like Button umožňuje sdílení obsahu se všemi přáteli, prostřednictvím Send Button je možné sdílet obsah s vybranou skupinou přátel.

Zpráva obsahuje odkaz na webovou stránku, název, obrázek a krátký popis. Pomocí Open Graph protokolu, je možné specifikovat, jaký se zobrazí název, obrázek nebo popis.



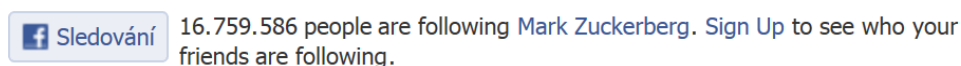
Obr. 6.2

Send

Button

[21]

- *Follow Button*: Follow Button umožňuje uživateli sledovat veřejné aktualizace Facebook stránky.



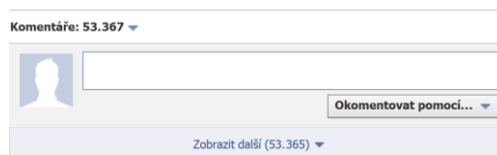
Obr. 6.3 Follow Button [21]

- *Comments Box*: Comments Box umožňuje uživateli komentovat obsah webové stránky.

Comments Box obsahuje funkce pro distribuci a moderování obsahu. Komentáře jsou seřazeny tak, aby se uživatelům zobrazovaly nejrelevantnější komentáře od přátel, přátel přátel, komentáře co se nejvíce „líbí“ nebo aktivní diskusní vlákna. Zatímco komentáře označené jako SPAM jsou uživateli skryty.

Komentáře lze snadno sdílet s přáteli nebo s ostatními uživateli, kterým se daná stránka „líbí“. Pokud uživatel označí pole Post to Facebook a odešle komentář, zobrazí se zpráva v jeho kanálu novinek, s odkazem na danou stránku, informující o tom, že uživatel napsal na dané stránce komentář.

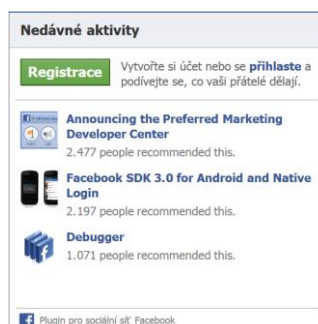
Přátelé a ostatní uživatelé, kterým se daná stránka „líbí“, mohou reagovat na diskuzi prostřednictvím Like Button, prostřednictvím kanálu novinek na Facebooku nebo přímo z dané stránky. Diskusní vlákna se automaticky synchronizují na všech místech, kde se zobrazují.



Obr. 6.4 Comment Box [21]

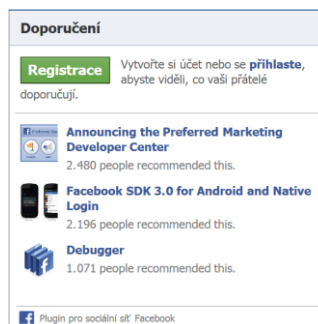
- *Activity Feed*: Activity Feed zobrazuje na dané stránce nedávné aktivity a nejzajímavější události.

Vzhledem k tomu, že je obsah hostován Facebookem, může Activity Feed zobrazovat personalizovaný obsah v závislosti na tom, zda je nebo není uživatel přihlášený. Activity Feed zobrazuje události, způsobené interakcí uživatele a dané stránky např. kliknutí na Like Button, přečtení článku, shlédnutí videa, když uživatel sdílí obsah z dané stránky na Facebooku, nebo když napíše komentář do Comment Boxu na dané stránce. Pokud je uživatel přihlášený, Activity Feed je personalizován a zvýrazní se obsah vytvořený přáteli. Pokud je uživatel odhlášený, Activity Feed zobrazuje doporučený obsah z dané stránky a nabízí uživateli možnost se přihlásit.

Obr. 6.5 Activity Feed
[21]

- *Recommendations Box*: Recommendations Box zobrazuje uživateli personalizovaná doporučení.

Vzhledem k tomu, že je obsah hostován Facebookem, může Recommendations Box zobrazovat personalizovaná doporučení v závislosti na tom, zda je nebo není uživatel přihlášený. K vytvoření doporučení je možné použít libovolný odkaz z dané stránky. Pokud je uživatel přihlášený, Recommendations Box zobrazuje doporučení a zároveň zvýrazní obsah, o který se zajímali jeho přátelé.



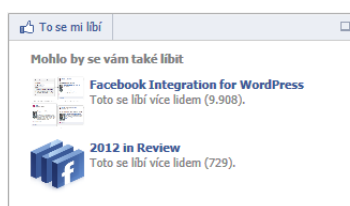
Obr. 6.6

Recommendation Box

[21]

- *Recommendations Bar*: Recommendations Bar umožňuje uživateli udělovat doporučení a sdílet obsah s jejich přáteli.

Když se webová stránka načte, Recommendations Bar je zavřený a umožňuje uživatelům sdílet obsah pomocí Like Button. Recommendations Bar je vždy ukotven v pravém dolním nebo v levém dolním rohu obrazovky.



Obr. 6.7

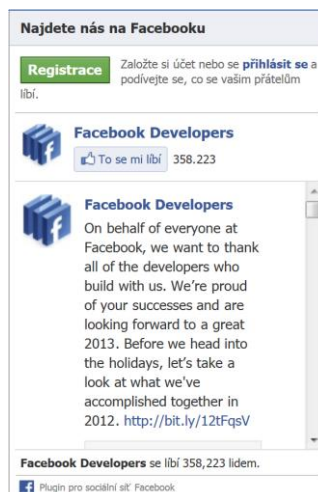
Recommendations Bar

[21]

- *Like Box*: Like Box umožňuje vlastníkům Facebook stránky získat další uživatele z vlastní webové stránky.

Like Box zobrazuje:

- Množství uživatelů a přátel uživatelů, kterým se tato stránka „líbí“.
- Nedávné aktuality na dané Facebook stránce.
- Tlačítko Like Button. (Po kliknutí není nutné navštěvovat danou Facebook stránku.)



Obr. 6.8 Like Box [21]

- *Login Button:* Login Button umožňuje uživateli registrovat se na dané stránce.

Zobrazuje profilové fotky přátel uživatele, kteří se na dané stránce už registrovali.



Bud'te první mezi svými přáteli, kdo použije developers.facebook.com.

Obr. 6.9 Login Button [21]

- *Registration:* Registration umožňuje uživatelům zaregistrovat se prostřednictvím dané stránky.

Pokud je uživatel přihlášený zobrazí se mu předvyplněný formulář s jeho údaji. Registration umožňuje přidávat další nepovinná pole např. oblíbený film nebo hudební žánr. Registration odstraňuje nutnost vytváření dvou oddělených přihlašovacích formulářů. Uživatelé, kteří nemají Facebook účet mohou Registration použít k vytvoření účtu a uživatel s již vytvořeným Facebook účtem k přihlášení.

 A screenshot of the Facebook registration form. At the top, it says 'Have a Facebook account? Log in to prefill the form below with your profile information.' Below this are several input fields: 'Jméno:' with a text box, 'Datum narození:' with dropdowns for 'Den:', 'Měsíc:', and 'Rok:', 'Pohlaví:' with a dropdown 'Vyberte jednu možnost:', 'Aktuální umístění:' with a text box, and 'E-mail:' with a text box. Below these fields is a blue 'Registrovat' button. At the bottom, there is a link 'Podmínky · Zásady ochrany osobních údajů'.

Obr. 6.10 Registration [21]

- *Facepile*: Zobrazuje profilové fotky uživatelů, kteří se již libovolným způsobem přihlásili na danou stránku, nebo může zobrazovat pouze ty uživatele, kteří se přihlásili prostřednictvím dané stránky.

[21]

6.1.2 Login

Login slouží k přihlašování do aplikací nebo webových stránek prostřednictvím Facebook účtu.

Login používá pro autentizaci a autorizaci uživatelů protokol OAuth 2.0¹⁵. Prostřednictvím tohoto protokolu je vytvořen přístupový token. Token slouží k identifikaci aplikace, stránky nebo uživatele. Token obsahuje informaci o době expirace tokenu, poskytnutých oprávněních a jaká aplikace token vytvořila.

Když se uživatel přihlásí do aplikace, aplikace získá přístup do jeho veřejného profilu a seznamu přátel. Pokud by jsme chtěli přistupovat k dalším informacím o uživateli, musí aplikace, při přihlášení uživatele, požádat o přidělení dalších oprávnění.

Oprávnění lze rozdělit do následujících skupin:

- *Public Profile and Friend List*: Oprávnění pro přístup k informacím v Facebook profilu uživatele např. jméno, věk, profilová fotka, jazyk, země původu, seznam přátel atd.
- *E-mail Permissions*: Oprávnění pro přístup k e-mailu uživatele. O toto oprávnění musí aplikace žádat odděleně od Public Profile and Friend List oprávnění.
- *Extended Profile Properties*: Oprávnění pro přístup k citlivým informacím o uživateli např. aktualizace uživatele a obsah vytvořený uživatelem.
- *Extended Permissions*: Oprávnění pro přístup k citlivým informacím o uživateli a možnost publikovat a mazat informace.
- *Open Graph Permissions*: Oprávnění pro přístup k Open Graph API.

¹⁵ OAuth je bezpečnostní protokol, který umožňuje uživatelům sdílet jejich obsah, aniž by museli poskytnout své uživatelské jméno a heslo.

- *Page permission*: Oprávnění zastupovat uživatele na všech Facebook stránkách, na kterých uživatel působí jako administrátor.

O oprávnění k publikování na Facebook profilu uživatele se žádá odděleně od oprávnění k čtení informací. Pokud uživatel udělí aplikaci oprávnění k publikování informací, aplikace může jménem uživatele:

- Psát na uživatelovu zeď.
- Ukládat fotografie, videa nebo vytvářet alba.
- Přidávat komentáře.
- Psát na zeď přátel.
- Vytvářet poznámky.

Aplikace by měly žádat pouze o ta oprávnění, která skutečně potřebují. Podle výzkumu společnosti Facebook, čím více oprávnění aplikace nebo webová stránka vyžaduje, tím více klesá pravděpodobnost, že se uživatel prostřednictvím aplikace nebo webové stránky přihlásí. Pokud tedy aplikace vyžaduje velké množství oprávnění, měla by uživateli poskytnout informaci, proč je dané oprávnění požadováno.

Pokud se uživatel rozhodne přihlásit k dané aplikaci nebo webové stránce prostřednictvím Facebook účtu, přihlašovací formulář mu zobrazí, jaká oprávnění daná aplikace nebo webová stránka vyžaduje. Jestliže aplikace vyžaduje další rozšiřující oprávnění, žádosti o jejich udělení se zobrazují v dalších krocích během procesu přihlašování. U oprávnění pro přístup k citlivým informacím je uživateli zobrazeno jaké informace budou sdílet a případně mohou tento krok přerušit. Uživatel může jednotlivá oprávnění, po jejich udělení, zakázat v nastavení Facebook účtu. Aplikace nebo webová stránka by tak měla být připravená na odebírání jednotlivých oprávnění a nenutit uživatele k jejich opětovnému přidělování. [22]

6.1.3 Dialogs

Dialogová okna umožňují přidávání prvků prostřednictvím JavaScriptu. Dialogs poskytují jednoduché, konzistentní rozhraní k poskytování funkcí uživateli. Dialogs nevyžadují žádná rozšiřující oprávnění a mohou být použita v libovolném kontextu v rámci aplikace nebo stránky.

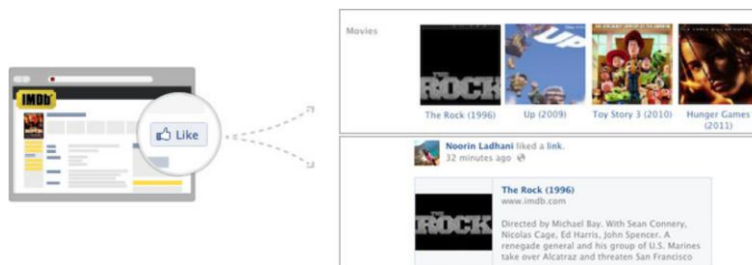
K dispozici je celkem sedm druhů dialogových oken:

- *The Feed Dialog*: Umožňuje uživatelům psát na svou zeď nebo na zeď jejich přátel.
- *The Login Dialog*: Umožňuje uživatelům přihlásit se k dané aplikaci nebo stránce prostřednictvím Facebook účtu.
- *The Friend Dialog*: Umožňuje uživatelům odesílat žádosti o přátelství.
- *The Pay Dialog*: Umožňuje uživatelům nákup Facebook Credits.
- *The Request Dialog*: Umožňuje uživatelům odeslat libovolný požadavek. Požadavek může být odeslán jednomu nebo více přátelům ze seznamu přátel.
- *The Send Dialog*: Umožňuje uživatelům odeslat zprávu. Zpráva může být odeslána jednomu nebo více přátelům ze seznamu přátel.

