

PŘÍRODOVĚDECKÁ FAKULTA UNIVERZITY PALACKÉHO
KATEDRA INFORMATIKY

DIPLOMOVÁ PRÁCE

Rich Internet Applications



2011

Tařána Kubátová

Anotace

Cílem mé diplomové práce bylo provést analýzu vybraných RIA technologií a podpory multimédií v těchto technologiích. Pro vybranou technologii poté vytvořit aplikaci.

Ráda bych touto cestou poděkovala vedoucímu diplomové práce RNDr. Davidu Skopilovi za odborné vedení a poskytnutí konzultací. Dále bych ráda poděkovala Bc. Jakubovi Semrádovi a Mgr. Pavlovi Kubátovi za podporu a trpělivost.

Obsah

1. Úvod	8
2. Historie	9
2.1. Zrození WWW	9
2.2. HTTP Protokol	10
2.3. Historický vývoj WWW stránek a aplikací	11
2.4. Web 2.0	12
3. RIA	14
3.1. Úvod	14
3.2. RIA Technologie	15
3.3. Multimédia	15
3.4. Metoda analýzy	17
3.5. Adobe Flash	17
3.5.1. Úvod	17
3.5.2. Popis	19
3.5.3. Podpora v multimédiích	19
3.5.4. Výhody a nevýhody	20
3.5.5. Příklad	21
3.6. Adobe Flex	21
3.6.1. Úvod	21
3.6.2. Popis	22
3.6.3. Podpora v multimédiích	24
3.6.4. Výhody a nevýhody	25
3.6.5. Příklad	25
3.7. Ajax	26
3.7.1. Úvod	26
3.7.2. Popis	27
3.7.3. Podpora v multimédiích	29
3.7.4. Výhody a nevýhody	30
3.7.5. Příklad	30
3.8. JavaFX	31
3.8.1. Úvod	31
3.8.2. Popis	32
3.8.3. Podpora v multimédiích	34
3.8.4. Výhody a nevýhody	35
3.8.5. Příklad	35
3.9. Microsoft Silverlight	36
3.9.1. Úvod	36
3.9.2. Popis	38
3.9.3. Podpora v multimédiích	40

3.9.4.	Výhody a nevýhody	40
3.9.5.	Příklad	41
4.	Implementovaná RIA Aplikace	42
4.1.	Úvod	42
4.2.	Základní popis systému	42
4.3.	Použité nástroje	43
4.3.1.	Expression Blend 4.0	43
4.3.2.	Fiddler	43
4.3.3.	FFMpeg	43
4.3.4.	IIS 7	43
4.3.5.	Internet Explorer 8, Google Chrome 8, Firefox 3.5	43
4.3.6.	Microsoft SQL Management Studio	43
4.3.7.	Visual Studio 2010 Ultimate	43
4.4.	Použité technologie	44
4.4.1.	ASP.NET 4.0	44
4.4.2.	C# 4.0	44
4.4.3.	Linq (Language Integrated Query)	44
4.4.4.	Microsoft Silverlight 4.0	44
4.4.5.	MS SQL 2008 R2 Express/Web Edition	44
4.4.6.	Silverlight Toolkit	44
4.4.7.	Silverlight 4 Tools	45
4.5.	Analýza aplikace	45
4.5.1.	Prezentační vrstva	45
4.5.2.	Vrstva pro přístup k datům a logická vrstva	48
4.5.3.	Návrh databáze	49
4.6.	Uživatelská dokumentace	50
4.6.1.	Spuštění aplikace	50
4.6.2.	Úvodní stránka	52
4.6.3.	Zobrazení stránky pro přehrání videa	53
4.6.4.	Registrace	54
4.6.5.	Přihlášení	54
4.6.6.	Administrační Menu	55
4.6.7.	Vytvoření zástupce na ploše	57
	Závěr	58
	Conclusions	59
	Reference	60
	A. Obsah příloženého CD	61