[23]

6.1.4 Open Graph Protocol

Pokud webová stránka zastupuje reálný objekt, jako je film, hudební album, restauraci nebo společnost, je možné prostřednictvím Open Graph protokolu specifikovat informace o tomto objektu. Pokud jsou součástí webové stránky tagy Open Graph protokolu, stává se webová stránka stránkou ekvivalentní k stránce vytvořené na Facebooku. To znamená, že pokud uživatel klikne na tlačítko Like Button, umístěné na webové stránce, vznikne spojení mezi webovou stránkou a profilem uživatele. Následně se webová stránka zobrazí v profilu uživatele a na všech ostatních místech jako u normální Facebook stránky např. ve vyhledávání. [24]



Obr. 6.11 Open Graph [24]

6.2 Google+

Sociální síť Google+ nabízí k integraci svých služeb softwarové prostředí Google+ Platform. Základní součásti Google+ Platform pro webové stránky je:

- Google+ Plugins

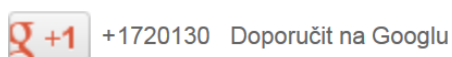
[25]

6.2.1 Google + Plugins

Google+ Plugins umožňují přidávání prvků prostřednictvím JavaScriptu. Výhodou tohoto řešení je snadná integrace kódu do webové stránky. Kód každého prvku je vygenerován prostřednictvím konfiguratoru.

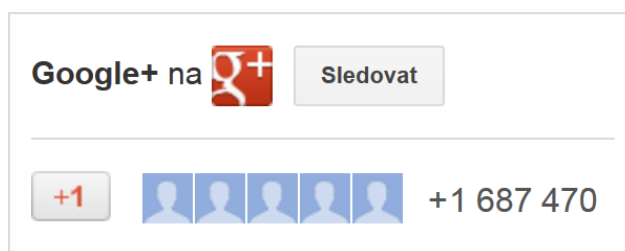
Součástí Google+ Plugins je:

- *+1 Button*: +1 Button umožňuje uživatelům doporučit danou stránku na Google+.



Obr. 6.12 +1 Button [25]

- *Google+ Badge*: Google+ Badge umožňuje uživatelům sledovat veřejné aktualizace Google+ stránky.



Obr. 6.13 Google+ Badge [25]

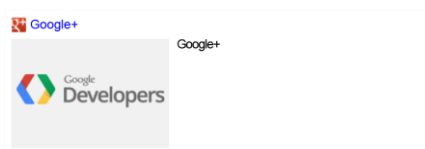
- *Share*: Share umožňuje uživateli sdílet obsah s přáteli na Google+.



Obr. 6.14 Share [25]

- *Snippet*: Snippet umožňuje uživatelům vytvářet krátké popisky stránek s obrázkem.

[25]



Obr. 6.15 Snippet [25]

6.3 Twitter

Sociální síť Twitter nabízí k integraci svých služeb softwarové prostředí Twitter Platform. Základní součástí Twitter Platform pro webové stránky je:

- Twitter Plugins

[26]

6.3.1 Twitter Plugins

Twitter Plugins umožňují přidávání prvků prostřednictvím JavaScriptu. Výhodou tohoto řešení je snadná integrace kódu do webové stránky. Kód každého prvku je vygenerován prostřednictvím konfiguratoru.

Součástí Twitter Plugins je:

- *Tweet Button*: Tweet Button umožňuje uživatelům sdílet odkaz na webovou stránku.

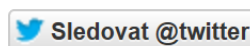


Obr. 6.16

Tweet

Button [26]

- *Follow Button*: Follow Button umožňuje uživateli stát se následovníkem dané Twitter stránky.



Obr. 6.17 Follow

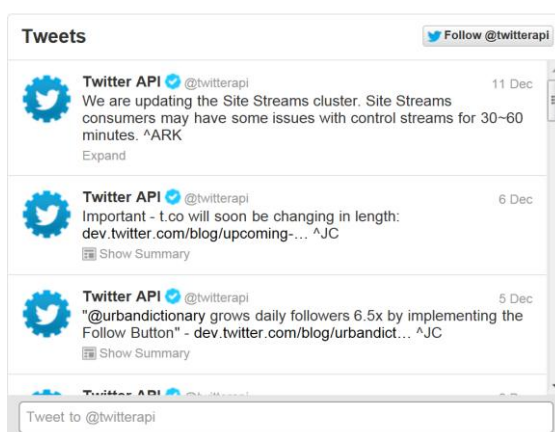
Button [26]

- *Embedded Tweets*: Embedded Tweets umožňuje uživateli vkládat vybrané tweety z Twitter účtu na svou webovou stránku.



Obr. 6.18 Embedded Tweets [26]

- *Embedded Timelines*: Embedded Timelines umožňují uživateli vkládat celý obsah z Twitter účtu. Prostřednictvím Embeddes Timelines je možné reagovat na tweety, sdílet tweety nebo je označovat jako oblíbené.



Obr. 6.19 Embedded Timelines [26]

- *Twitter Cards*: Twitter Cards umožňují uživateli přidávat multimediální obsah k tweetům, které odkazují na obsah na dané stránce.

[26]



Obr. 6.20 Twitter Cards [26]

7 ANALÝZA A PREDIKCE CHOVÁNÍ UŽIVATELŮ

7.1 Web Mining

Web Mining je skupina technik používaných k automatickému procházení webových stránek a získávání informací. [28]

Podle [27] Web Mining neshromažďuje pouze běžná statistická data jako je například počet návštěvníků webové stránky za den. Web Mining může pomoci zodpovědět otázky typu:

- Odkud přicházejí návštěvníci.
- Jak se návštěvníci na webové stránce chovají.
- Jaké jsou typické sekvence průchodu stránkami.
- Při jaké sekvenci průchodu došlo k nákupu zboží nebo rezervaci.
- Jak dlouho se na stránkách návštěvníci zdrží.
- Jak a odkud návštěvníci opouští stránky.
- Jaký internetový prohlížeč návštěvníci používají.
- Jaký operační systém návštěvníci používají.

Na těchto datech může být aplikováno rozsáhlé množství Data Miningových technik, které vedou k úpravě nebo vytvoření webových stránek tak, aby jejich návštěvníci rychle dostali požadované informace, a pokud se jedná např. o internetový obchod, tak aby si návštěvník objednal zboží.

Web Mining, lze rozdělit do několika skupin:

- *Web Content Mining*: Získávání užitečných informací z webových stránek a dokumentů.
- *Web Structure Mining*: Získávání užitečných informací ze struktury webových stránek.
- *Web Usage Mining*: Získávání užitečných informací z chování uživatelů webových stránek.

7.1.1 Web Usage Mining

Zatímco Web Content Mining a Web Structure Mining se snaží získávat informace pouze z dat dostupných na webových stránkách, Web Usage Mining se snaží získávat informace z interakce uživatelů s webovými stránkami. Tyto informace jsou získávány z:

- Cookies¹⁶.
- Logů zaznamenávající pohyb na serveru.
- Logů webových prohlížečů.
- Počtu kliknutí myši.
- Uživatelských profilů.
- Uživatelských transakcí.
- Vyhledávaných informací.

[28]

7.1.2 Sběr dat z úrovně serveru

Serverové logy jsou jedny z nejdůležitějších zdrojů informací o pohybu uživatelů na webových stránkách. Informace uložené v serverovém logu zaznamenávají přístup a pohyb uživatelů na webových stránkách. Tyto serverové logy mohou být uloženy v několika formátech, které zjednodušují získávání informací z logů:

- Common Log Format (CLF).
- Extended Common Log Format (ECLF).

Nicméně, informace získané ze serverových logů nemusí být úplně přesné. Serverové logy totiž neobsahují požadavky, které byly vytvořeny prostřednictvím metody POST. Problém může způsobovat i cachování¹⁷ webových stránek. Cachování webových stránek způsobuje, že se webová stránka nenačítá z webového serveru, ale z cache webového prohlížeče.

¹⁶ Cookies označují malé množství dat sloužící např. k identifikaci uživatele, sledování pohybu uživatele nebo k ukládání nastavení uživatele. [9]

¹⁷ Cache je paměť pro uchovávání kopií navštěvovaných stránek. [9]

Další možností k získávání informací prostřednictvím serverových logů je Packet Sniffing. Packet Sniffing monitoruje síťový provoz způsobený připojováním se na webový server a získává informace přímo z TCP/IP¹⁸ paketů¹⁹.

Další možností k získávání informací je zaznamenávání informací z Cookies. Cookies mohou být automaticky generovány webovým serverem za účelem sledování pohybu uživatele. Sledování pohybu prostřednictvím Cookies, ale není jednoduché kvůli HTTP protokolu, protože neumí uchovávat stav komunikace. Cookies se spoléhají na interakci uživatelů, čímž způsobily růst obav ohledně soukromí uživatelů.

Webové servery také mohou zaznamenávat vyhledávací dotazy uživatelů, které jsou typicky vytvářeny, když uživatel prohledává webové stránky. [29]

7.1.3 Sběr dat z úrovně uživatele

Implementace metod pro sběr dat z úrovně uživatele vyžaduje interakci uživatele a webového serveru. Sběr dat z úrovně uživatele má oproti sběru dat z úrovně serveru několik výhod např. řeší problém cachování webových stránek a identifikace relací.

Sběr dat z úrovně uživatele může být implementován prostřednictvím vzdálených agentů, jako jsou Java Applety nebo JavaScripty. Java Applety, oproti server logům, nepřinášejí žádné výrazné vylepšení z hlediska zjišťování počtu zobrazení atd. Ve skutečnosti mohou spíše způsobit další režie v momentu, kdy je Java Applet načítán poprvé. JavaScripty zase způsobují malá časová prodlení při zobrazování webové stránky navíc, nedokážou rozpoznat, když uživatel klikne ve webovém prohlížeči na tlačítko obnovit nebo zpět/vpřed.

Další možností je modifikace zdrojového kódu již existujícího webového prohlížeče a vylepšit tak jeho funkce pro sběr dat. Využití modifikovaného webového prohlížeče je mnohem všestrannější a umožňují nám sběr dat z více webových stránek, které uživatel navštívil. Největší problém ale spočívá v přesvědčení uživatelů, aby denně používali modifikovaný webový prohlížeč. Tento problém je možné vyřešit prostřednictvím

¹⁸ TCP/IP je komunikační protokol používaný pro komunikaci v počítačových sítích a Internetu. [9]

¹⁹ Paket je blokem dat obsahující data a informace o tom, odkud a kam mají být data poslána. [9]

společností, které uživatele, za používání modifikovaných prohlížečů a klikání na webové odkazy nějakým způsobem odměňují. [29]

II. PRAKTICKÁ ČÁST

8 ANALÝZA

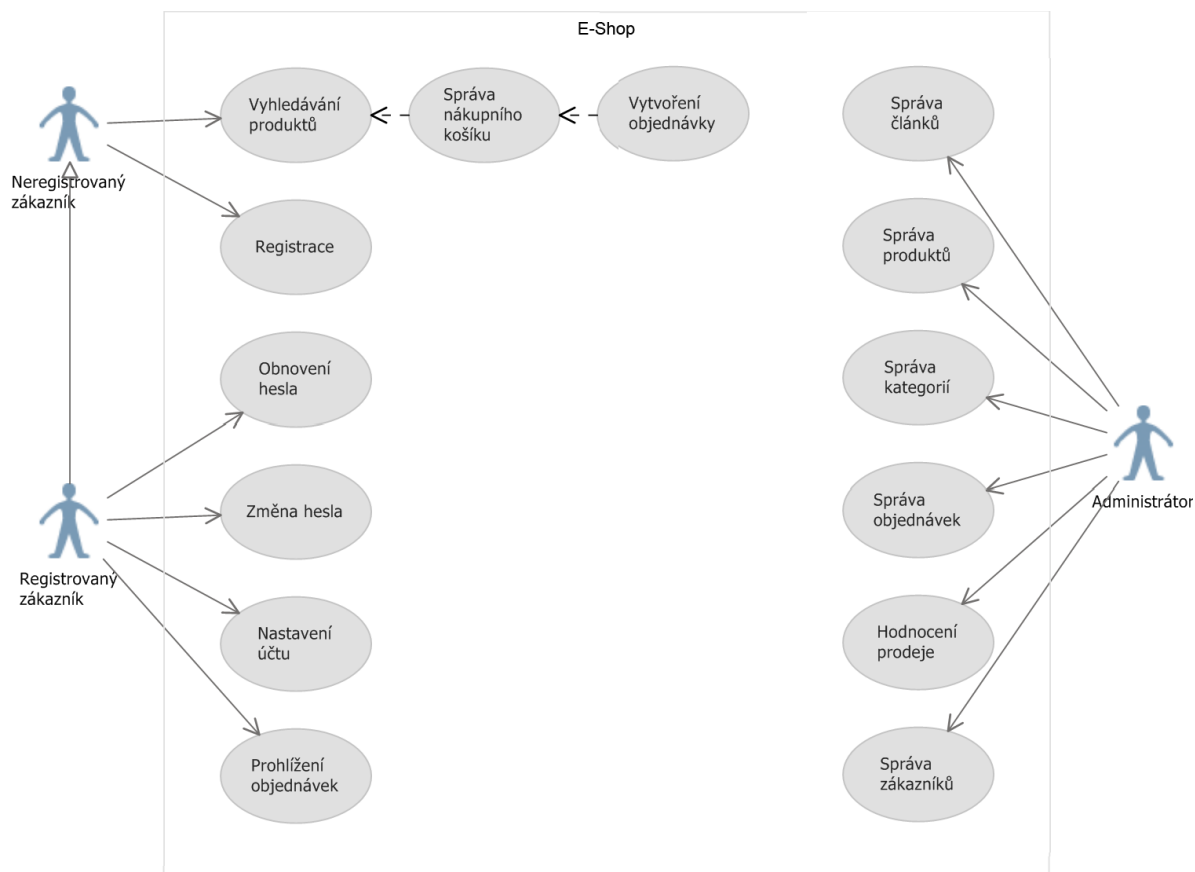
Návrhu řešení internetového obchodu předchází analýza požadavků firmy na internetový obchod. Jednotlivé požadavky jsou popsány v následujících bodech:

- Vytvořit přehledné uživatelské prostředí.
- Umožnit zákazníkům získat dostatek informací o produktech.
- Nabídnout zákazníkům informace o produktech v kontextu jejich užívání v souladu s nařízením komise (EU) č. 432/2012.
- Nabídnout zákazníkům možnost komunikace s provozovateli internetového obchodu.
- Umožnit zákazníkovi informovat své blízké o produktech.
- Umožnit zákazníkům vyhledávat produkty.
- Umožnit zákazníkům nakupovat produkty.
- Vytvořit přehledné administrátorské rozhraní.
- Umožnit administrátorovi správu článků.
- Umožnit administrátorovi správu produktů.
- Umožnit administrátorovi správu kategorií.
- Umožnit administrátorovi správu objednávek.
- Umožnit administrátorovi hodnotit prodej.
- Umožnit administrátorovi správu zákazníků.

Z obchodního hlediska by měl internetový obchod reprezentovat firmu a zvýšit povědomí o firmě.

8.1 Diagram užití

Diagram užití slouží k popisu možností uživatelů, kteří internetový obchod používají, popisuje jednotlivé služby poskytované uživatelům a vazby mezi uživateli. Internetový obchod pracuje se třemi skupinami uživatelů, kterým je přiřazena určitá role.



Obr. 8.1 Diagram užití

Uživatelské role:

- *Neregistrovaný zákazník*: Neregistrovaný zákazník představuje množinu zákazníků, kteří nejsou zaregistrováni na internetovém obchodu.

Služby poskytované internetovým obchodem:

- *Vyhledávání produktů*: Procházení katalogu produktů a veřejného obsahu internetového obchodu.
- *Správa nákupního košíku*: Přidávání a odebírání položek z nákupního košíku.
- *Vytvoření objednávky*: Vyplnění kontaktních, dodacích údajů a odeslání objednávky.
- *Registrace*: Registrace do internetového obchodu.
- *Registrovaný zákazník*: Registrovaný zákazník představuje množinu zákazníků, kteří jsou zaregistrováni na internetovém obchodu.

Služby poskytované internetovým obchodem:

- *Registrovaný zákazník má přístup ke všem službám jako neregistrovaný zákazník.*
- *Obnovení hesla:* Při ztrátě hesla je možné požádat o zaslání nového hesla e-mailem.
- *Změna hesla.*
- *Nastavení účtu:* Vyplnění kontaktních a dodacích údajů.
- *Prohlížení objednávek:* Přístup k historii objednávek.
- *Administrátor:* Administrátor představuje uživatele, který může spravovat obsah, objednávky a uživatele.

Služby poskytované internetovým obchodem:

- *Správa článků:* Přidávání článků.
- *Správa produktů:* Přidávání nových produktů.
- *Správa kategorií:* Přidávání nových kategorií zboží.
- *Správa objednávek:* Zobrazení a změna stavu objednávek.
- *Hodnocení prodeje:* Zobrazení množství přichozích objednávek a prodaných produktů.
- *Správa zákazníku:* Přístup ke kontaktním a dodacím údajům registrovaných uživatelů.

9 POPIS POUŽITÝCH NÁSTROJŮ

9.1 Visual Web Developer 2010 Express

Visual Web Developer 2010 Express (dále jen VWD) je vývojovým prostředím pro tvorbu webových aplikací, navrženým společností Microsoft. Vývojové prostředí VWD obsahuje několik šablon pro tvorbu webových stránek.

- *ASP.NET Web Site*: Šablona obsahuje jeden předem vytvořený adresář App_Data, jednu ASPX stránku Default.aspx a konfigurační soubor web.config.
- *ASP.NET Web Site (Razor)*: Šablona je stejná jako předchozí šablona, ale využívá Razor syntaxi.
- *ASP.NET Empty Web Site*: Šablona obsahuje předem vytvořený konfigurační soubor web.config.
- *ASP.NET Dynamic Data Entities Website*: Šablona obsahuje rozsáhlejší strukturu adresářů a souborů pro práci s ADO.NET Entity Frameworkem (dále jen EF). EF využívá objektově orientovaného modelu databáze. Pomocí EF můžeme převádět databázové prvky jako jsou tabulky, sloupce nebo řádky do objektů, ke kterým můžeme přistupovat a následně s nimi libovolně manipulovat.
- *ASP.NET Dynamic Data Entities LINQ to SQL Website*: Šablona obsahuje rozsáhlejší strukturu adresářů a souborů pro práci s LINQ. LINQ to SQL využívá objektově orientovaného modelu databáze. Dotazy napsané pomocí LINQ jsou překládány do SQL dotazů.
- *WCF Service*: Šablona slouží k vytváření WCF služeb. WCF služby jsou podobné standardním webovým službám. Webové služby slouží k volání metod prostřednictvím lokální sítě nebo Internetu.

[1] [29]

9.2 Adobe Photoshop CS2

Adobe Photoshop CS2 je grafickým editorem pro tvorbu a úpravu grafiky, navrženým společností Adobe.

10 NÁVRH INTERNETOVÉHO OBCHODU

10.1 Rozmístění prvků

Základem internetového obchodu je přehledné uživatelské prostředí. Uživatelské prostředí je rozděleno do následujících bloků:

- *Hlavička stránky*: Hlavička stránky obsahuje:
 - Logo firmy.
 - Název firmy.
- *Horizontální menu*: Slouží k základní navigaci po internetovém obchodu.
- *Kategorie produktů*: Slouží k základní navigaci mezi kategoriemi produktů.
- *Vyhledávání*: Slouží k prohledávání veškerého obsahu internetového obchodu.
- *Tělo stránky*: Slouží k zobrazování obsahu jednotlivých stránek a výsledků vyhledávání.
- *Patička stránky*.

Logo firmy	Hlavička stránky
Název firmy	
Horizontální menu	
Kategorie	Tělo stránky
Vyhledávání	
Patička stránky	

Obr. 10.1 Návrh blokového rozmístění prvků

10.2 Grafický návrh

Při tvorbě grafického návrhu internetového obchodu jsem se snažil docílit funkčního a jednoduchého návrhu. Při výběru barevné palety jsem vycházel z barev použitých u loga firmy, abych docílil konzistentního vzhledu. K návrhu layoutu²⁰ jsem použil program Adobe Photoshop CS2 využívající šablonu 960 Grid System²¹.



Obr. 10.2 Výsledný vzhled internetového obchodu

²⁰ Layout je grafické rozvržení webové stránky.

²¹ Soubor šablon o standardní šířce 960px. Dostupné z: <http://960.gs>

11 REALIZACE INTERNETOVÉHO OBCHODU

11.1 Grafický návrh

Pro realizaci internetového obchodu jsem si ve VWD vybral šablonu ASP.NET Empty Web Site. Šablona obsahuje pouze konfigurační soubor web.config a lze ji libovolně rozšiřovat. Abych mohl aplikovat grafický návrh, je nutné všechny grafické prvky internetového obchodu rozdělit na menší části a uložit je do oddělených souborů. K realizaci grafického návrhu jsem vytvořil MasterPage a soubor kaskádových stylů CSS.

MasterPage je vzorovou stránkou, jejíž vzhled budou dědit všechny ostatní stránky, celý vzhled je tak definovaný v jednom souboru a všechny změny jsou automaticky aplikovány na všechny stránky, které jsou založené na této MasterPage. MasterPage je uvozena hlavičkou:

```
<%@ Master Language="C#" AutoEventWireup="true" CodeFile="Site.master.cs" Inherits="Site" %>
```

Za hlavičkou následuje standardní struktura HTML stránky. Abych docílil dědičnosti vzhledu, musí být do MasterPage vložen ContentPlaceHolder:

```
<asp:ContentPlaceHolder ID="CPMainContent" runat="server">
</asp:ContentPlaceHolder>
```

ContentPlaceHolder se chová jako vložený rámec. Každá další vložená stránka automaticky obsahuje následující strukturu:

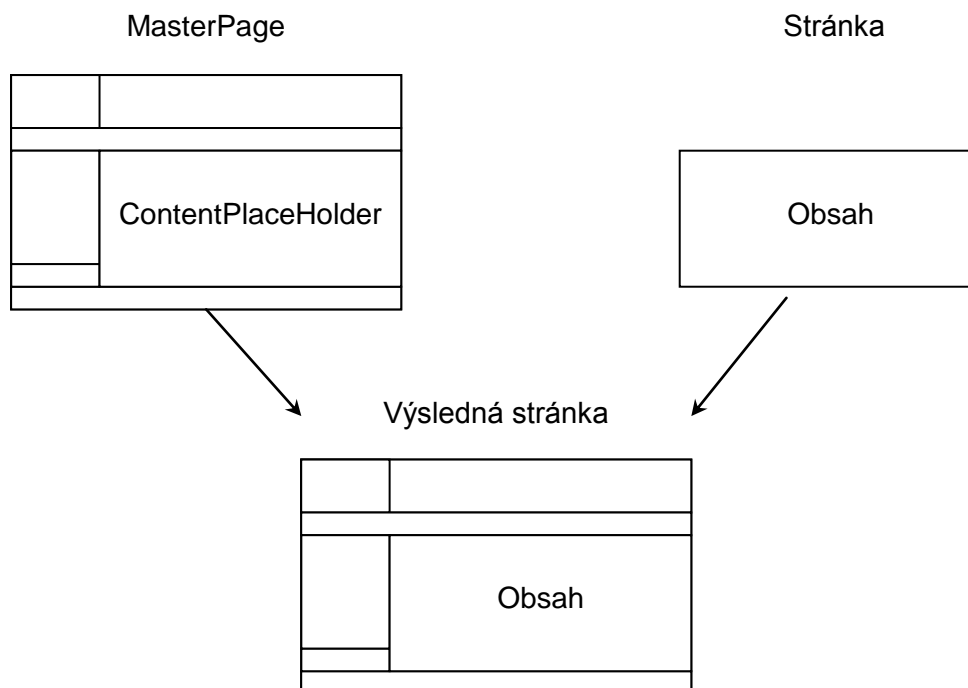
```
<%@ Page Title="PrevenceZdravi.eu - Doplnky stravy a Kosmetika" Language="C#"
MasterPageFile="~/Site.master" AutoEventWireup="true"
CodeFile="Default.aspx.cs" Inherits="_Default" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="Server">
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="CPMainContent" runat="Server">
</asp:Content>
```

Kód vložený mezi značky Content se automaticky zobrazí na místě, kde byl vložen ContentPlaceHolder.

Soubor kaskádových stylů CSS obsahuje styly všech prvků vyskytujících se v internetovém obchodu. Jednotlivé styly lze aplikovat pomocí parametrů ID nebo CssClass.



Obr. 11.1 Sestavení výsledné stránky

11.2 Struktura databáze

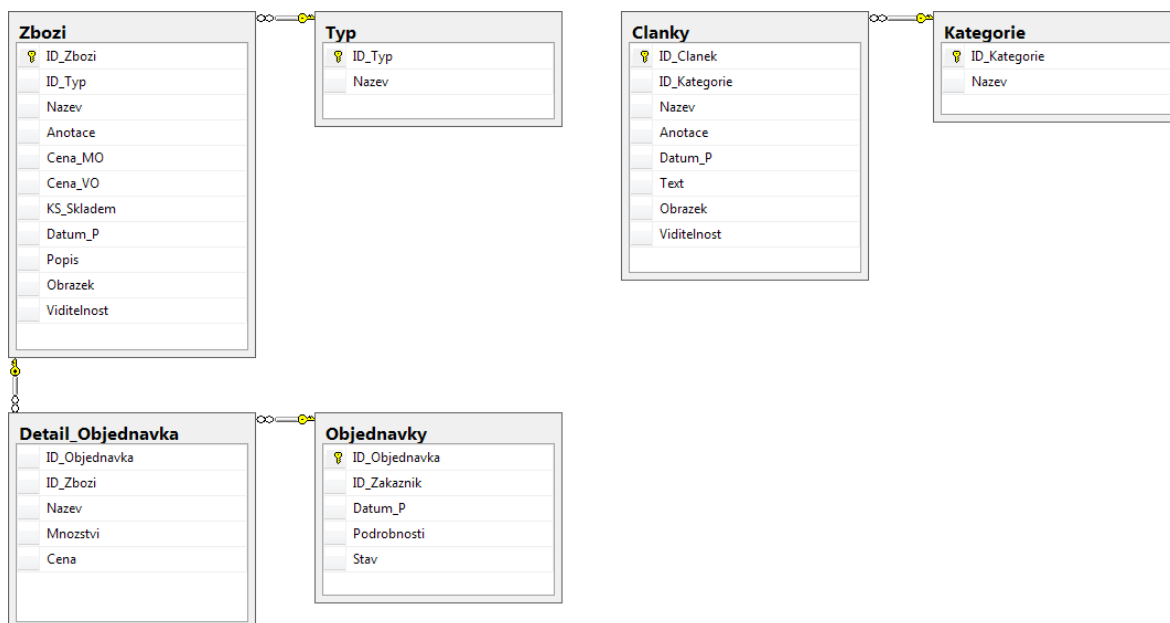
Databázi lze definovat jako kolekci dat, která jsou uspořádána tak, že s nimi můžeme manipulovat. Pro přístup do databáze se využívá jazyk SQL, prostřednictvím kterého můžeme data získávat nebo s nimi manipulovat. Pro realizaci databáze jsem použil databázi SQL Express. Po přidání databáze se v adresáři App_Data vytvoří databáze DATABASE.MDF.

Obr. 11.2 zobrazuje strukturu databáze. Databáze obsahuje celkem šest tabulek, které se vztahují k obsahu internetového obchodu:

- *Zbozi*: Obsahuje záznamy o všech produktech.
- *Typ*: Obsahuje záznamy o všech kategoriích produktů.
- *Objednavka*: Obsahuje záznamy o všech vytvořených objednávkách. Do položky ID_Zakanik se ukládá unikátní identifikátor uživatele z uživatelské databáze.
- *Detail_Objednavka*: Obsahuje záznamy o obsahu jednotlivých objednávek.
- *Clanky*: Obsahuje záznamy o článcích.

- *Kategorie*: Obsahuje záznamy o všech kategoriích článků.

Aby nedocházelo k porušení integrity dat uložených v databázi, jsou mezi tabulkami vytvořeny relace. Kromě udržování integrity dat, slouží relace k snadnějšímu pochopení struktury databáze.



Obr. 11.2 Databázový diagram

Pokud budu chtít použít tuto databázi jako zdroj dat pro nějaký dynamický prvek, musím nejprve vytvořit zdroj dat. Konfigurace zdroje dat se provádí prostřednictvím průvodce, kde lze zvolit, jakou databázi chci použít a jaký databázový dotaz chci aplikovat. Po dokončení konfigurace zdroje dat se do stránky vloží `SqlDataSource`:

```
<asp:SqlDataSource ID="Katalog" runat="server" ConnectionString="<%"$
ConnectionStrings:ConnectionToDatabase %>" SelectCommand="SELECT * FROM [Typ]">
</asp:SqlDataSource>
```

V konfiguračním souboru `web.config` se vytvoří `ConnectionString`, který obsahuje informace o umístění a nastavení databáze:

```
<connectionStrings>
  <add name="ConnectionToDatabase" connectionString="Data
Source=.\SQLEXPRESS;AttachDbFilename=|DataDirectory|\DATABASE.MDF;Integrate
d Security=True;User Instance=True" providerName="System.Data.SqlClient" />
</connectionStrings>
```

Databáze, vztahující se k uživatelským účtům a profilům bude popsána v následující kapitole.

11.3 Uživatelské rozhraní

Pro realizaci odděleného uživatelského rozhraní jak pro neregistrované/registrované uživatele a administrátora je nutné vytvořit všechny dostupné formuláře:

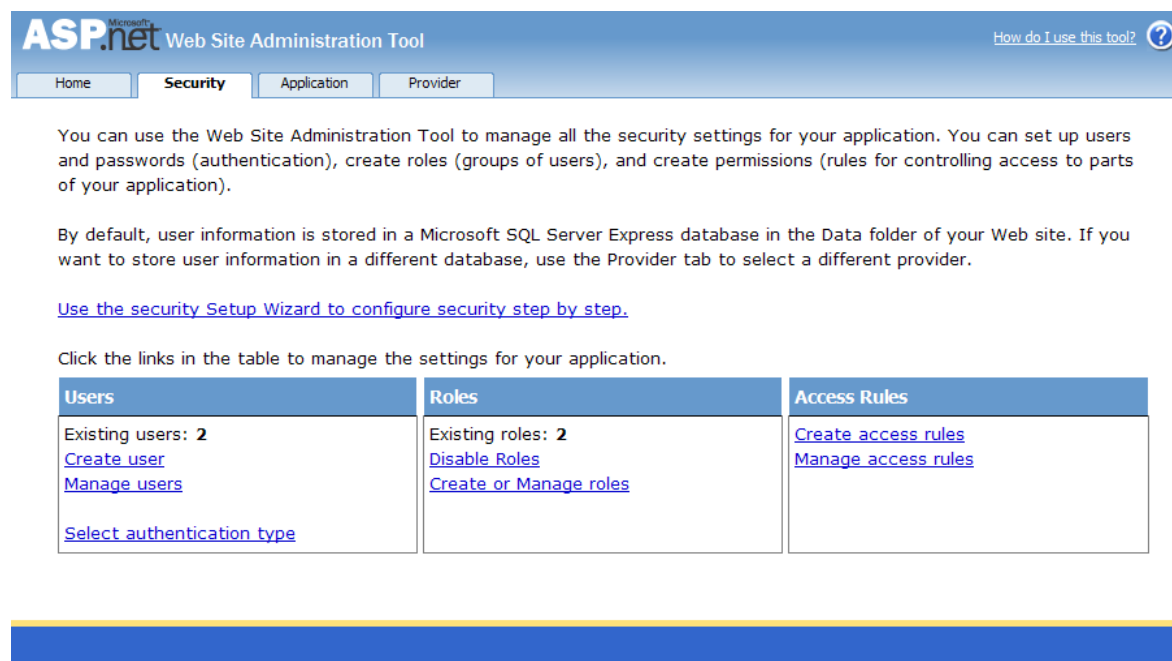
- Přihlašovací formulář.
- Registrační formulář.
- Formulář pro obnovení hesla.
- Formulář pro změnu hesla.

Všechny výše zmíněné formuláře obsahují veškerou logiku pro ověřování a správu uživatelů. Tyto formuláře nekomunikují přímo s příslušným zdrojem dat, ale s nakonfigurovaným providerem. Pokud některý z těchto formulářů použijeme poprvé, provider provede kontrolu, zda je dostupná databáze. V případě, že databáze neexistuje, vytvoří se v adresáři App_Data databáze ASPNETDB.MDF a v konfiguračním souboru web.config se vytvoří ConnectionString, který obsahuje informace o umístění a nastavení databáze:

```
<connectionStrings>
  <add name="ConnectionToUserDatabase" connectionString="Data
Source=.\SQLEXPRESS;AttachDbFilename=|DataDirectory|\ASPNETDB.MDF;
Integrated Security=True; User Instance=True"
  providerName="System.Data.SqlClient" />
</connectionStrings>
```

Veškerou správu uživatelů je možné provádět prostřednictvím ASP.NET Web Site Administration Tool (dále jen WSAT). WSAT je standardní součástí VWD, používáním WSAT je eliminována potřeba ruční editace konfiguračního souboru web.config. Není-li soubor web.config vytvořený, při prvním spuštění WSAT se automaticky vytvoří. WSAT lze použít k:

- Správě uživatelů.
- Správě rolí.
- Správě pravidel přístupu.
- Konfiguraci aplikace.
- Konfiguraci SMTP.



Obr. 11.3 Okno Web Site Administration Tool

K oddělení uživatelského rozhraní slouží LoginView: LoginView je ovládacím prvkem, který umožňuje zobrazit různé informace různým uživatelům. Umožňuje tak rozlišovat mezi neregistrovanými/registrovanými uživateli a administrátorem.

```
<asp:LoginView ID="LoginView1" runat="server">
  <AnonymousTemplate>
</AnonymousTemplate>
  <LoggedInTemplate>
</LoggedInTemplate>
  <RoleGroups>
    <asp:RoleGroup Roles="Uzivatel">
      <ContentTemplate>
</ContentTemplate>
    </asp:RoleGroup>
    <asp:RoleGroup Roles="Admin">
      <ContentTemplate>
</ContentTemplate>
    </asp:RoleGroup>
  </RoleGroups>
</asp:LoginView>
```

LoginView obsahuje několik skupin značek:

- *AnonymousTemplate*: Mezi značky AnonymousTemplate se vkládá obsah, který se zobrazuje neregistrovaným uživatelům.
- *LoggedInTemplate*: Mezi značky LoggedInTemplate se vkládá obsah, který se zobrazuje všem registrovaným uživatelům

- *RoleGroups*: Mezi značky ContentTemplate se vkládá obsah, který se zobrazuje skupinám uživatelů uvedeným v parametru Roles.

Každý uživatel má svůj osobní profil, který obsahuje dodací a fakturační údaje. Tento profil se používá i u neregistrovaných uživatelů v případě, že provedou objednávku. Veškeré údaje uložené v profilu jsou ukládány do uživatelské databáze.

```
<system.web>
  <profile>
    <properties>
      <add name="Jmeno" type="System.String" allowAnonymous="true" />
      <add name="Prijmeni" type="System.String" allowAnonymous="true" />
      <add name="EMail" type="System.String" allowAnonymous="true" />
      <add name="Ulice" type="System.String" allowAnonymous="true" />
      <add name="Mesto" type="System.String" allowAnonymous="true" />
      <add name="PSC" type="System.String" allowAnonymous="true" />
      <add name="Telefon" type="System.String" allowAnonymous="true" />
    </properties>
  </profile>
</system.web>
```

Veškeré informace o odeslání objednávky nebo vytvoření účtu jsou zasílány uživatelům na uvedenou e-mailovou adresu. K automatizaci procesu odesílání e-mailů je nutné v konfiguračním souboru nastavit SMTP:

```
<system.net>
  <mailSettings>
    <smtp deliveryMethod="Network" from="PrevenceZdravi.eu
    &lt;info@prevencezdravi.eu>">
      <network host="****" userName="****" password="****" port="25" />
    </smtp>
  </mailSettings>
</system.net>
```

K práci s e-maily slouží knihovna tříd System.Net.Mail, která obsahuje všechny třídy a metody potřebné k vytvoření a odeslání e-mailů. Pro každý e-mail je možné vytvořit šablonu. Šablonu je nutné uložit do složky App_Data a při vytváření e-mailu je nutné označit jakou šablonu chceme použít. Například, při registraci uživatele je odeslán e-mail obsahující informace o uživatelském jménu a heslu. Tento e-mail používá následující šablonu:

Dobrý den,

vítejte na www.PrevenceZdravi.eu.

Vaše přihlašovací údaje:

Uživatelské jméno: <% UserName %>

Heslo: <% Password %>

Tým PrevenceZdravi.eu

(Na tento email neodpovídejte, byl automaticky vygenerován.)

Při odeslání e-mailu jsou elementy <% UserName %> a <% Password %> nahrazeny přihlašovacími údaji uživatele.

11.4 Zabezpečení aplikace

Bezpečnost webová aplikace je velice důležitým aspektem, zejména u aplikací shromažďujících informace o registrovaných uživateli. Webová aplikace by měla obsahovat ochranu proti následujícím bezpečnostním hrozbám:

- *Injection:* Ochrana spočívá v kontrole hodnot vkládaných do parametrů předávaných v URL. V mém případě např. http://www.prevencezdravi.eu/Zbozi.aspx?ID_Zbozi=100. Kontrolu hodnoty lze provést několika způsoby:

- *Definice datového typu parametru:*

```
<asp:QueryStringParameter                                Name="ID_Zbozi"
QueryStringField="ID_Zbozi" Type="Int32" />
```

- *Kontrola hodnoty parametru pomocí regulérního výrazu:*

```
String idZbozi = Request.QueryString["ID_Zbozi"];
Regex kladneCislo = new Regex(@"^0*[1-9][0-9]*$");

if (kladneCislo.IsMatch(idZbozi))
{
}
else
{
}
```

- *Cross-Site Scripting (XSS):* Ochrana spočívá v kontrole formulářových polí a URL. Automatickou kontrolu lze zapnout několika způsoby:

- Nastavením konfiguračního souboru *web.config*:

```
<system.web>
  <httpRuntime requestValidationMode="4.0"
</system.web>
```

nebo

```
<system.web>
  <pages validateRequest="true"
</system.web>
```

- Vložením parametru *ValidateRequest* do hlavičky stránky:

```
<%@ Page Title="PrevenceZdravi.eu - Doplnky stravy a Kosmetika"
Language="C#" MasterPageFile="~/Site.master"
AutoEventWireup="true" CodeFile="Default.aspx.cs"
Inherits="_Default" ValidateRequest="true" %>
```

- *Broken Authentication and Session Management*: Ochrana spočívá v používání aplikačních služeb ASP.NET a konfiguraci SqlMembership providera prostřednictvím konfiguračního souboru *web.config*:

```
<system.web>
  <membership>
    <providers>
      <clear />
      <add
        name="AspNetSqlMembershipProvider"
        type="System.Web.Security.SqlMembershipProvider,
        System.Web, Version=4.0.0.0, Culture=neutral,
        PublicKeyToken=b03f5f7f11d50a3a"
        connectionStringName="LocalSqlServer"
        enablePasswordRetrieval="false" enablePasswordReset="true"
        requiresQuestionAndAnswer="true" applicationName="/"
        requiresUniqueEmail="true" passwordFormat="Hashed"
        maxInvalidPasswordAttempts="5" minRequiredPasswordLength="6"
        minRequiredNonalphanumericCharacters="0"
        passwordAttemptWindow="10"
        passwordStrengthRegularExpression="" />
    </providers>
  </membership>
</system.web>
```

- *Insecure Direct Object References*: Ochrana spočívá v kontrole hodnot vkládaných do parametrů předávaných v URL. V mém případě např. http://www.prevencezdravi.eu/Uzivatel/Detail-objednavky.aspx?ID_Objednavka=1000. Primárním klíčem objednávky je čtyřmístné číslo, které se automaticky inkrementuje, aby uživatel neměl přístup k objednávkám ostatních uživatelů, musí být před přístupem k objednávce

provedena autentizace uživatele. Autentizace se provádí vždy při přístupu k detailům o objednávce:

```
protected void SqlDataSource1_Selecting(object sender,
SqlDataSourceSelectingEventArgs e)
{
    MembershipUser uzivatel = Membership.GetUser();
    Guid idZakaznik = (Guid)uzivatel.ProviderUserKey;
    e.Command.Parameters["@ID_Zakaznik"].Value =
    idZakaznik.ToString().ToLower();
}
```

Tato metoda se automaticky zavolá, když uživatel přistupuje k detailům o objednávce. Získá se GUID uživatele a porovná se s GUID uvedeným u účísla objednávky v databázi.

- *Cross-Site Request Forgery (CSRF)*: Ochrana spočívá v zákazu vkládání obsahu, který by se zobrazoval ostatním uživatelům.
- *Security Misconfiguration*: Předpokladem pro úspěšné napadení webové aplikace je získat dostatek informací o použité platformě a struktuře webové aplikace. Z tohoto hlediska jsou velmi nebezpečné chybové stránky. Ochrana spočívá v nastavení vlastních chybových stránek v konfiguračním souboru web.config, které tyto informace neobsahují:

```
<system.web>
  <customErrors defaultRedirect="~/errors/Chyba.aspx" mode="On" >
    <error statusCode="404" redirect="~/Errors/Stranka-Nebyla-
      Nalezena.aspx" />
  </customErrors>
</system.web>
```

- *Insecure Cryptographic Storage*: Ochrana spočívá v používání aplikačních služeb ASP.NET a konfiguraci SqlMembership providera prostřednictvím konfiguračního souboru web.config. Výchozím hashovacím algoritmem je SHA1. Automatické hashování hesla se nastavuje pomocí parametru:

```
passwordFormat="Hashed"
```

- *Failure to Restrict URL Access*: Ochrana spočívá v nastavení přístupových práv každé skupině uživatelů v konfiguračním souboru web.config:

```
<location path="Admin">
  <system.web>
    <authorization>
      <allow roles="Admin" />
      <deny users="*" />
    </authorization>
  </system.web>
</location>

<location path="Uzivatel">
  <system.web>
    <authorization>
      <allow roles="Uzivatel" />
      <deny users="*" />
    </authorization>
  </system.web>
</location>
```

Každá skupina uživatelů definovaná v parametru roles, má přístup k adresářům definovaným v parametru path.

- *Insufficient Transport Layer Protection*: Ochrana spočívá v používání SSL/ TLS na stránkách v konfiguračním souboru web.config, kde dochází k odesílání citlivých údajů z webového prohlížeče na webový server. Abych mohl aplikovat tuto ochranu, musel bych nejdříve požádat o vytvoření certifikátu jednu z kvalifikovaných certifikačních autorit.

```
<system.web>
  <authentication mode="Forms">
    <forms loginUrl="~/Prihlaseni.aspx" timeout="30"
      requireSSL="true" />
  </authentication>
</system.web>
```

- *Unvalidated Redirects and Forwards*: Ochrana spočívá v provádění přesměrování bez nutnosti vkládání koncové webové stránky do parametru v URL nebo v kontrole zadané koncové webové stránky do parametru se seznamem platných webových stránek.

V internetovém obchodu nedochází k přesměrovávání uživatele na jinou webovou stránku prostřednictvím parametru uvedeného v URL.

11.5 Integrace sociálních sítí

Sociální sítě jsou v dnešní době důležitým marketingovým nástrojem. Umožňují firmám a jiným soukromým subjektům působit na cílovou skupinu zákazníků a poskytovat jim zajímavé informace.

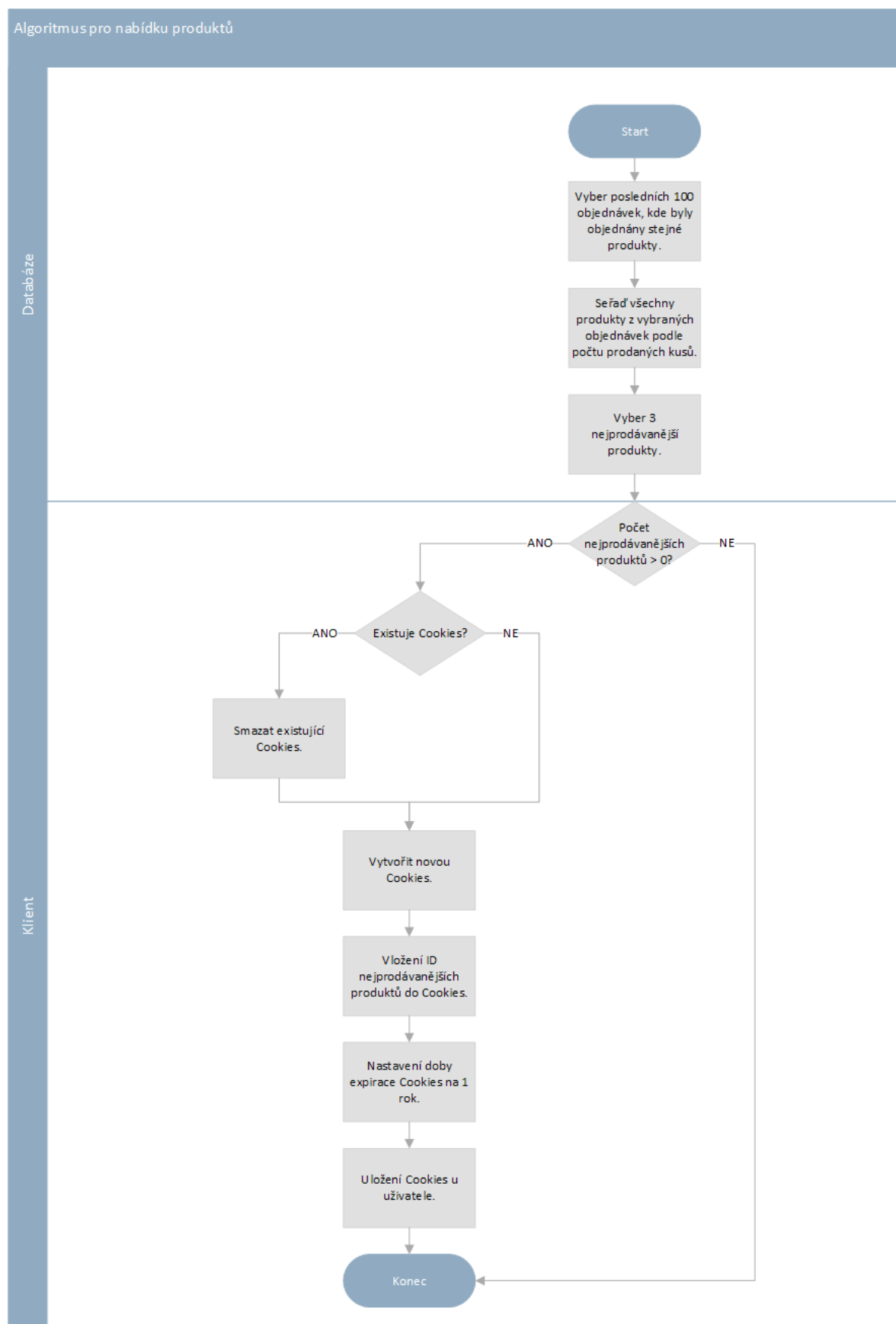
K integraci tlačítek sociálních sítí jsem použil API služby AddThis²², která umožňuje vkládat na webové stránky tlačítka většiny sociálních sítí. Výhodou tohoto řešení je snadná implementace díky konfiguratoru, který generuje kód na základě zvolených parametrů. Další výhodou je možnost sledování statistik počtu sdílení na sociálních sítích a zpětné vazby ze sociálních sítí.

```
<div id="DivSocial" class="addthis_toolbox addthis_counter_style">
  <a class="addthis_button_facebook_like" <%= "fb:like:layout"%>="box_count"
    <%= "fb:like:width"%>="80"></a>
  <a class="addthis_button_tweet" <%= "tw:count"%>="vertical"></a>
  <a class="addthis_button_google_plusone" <%= "g:plusone:size"%>="tall"></a>
</div>
<script type="text/javascript"> var addthis_config = { "data_track_addressbar":
true };</script>
<script type="text/javascript"
src="//s7.addthis.com/js/300/addthis_widget.js#pubid=****"></script>
```

11.6 Analýza a predikce chování uživatelů

Úvodní stránka internetového obchodu obsahuje krátký seznam vybraných produktů. Uživatelům, kteří ještě neprovedli žádnou objednávku, zobrazuje náhodně vybrané produkty z databáze. Pokud uživatel úspěšně dokončí objednávku, seznam se aktualizuje a zobrazí produkty, které si nakoupili ostatní uživatelé k produktům, které si nakoupil uživatel. Tato informace je uložena do cookies a uložena na počítači uživatele. Při dalších přístupech na úvodní stránku jsou tak uživatelům nabízeny produkty, o které by mohl mít zájem. Celý algoritmus popisuje následující vývojový diagram:

²² Dostupné z: <http://www.addthis.com>



Obr. 11.4 Vývojový diagram algoritmu pro nabídku produktů

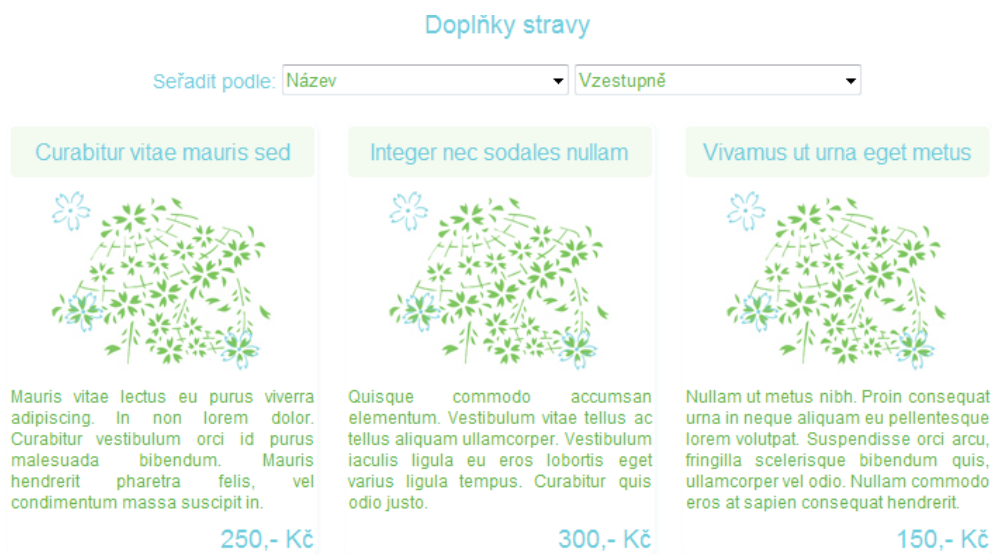
12 POPIS INTERNETOVÉHO OBCHODU

12.1 Katalog produktů

Hlavní součástí internetového obchodu je katalog nabízených produktů. Obsah katalogu je rozdělen do kategorií. Kategorie a seznamy produktů jsou generovány z databáze. Každá položka seznamu produktů obsahuje:

- *Název produktu:* Slouží jako odkaz na detail produktu.
- *Obrázek produktu:* Slouží jako odkaz na detail produktu.
- Anotaci.
- Cenu produktu.

Seznam produktů je ve výchozím stavu seřazen vzestupně podle abecedy. Celý seznam lze dále řadit podle ceny nebo názvu a to v sestupném nebo vzestupném pořadí.



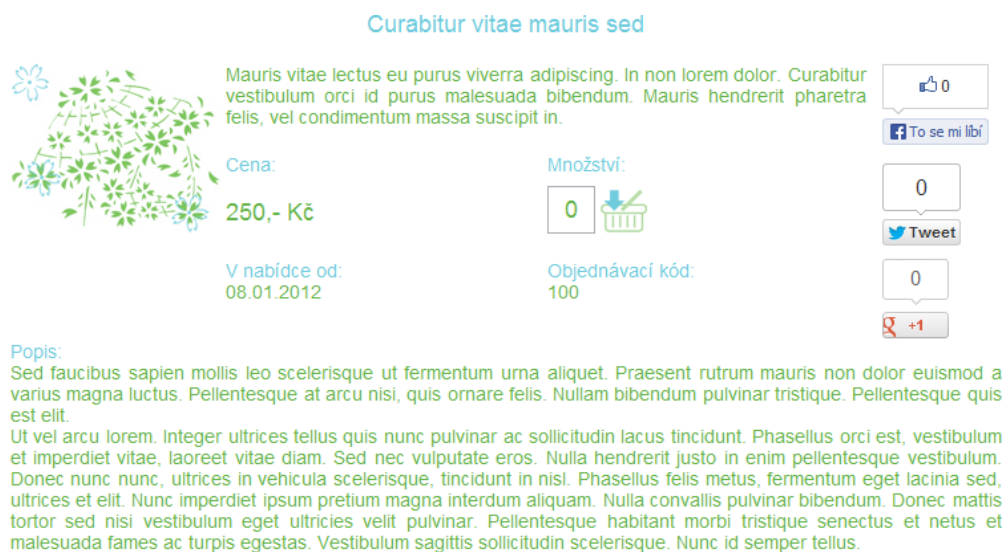
Obr. 12.1 Katalog produktů

12.1.1 Detail produktu

Internetový obchod by měl potenciálním zákazníkům poskytnout dostatek informací o nabízených produktech. Detail produktu obsahuje kromě informací uvedených v seznamu produktů:

- Datum vložení.

- *Podrobný popis*: Slouží jako zdroj informací o produktech v kontextu jejich užívání v souladu s nařízením komise (EU) č. 432/2012.
- *Objednávací kód*: Slouží k jednoznačné identifikaci produktu v případě e-mailové nebo telefonické objednávky.
- *Tlačítko pro vložení do košíku*: Slouží k vložení zadaného množství produktů do nákupního košíku. Při stisknutí tlačítka „Do košíku“ se provede kontrola, zda je již nákupní košík vytvořený. V případě, že nákupní košík není vytvořený, vytvoří se instance třídy NakupniKosik, která se uloží do session²³ a zavolá se metoda PridatDoNakupniKosiku. Pokud je produkt v nákupním košíku již vložený, přičte se vkládané množství produktů k množství produktů v nákupním košíku.
- *Tlačítka pro sdílení na sociálních sítích*: Slouží k sdílení informací o vybraném produktu na sociálních sítích.



Obr. 12.2 Detail produktu

12.2 Nákupní košík

Nákupní košík obsahuje základní informace o produktech připravených k objednávce. Součástí stránky zobrazující obsah nákupního košíku je:

²³ Session respektive session cookie je dočasná cookie označující malé množství dat sloužící v mém případě k uložení nákupního košíku. Session cookie se smaže, pokud uživatel vypne webový prohlížeč. [9]

- *Seznam produktů vložených do košíku:* Slouží k vizualizaci obsahu nákupního košíku. Produkty lze z nákupního košíku odebrat pomocí tlačítka označeného červeným křížkem. Při stisknutí tohoto tlačítka se provede kontrola, zda je již nákupní košík vytvořený. V případě, že nákupní košík není vytvořený, což znamená, že session neexistuje, neprovede se žádná akce a stránka se aktualizuje. V případě, že nákupní košík existuje je zavolána metoda `OdebratZNakupnihoKosiku`. Množství produktů v nákupním košíku lze měnit zapsáním požadovaného množství do příslušného pole a stisknutím tlačítka označeného zelenou šipkou. Při stisknutí tohoto tlačítka se provede kontrola a zavolá se metoda `ZmenitMnozstvi`.
- *Volba platební metody:* Slouží k výběru platební metody.
- *Formulář:* Slouží pro vyplnění dodacích a fakturačních údajů, případně podrobností k objednávce. V případě že je uživatel přihlášený, formulář je skryt a v případě objednávky jsou použity fakturační dodací údaje z profilu uživatele.
- *Tlačítko pro odeslání objednávky:* Slouží k odeslání objednávky. Při stisknutí tlačítka „Odeslat objednávku“ se provede kontrola. V případě, že nákupní košík existuje, zavolá se metoda `OdeslatObjednavku` nebo metoda `OdeslatObjednavkuProfil` v závislosti na tom, zda je uživatel přihlášený. Algoritmus obou metod je obdobný, v databázi se vytvoří záznam o objednávce, na e-mail zákazníka je odeslán e-mail informující o objednaných produktech a na e-mail administrátora je odeslán e-mail informující o příchozí objednávce.

Nákupní košík

Název produktu	Množství	Cena	
Curabitur vitae mauris sed	10 	250 ,- Kč	

Cena: 2500 ,- Kč

Způsob dopravy:
☒ Česká Pošta - Balík do ruky

Způsob platby:
☒ Dobírka

Cena zásilky: ,- Kč



Objednávka Vám bude zaslána na adresu, kterou máte uvedenu v profilu!

Podrobnosti k objednávce:

[Odeslat objednávku](#)

Obr. 12.3 Nákupní košík – Registrovaný uživatel

Nákupní košík

Název produktu	Množství	Cena	
Curabitur vitae mauris sed	10 	250 ,- Kč	

Cena: 2500 ,- Kč

Způsob dopravy:
☒ Česká Pošta - Balík do ruky

Způsob platby:
☒ Dobírka

Cena zásilky: ,- Kč

Fakturační a dodací údaje:

Jméno:

Příjmení:

E-Mail:

Telefon: +420

Ulice:

Město:

PSČ:

Podrobnosti k objednávce:

[Odeslat objednávku](#)

Obr. 12.4 Nákupní košík – Neregistrovaný uživatel

12.3 Úvodní stránka

První stránkou, se kterou se setká většina uživatelů při vstupu do internetového obchodu, je úvodní stránka. Úvodní stránka hraje velmi důležitou informační a marketingovou roli,

měla by potencionálním zákazníkům zobrazovat aktuální informace o článcích, produktech nebo jiných aktualizacích. Úvodní stránka obsahuje:

- *Slider*: Slouží k zobrazování vybraných produktů nebo článků, případně jiných aktualizací.
- *Seznam vybraných produktů*: Slouží k zobrazení náhodně vybraných produktů nebo produktů, které si koupili ostatní zákazníci.
- *Seznam nových článků*: Slouží k zobrazení nově přidaných článků.



Doplňky Stravy

Vitazyme Energy
Cena: 755,- Kč

Nejprodávanější produkty:

Curabitur vitae mauris sed	Vivamus ut urna eget metus	Integer nec sodales nullam
 <p>Mauris vitae lectus eu purus viverra adipiscing. In non lorem dolor. Curabitur vestibulum orci id purus malesuada bibendum. Mauris hendrerit pharetra felis, vel condimentum massa suscipit in.</p> <p>250,- Kč</p>	 <p>Nullam ut metus nibh. Proin consequat urna in neque aliquam eu pellentesque lorem volutpat. Suspendisse orci arcu, fringilla scelerisque bibendum quis, ullamcorper vel odio. Nullam commodo eros at sapien consequat hendrerit.</p> <p>150,- Kč</p>	 <p>Quisque commodo accumsan elementum. Vestibulum vitae tellus ac tellus aliquam ullamcorper. Vestibulum iaculis ligula eu eros lobortis eget varius ligula tempus. Curabitur quis odio justo.</p> <p>300,- Kč</p>

Nejnovější článek:



Aliquam sapien diam, volutpat sed porttitor accumsan, placerat 08.02.2013
 Aliquam auctor leo eget erat fringilla convallis. Ut nulla tellus, commodo et ultricies at, consequat at odio. Morbi elementum, leo in dignissim facilisis, massa erat tempus mi, vitae condimentum odio est vel metus. Suspendisse a purus lectus, id ultrices libero. Nam justo ipsum, porta at fermentum in, lobortis ac turpis. Cras ornare ligula sed ipsum laoreet a pharetra massa tincidunt. Cras sit amet justo nec eros mattis hendrerit. Aliquam erat volutpat.

Kategorie: Produkty

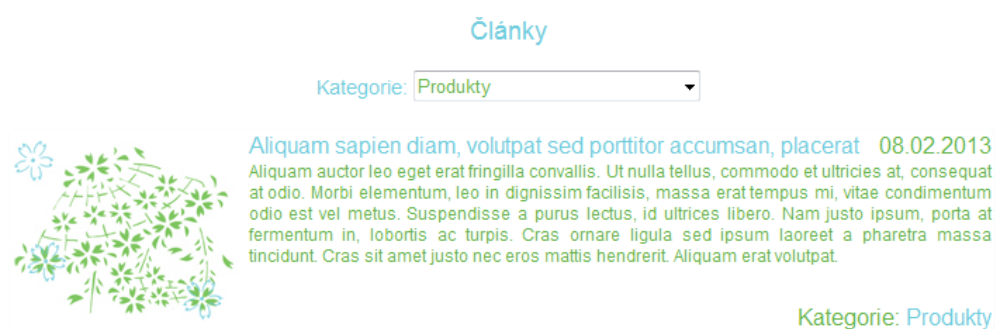
Obr. 12.5 Úvodní stránka internetového obchodu

12.4 Články

Jako zdroj informací o produktech, zkušenostech z jejich užívání nebo jiných informací slouží sekce článků. Obsah sekce s články je rozdělen do kategorií článků. Kategorie a seznamy článků jsou generovány z databáze. Každá položka seznamu článků obsahuje:

- *Název článku:* Slouží jako odkaz na celý text článku.
- *Ilustrační obrázek:* Slouží jako odkaz na celý text článku.
- Datum vložení.
- Anotaci.
- Kategorii.

Celý seznam článků je seřazen podle data vložení.



Obr. 12.6 Články

12.4.1 Obsah článku

Obsah článku obsahuje kromě informací uvedených v seznamu článků:

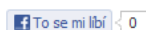
- Text článku.
- *Tlačítka pro sdílení na sociálních sítích:* Slouží k sdílení informací o vybraném článku na sociálních sítích.

Aliquam sapien diam, volutpat sed porttitor accumsan, placerat

Aliquam auctor leo eget erat fringilla convallis. Ut nulla tellus, commodo et ultricies at, consequat at odio. Morbi elementum, leo in dignissim facilisis, massa erat tempus mi, vitae condimentum odio est vel metus. Suspendisse a purus lectus, id ultrices libero. Nam justo ipsum, porta at fermentum in, lobortis ac turpis. Cras ornare ligula sed ipsum laoreet a pharetra massa tincidunt. Cras sit amet justo nec eros mattis hendrerit. Aliquam erat volutpat.

Ut ac arcu odio. Aliquam sed odio ante, a porttitor quam. Aliquam neque quam, fermentum ut tincidunt eu, scelerisque a ipsum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Curabitur faucibus metus quis ligula luctus vel hendrerit tellus mattis. Donec sagittis cursus vehicula. Fusce ut tortor nisi. Maecenas sit amet dui dapibus odio aliquam pellentesque. Suspendisse potenti. In ut pretium odio. Cras magna leo, eleifend at interdum eu, pellentesque in nisi.

Etiam et lectus vitae lacus pellentesque facilisis in sit amet odio. Nunc placerat luctus consequat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam erat volutpat. Ut gravida pretium justo eget varius. Praesent sodales, dolor et mattis sodales, nisi dui faucibus lectus, ac tincidunt urna diam eu nisi. In augue magna, sollicitudin quis ultricies vitae, vehicula ornare nisi. Proin dapibus sapien est. Integer nec ligula sit amet odio cursus condimentum consequat id nibh. Proin vehicula diam id purus iaculis vitae dignissim augue eleifend. Duis iaculis ultrices tempor. Quisque odio ante, facilisis eget bibendum eu, scelerisque et massa. Donec risus lectus, lobortis id tempus at, pretium in massa.



Obr. 12.7 Obsah článku

12.5 Profil

Každý registrovaný uživatel má po přihlášení ke svému účtu přístup do svého profilu.

Profil uživatele obsahuje:

- Odkaz na formulář pro změnu hesla.
- Odkaz na formulář pro změnu dodacích a fakturačních údajů.
- *Historii objednávek*: Slouží k prohlížení stavu a obsahu odeslaných objednávek.

Profil

Nastavení:

[Změnit heslo](#)

[Změnit údaje](#)

Údaje o uživateli:

Pokud se rozhodnete nakupovat v našem obchodu, ujistěte se prosím, že jsou všechna pole vyplněná.

Jméno: Tomáš

Příjmení: Říha

E-Mail: tomas.riha@gmail.com

Telefon:

Ulice:

Město:

PSČ:

Moje objednávky:

ID Objednávky	Datum vytvoření	Stav	
1012	25.04.2013	Odesláno	
1004	21.02.2013	Nevyřízeno	
1000	21.04.2013	Nevyřízeno	

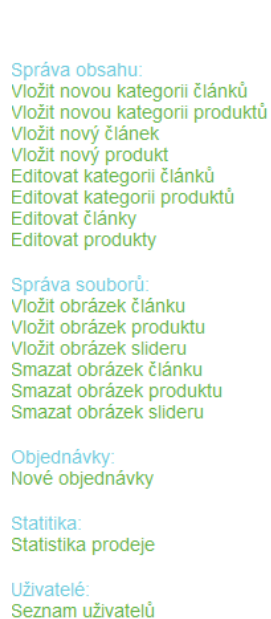
Obr. 12.8 Uživatelský profil

12.6 Administrace

Administrátorské rozhraní by mělo vlastníkově internetového obchodu umožnit spravovat obsah internetového obchodu a vyřizovat objednávky. Administrátorské rozhraní umožňuje:

- *Správa kategorií článků*: Slouží k vkládání/upravování/mazání kategorií článků.
- *Správa článků*: Slouží k vkládání/upravování/mazání článků.
- *Správa kategorií produktů*: Slouží k vkládání/upravování/mazání kategorií produktů.
- *Správa produktů*: Slouží k vkládání/upravování/mazání produktů.
- *Správa obrázků slideru*: Slouží k vkládání/mazání obrázků slideru.
- *Správa vložených obrázků*: Slouží k vkládání/mazání obrázků.
- *Správa objednávek*: Slouží k vyřizování/mazání objednávek.
- *Hodnocení prodeje*: Slouží k měsíčnímu hodnocení množství příchozích objednávek a prodaných produktů.
- *Správa uživatelů*: Slouží k přístupu k dodacím a fakturačním údajům uživatelů nebo k smazání uživatelského účtu.

Pokud administrátor vloží/smaže článek nebo produkt nebo odstraní celou kategorii, dojde k aktualizaci souboru sitemap.xml.



Obr. 12.9 Administrátorské rozhraní

12.7 Náměty na vylepšení

Uvedený internetový obchod je vytvořený pomocí technologie ASP.NET založené na webových formulářích. Nevýhodou tohoto řešení je, že webová aplikace nevyužívá žádný z návrhových vzorů. Nevyužitím návrhového vzoru klesá přehlednost struktury webové aplikace, protože velká část programového kódu je umístěna v tzv. Code Behind file. Využíváním Code Behind file tak klesá možnost znovu využitelnosti programového kódu a možnost rozšíření webové aplikace. Z dlouhodobého hlediska je tak webová aplikace hůře udržitelná.

Další možností, jak webovou aplikaci vylepšit je nasazení ADO.NET Entity Frameworku, který by výrazně ulehčil manipulaci s daty umístěnými v databázi.

ZÁVĚR

Cílem bakalářské práce byla analýza a implementace moderních metod internetového obchodu.

Základem každé dynamické webové stránky je bezpečný a výkonný aplikační framework. Jednou z možností je .NET Framework vyvíjený firmou Microsoft. Součástí .NET Frameworku je ASP.NET, která slouží k vývoji dynamických webových stránek, webových aplikací a webových služeb.

ASP.NET můžeme rozšířit o AJAX funkcionalitu pomocí ASP.NET AJAX. Jednou z nejdůležitějších vlastností ASP.NET AJAX je možnost obnovovat pouze části webových stránek bez nutnosti obnovení celé webové stránky.

Alternativou k ASP.NET založeného na webových formulářích je návrhový vzor MVC. Pro tvorbu webových aplikací využívajících návrhového vzoru MVC je vyvíjen firmou Microsoft ASP.NET MVC Framework. Výhodou aplikací postavených na MVC je oddělení prezentační, aplikační a datové vrstvy v aplikaci.

Součástí ASP.NET jsou nástroje k zabezpečení webové aplikace. Důležitým aspektem zabezpečení webové aplikace je autentizace uživatelů. Nejčastěji používanou metodou autentizace je autentizace založená na formulářích využívající základních aplikačních služeb. Pro zajištění důvěryhodnosti webové aplikace slouží serverové certifikáty, které mimo jiné umožňují používat zabezpečenou komunikaci SSL/TLS.

Internetové platební systémy slouží provozovatelům internetového obchodu k rozšíření platebních možností za nakoupené zboží. Mezi nejrozšířenější platební systémy v ČR patří PaySec a PayU, které nabízejí široké spektrum platebních metod. Nevýhodou platebního systému PayPal je malé rozšíření a malá podpora ze stran bank v ČR.

Sociální sítě jsou důležitým informačním kanálem. Prostřednictvím sociálních sítí můžeme informovat cílovou skupinu uživatelů o novinkách nebo zvýhodněných nabídkách zboží. Sociální sítě nabízejí široké spektrum prvků k integraci svých služeb do webových aplikací. Jednotlivé prvky se přidávají nejčastěji prostřednictvím JavaScriptu.

Jednou z technik pro analýzu a predikci chování uživatelů je Web Mining. Z hlediska internetového obchodu je důležitou skupinou Web Miningu Web Usage Mining. Web

Usage Mining získává informace z interakce uživatelů s webovou aplikací. Dalším zdrojem informací o uživateli mohou být Java Applety nebo serverové logy.

V praktické části jsem se věnoval analýze a návrhu internetového obchodu. Internetový obchod jsem vytvořil pomocí technologie ASP.NET. Výhodou použitého frameworku je rychlá a snadná implementace všech klíčových částí internetového obchodu a možnost konfigurace aplikace prostřednictvím konfiguračního souboru. Konfiguraci uživatelských skupin a zabezpečení aplikace lze pohodlně provést prostřednictvím Web Site Administration Tool. Prvky sociálních sítí jsem implementoval pomocí API služby AddThis, která mimo jiné umožňuje sledovat statistiky sdílení na sociálních sítích. Jednou z metod analýzy a predikce chování uživatelů je algoritmus pro nabídku zboží, tento algoritmus na základě analýzy objednávek nabídne zákazníkovi internetového obchodu zboží, o které by mohl mít zájem.

Uvedenou aplikaci jsem vytvořil jako možnou alternativu k existujícímu e-shopu firmy PrevenceZdravi.eu.

CONCLUSION

This thesis deals with the analysis and implementation of modern methods for an online store.

The base of every dynamic web site is a secure and fast application framework. One of the possibilities is .NET Framework developed by Microsoft. Part of the .NET Framework is an ASP.NET which is used to develop dynamic web pages, web applications and web services.

The ASP.NET functionality can be extended by AJAX using an ASP.NET AJAX. One of the most important features of the ASP.NET AJAX is the possibility to make partial page refreshes instead of a full refresh.

An alternative to the ASP.NET based on web forms is a design pattern called a MVC. Microsoft has developed an ASP.NET MVC Framework for creating web applications using the MVC. The advantage of applications based on the MVC is the separation of input/output and data-processing in an application.

Tools to secure web applications are part of the ASP.NET. An important aspect of web applications security is user authentication. The most commonly used method of authentication is form-based authentication, using basic application services. To ensure the credibility of a web application server certificates are used, which among other things, allows the use of a secure communication layer, called SSL/TLS.

Internet payment systems are used by online stores operators to expand payment options for purchased goods. PaySec and PayU are two of the most popular payment systems in the Czech Republic which offer a wide range of payment methods. A disadvantage of the PayPal payment system is its small share of this market and low support from banks in the Czech Republic.

Social networks are important information channels. Through the social networks, we can inform the target group of users about news and discounts. Social networks offer a wide range of features to integrate their services into a web application. Each element can be added to a web application using a JavaScript.

One of the analysis and implementation techniques and the prediction of user behaviour is Web Mining. Web Usage Mining is an important element in term of online stores. Web

Usage Mining receives information from users' interactions with a web application. Java Applets or server logs can be other sources of information about users' interactions.

In the practical part of the thesis, I dealt with the analysis and design of a particular online store. For the creation of the online store, I used ASP.NET technology. The advantage of the framework used is the rapid and easy implementation of all key features of the online store and the possibility of application configuration using a configuration file. The configuration of user groups and security configurations can be done using the Web Site Administration Tool. To implement elements of the social networks, I used an API from AddThis service – which, inter alia, allows the monitoring of statistics of sharing on social networks. One of the methods for the analysis and prediction of user behaviour is an algorithm for offering goods which offers goods to customers based on the analysis of previous orders.

The aforementioned e-shop was created as an alternative to the existing e-shop of the PrevenceZdravi.eu company.

SEZNAM POUŽITÉ LITERATURY

- [1] SPAANJAARS, Imar. Beginning ASP.NET 4 in C# and VB. Indianapolis, IN: Wiley Pub., 2010, XXXVI, 803 s. Wrox beginning guides. ISBN 04-705-0221-5.
- [2] Handlers. ISS [online]. © 2013 [cit. 2013-01-22]. Dostupné z: <http://www.iis.net/configreference/system.webserver/handlers#004>
- [3] Configuring an ASP.NET Application for an ASP.NET Version. MSDN [online]. © 2013 [cit. 2013-01-22]. Dostupné z: [http://msdn.microsoft.com/en-us/library/1kdfe21k\(v=vs.71\).aspx](http://msdn.microsoft.com/en-us/library/1kdfe21k(v=vs.71).aspx)
- [4] EVJEN, Bill, Scott HANSELMAN a Devin RADER. Professional ASP.NET 4 in C# and VB. Indianapolis, IN: Wiley Pub., 2010, IVIII, 1477 s. ISBN 04-705-0220-7.
- [5] DORRANS, Barry. Beginning ASP.NET Security. Chichester, U.K.: Wiley, 2010, XX, 412 s. ISBN 978-047-0743-652.
- [6] Zdroják: MVC a další prezentační vzory. Zdroják [online]. 2009 [cit. 2013-02-20]. Dostupné z: <http://www.zdrojak.cz/serialy/mvc-a-dalsi-prezentacni-vzory/>
- [7] Albert Zuurbier - Business Analyst: MVC vs. MVP vs. MVVM. Albert Zuurbier - Business Analyst [online]. 2013 [cit. 2013-02-20]. Dostupné z: <http://www.albertzuurbier.com/index.php/programming/84-mvc-vs-mvp-vs-mvvm>
- [8] CodeProject: Model View Controller, Model View Presenter, and Model View ViewModel Design Patterns. CodeProject [online]. 2009 [cit. 2013-02-20]. Dostupné z: <http://www.codeproject.com/Articles/42830/Model-View-Controller-Model-View-Presenter-and-Mod>
- [9] GOURLEY, David a Brian TOTTY. HTTP: The definitive guide. 1. Vyd. Sebastopol, CA: O'Reilly, 2002, XVIII, 635 s. ISBN 15-659-2509-2.
- [10] Přehled kvalifikovaných poskytovatelů certifikačních služeb a jejich kvalifikovaných služeb. Ministerstvo vnitra České republiky [online]. 1.8.2012 [cit. 2013-01-26]. Dostupné z: <http://www.mvcr.cz/clanek/prehled->

kvalifikovanych-poskytovatelu-certifikacnich-sluzeb-a-jejich-kvalifikovanych-sluzeb.aspx

- [11] Certifikační autorita PostSignum [online]. © 2010 [cit. 2013-01-26]. Dostupné z: <http://www.postsignum.cz>
- [12] APCS eIdentity a.s. [online]. [2004] [cit. 2013-01-26]. Dostupné z: <http://www.eidentity.cz>
- [13] První certifikační autorita a.s. [online]. [2002] [cit. 2013-01-26]. Dostupné z: <http://www.ica.cz>
- [14] OWASP Top 10 - 2010 [online]. 2010 [cit. 2013-03-16]. Dostupné z: https://www.owasp.org/images/0/0f/OWASP_T10_-_2010_rc1.pdf
- [15] ZEMEK, Lukáš. Bezpečnost webových aplikací. Zlín, 2012. Bakalářská práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky.
- [16] OWASP Top 10 for .NET developers part 1: Injection. Troy Hunt [online]. 12.5.2010 [cit. 2013-03-16]. Dostupné z: <http://www.troyhunt.com/2010/05/owasp-top-10-for-net-developers-part-1.html>
- [17] Příručka pro použití Platební brány PaySec. In: Příručka pro použití Platební brány PaySec [online]. © 2007-2012 [cit. 2013-01-26]. Dostupné z: <http://www.paysec.cz/CmsPage.aspx?Id=gatewayManual>
- [18] PayU pro e-shopy [online]. [2011] [cit. 2013-01-26]. Dostupné z: <http://www.payu.cz>
- [19] Accept and Process Payments. Products | X.commerce [online]. © 2013 [cit. 2013-01-26]. Dostupné z: <https://www.x.com/developers/paypal/products>
- [20] Core Concepts. Facebook Developers [online]. © 2013 [cit. 2013-01-22]. Dostupné z: <http://developers.facebook.com/docs/guides/web/>
- [21] Social Plugins. Facebook Developers [online]. © 2013 [cit. 2013-01-22]. Dostupné z: <http://developers.facebook.com/docs/plugins/>
- [22] Facebook Login. Facebook Developers [online]. © 2013 [cit. 2013-01-22]. Dostupné z: <http://developers.facebook.com/docs/concepts/login/>

- [23] Dialogs Overview. Facebook Developers [online]. © 2013 [cit. 2013-01-22]. Dostupné z: <http://developers.facebook.com/docs/reference/dialogs/>
- [24] Open Graph Protocol. Facebook Developers [online]. © 2013 [cit. 2013-01-22]. Dostupné z: <http://developers.facebook.com/docs/opengraphprotocol/>
- [25] Plugins. Google Developers [online]. © 2013 [cit. 2013-01-22]. Dostupné z: <https://developers.google.com/+/plugins/>
- [26] Documentation. Twitter Developers [online]. © 2013 [cit. 2013-01-22]. Dostupné z: <https://dev.twitter.com/docs/>
- [27] ŠTĚPÁNKOVÁ, Kateřina. Modelování predikce časové řady návštěvnosti web domény pomocí RBF neuronových sítí. 2011. 100 s. Dostupné z: http://dspace.upce.cz/bitstream/10195/39809/1/StepankovaK_ModelovaniPredikce_VO_2011.pdf. Diplomová práce. Univerzita Pardubice Fakulta ekonomicko-správní. Vedoucí práce prof. Ing. Vladimír Olej, CSc.
- [28] KOSALA, Raymond a Hendrik BLOCKEEL. Web Mining Research: A Survey [online]. Heverlee (Belgium), 2000 [cit. 2013-01-08]. Dostupné z: <http://arxiv.org/pdf/cs/0011033v1.pdf>. Rešerše. Department of Computer Science Katholieke Universiteit Leuven.
- [29] SRIVASTAVA, Jaideep, Robert COOLEYZ, Mukund DESHPANDE a Pang-Ning TAN. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data [online]. Minneapolis (Minnesota), 2000 [cit. 2013-01-08]. Dostupné z: <http://nlp.uned.es/WebMining/Tema5.Uso/srivastava2000.pdf>. Rešerše. Department of Computer Science and Engineering University of Minnesota.
- [30] NAGEL, Christian. Professional C# 4 and .Net 4. Indianapolis, IN: Wiley Pub., © 2010. ISBN 978-0-470-50225-9.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ADO.NET	Označuje sadu softwarových komponent pro přístup k datům z různých datových zdrojů.
API	Application Programming Interface – Rozhraní pro programování aplikací.
ASP	Active Server Pages – Skriptovací jazyk sloužící k dynamickému zpracování webových stránek na straně serveru.
ASP.NET	Active Server Pages – Nástupce ASP.
AJAX	Asynchronous JavaScript and XML – Označuje technologii k vytváření webových stránek bez „prokliknutí“.
ASPX	Označení stránek s příponou .aspx.
CSS	Cascading Style Sheets – Jazyk používaný k formátování a vytváření vzhledu webové stránky.
GUID	Globally Unique Identifier – Označuje jednoznační identifikátor.
HTML	HyperText Markup Language – Jazyk používaný prohlížeči k zobrazování webových stránek.
HTML5	HyperText Markup Language 5 – Nový standard jazyka HTML.
HTTP	Hypertext Transfer Protocol – Protokol umožňující komunikaci webového prohlížeče a serveru.
HTTP 1.1	Hypertext Transfer Protocol 1.1 – Verze HTTP protokolu.
HTTPS	Hypertext Transfer Protocol Secure - Protokol umožňující zabezpečenou komunikaci webového prohlížeče a serveru.
IFRAME	Označení pro rámec vložený do HTML.
IIS	Internet Information Services – Webový server vyvinutý firmou Microsoft.
LINQ	Language Integrated Query – Jazyk používaný k manipulaci s daty.
MVC	Mode View Controller – Návrhový vzor.
MVP	Model View Presenter – Návrhový vzor.

MVVP	Model View ViewModel – Návrhový vzor.
SPAM	Označení pro nevyžádanou zprávu.
SQL	Structured Query Language – Jazyk používaný k manipulaci s daty.
SSL	Secure Sockets Layer, SSL – Protokol zabezpečující šifrování dat mezi webovým prohlížečem a serverem.
TCP/IP	Komunikační protokol používaný pro komunikaci v počítačových sítích a Internetu.
TLS	Transport Layer Security – Nástupce SSL.
URL	Uniform Resource Locator – Identifikátor zdroje informací na Internetu.
WCF	Windows Communication Foundation – Framework používaný pro tvorbu distribuovaných aplikací.
WPF	Windows Presentation Foundation – Framework používaný k vykreslování uživatelského rozhraní.
XAML	Extensible Application Markup Language – Jazyk používaný k tvorbě uživatelského rozhraní.
XML	Extensible Markup Language – Jazyk používaný k výměně dat mezi aplikacemi.

SEZNAM OBRÁZKŮ

Obr. 1.1 Funkce HTTP protokolu	12
Obr. 2.1 Běžné zpracování stránky	15
Obr. 2.2 AJAX zpracování stránky	16
Obr. 3.1 Architektura MVC	17
Obr. 3.2 Architektura MVP	18
Obr. 3.3 Architektura MVVM	19
Obr. 4.1 Vrstvy HTTP a HTTPS	24
Obr. 4.2 Komunikace HTTP a HTTPS	25
Obr. 4.3 Schéma webové stránky.....	29
Obr. 6.1 Like Button [21]	37
Obr. 6.2 Send Button [21].....	38
Obr. 6.3 Follow Button [21]	38
Obr. 6.4 Comment Box [21]	39
Obr. 6.5 Activity Feed [21].....	39
Obr. 6.6 Recommendation Box [21].....	40
Obr. 6.7 Recommendations Bar [21]	40
Obr. 6.8 Like Box [21].....	41
Obr. 6.9 Login Button [21]	41
Obr. 6.10 Registration [21]	41
Obr. 6.11 Open Graph [24]	44
Obr. 6.12 +1 Button [25]	45
Obr. 6.13 Google+ Badge [25]	45
Obr. 6.14 Share [25]	45
Obr. 6.15 Snippet [25]	46
Obr. 6.16 Tweet Button [26].....	46
Obr. 6.17 Folow Button [26]	46
Obr. 6.18 Embedded Tweets [26]	47
Obr. 6.19 Embedded Timelines [26]	47
Obr. 6.20 Twitter Cards [26]	47
Obr. 8.1 Diagram užití	54
Obr. 10.1 Návrh blokového rozmístění prvků	57

Obr. 10.2 Výsledný vzhled internetového obchodu.....	58
Obr. 11.1 Sestavení výsledné stránky	60
Obr. 11.2 Databázový diagram	61
Obr. 11.3 Okno Web Site Administration Tool.....	63
Obr. 11.4 Vývojový diagram algoritmu pro nabídku produktů	70
Obr. 12.1 Katalog produktů	71
Obr. 12.2 Detail produktu	72
Obr. 12.3 Nákupní košík – Registrovaný uživatel	74
Obr. 12.4 Nákupní košík – Neregistrovaný uživatel.....	74
Obr. 12.5 Úvodní stránka internetového obchodu	75
Obr. 12.6 Články.....	76
Obr. 12.7 Obsah článku	77
Obr. 12.8 Uživatelský profil	77
Obr. 12.9 Administrátorské rozhraní	79

SEZNAM PŘÍLOH

P I CD

PŘÍLOHA P I: CD

CD obsahuje:

- Elektronickou podobu bakalářské práce.
- Zdrojové kódy internetového obchodu.