Seznam obrázků

1.	Aplikace Pixlr	21
2.	Průběh kompilace	24
3.	Aplikace Grooveshark	25
4.	Aplikace Global Debt Comparsion	31
5.	Průnik profilů	34
6.	Aplikace Vancouver 2010	36
7.	Architektura Silverlight	39
8.	Aplikace MSTV	41
9.	Rozdělení stránky	46
10.	Schéma databáze	49
11.	Obrazovka - není nainstalován plug-in pro Silverlight	51
12.	Instalace	51
13.	Úvodní stránka	52
14.	Stránka pro přehrávání videa	53
15.	Přehrávač videa	53
16.	Editor pro psaní poznámek	54
17.	Registrační formulář	54
18.	Přihlašovací formulář	55
19.	Vytvoření nové skupiny	56
20.	Připojení skupiny	56

Seznam tabulek

1.	Historický přehled vývoje Adobe Flash	18
2.	Historický přehled vývoje Adobe Flex	22
3.	Historický přehled vývoje JavaFX	32
4.	Historický přehled vývoje Silverlight	38

1. Úvod

S webem se dnes setkáváme prakticky každý den. Život bez něj si snad už ani neumíme představit. Používáme jej k vyhledávání informací, ať už do školy nebo do práce, komunikaci s přáteli, či kolegy, k nakupování, ke čtení zpráv z domova či ze světa, ke vstupu do internetového bankovníctví, atd. . .

V této diplomové práci se nejprve zaměřím na historický přehled vývoje webu, co bylo počátečním impulzem jeho vzniku a jakými důležitými historickými mezníky prošel. V další kapitole se rozepíši o trendu dnešní doby - tzv. bohatých internetových aplikacích (RIA). Zaměřím se, jak na vysvětlení tohoto pojmu, tak na nejpoužívanější technologie pro tvorbu těchto aplikací. U každé technologie najdete historický přehled, princip této technologie a popis možností využití multimédií. Poslední část této diplomové práce je věnována demonstraci vybrané technologie na reálném projektu. V této části najdete, jak programátorskou, tak uživatelskou dokumentaci. K práci je přiloženo CD s projektem. Popis adresářové struktury najdete v příloze.

2. Historie

2.1. Zrození WWW

WWW je anglickou zkratkou pro World Wide Web (později pouze web). Tato zkratka je označením pro systém provázaných hypertextových dokumentů na Internetu.

Základní myšlenkou pro vytvoření webu bylo sjednocení informací a zjednodušení sdílení těchto informací pro vědecké pracovníky ve společnosti CERN¹. Problémem byla především praktická implementace této myšlenky. Roku 1989 sepsal fyzik Tim Berners-Lee jeden z prvních návrhů, jak toto propojení dokumentů realizovat. Hlavní myšlenkou byl tzv. Hypertext, propojení jednotlivých dokumentů pomocí odkazů na další dokument. V dnešní době je tato věc samozřejmostí, v tehdejší době se však jednalo o revoluční myšlenku.

Tim Berners-Lee stál u zrodu jazyka HTML² a protokolu pro přenos v síti HTTP³, o kterém si povíme později. Jazyk HTML byl založen na v té době již existujícím formátu SGML⁴. Tento jazyk je charakterizován množinou značek a jejich atributů. Mezi tyto značky se vkládá text dokumentu a tím se určuje význam (sémantika) obsaženého textu.

Byl to právě Tim Berners-Lee, kdo naprogramoval první server a vytvořil první prohlížeč nazvaný WorldWideWeb v roce 1991. Tento prohlížeč umožňoval, jak hypertextové dokumenty prohlížet, tak je editovat a je zároveň prvním WYSIWYG⁵ editorem. Fungoval však pouze na počítačích Next s operačním systémem NeXTStep. Později byl tento prohlížeč přejmenovaný na Nexus.

Zlomovým datem pro historii webu je 30. duben 1993, kdy společnost CERN oznámila, že WWW bude šířena jako bezplatná služba. V tomto roce byl také představen webový prohlížeč Mosaic později Mosaic Nescape. Byl to první webový prohlížeč, který fungoval jak na UNIXu⁶, Windowsu tak i na operačním systému pro Apple Macintosh. Mosaic byl někdy označován jako tzv. Killer application devadesátých let. Toto označení dostávaly programy, které měly pro uživatele tak veliký význam, že si kvůli nim uživatel koupil počítač nebo daný operační systém, jen aby tento program mohl mít. Na základech Mosaicu byl vystavěn i

¹CERN - z francouzského Conseil Européen pour la recherche nucléaire - mezinárodní organizace pro jaderný výzkum.

²HTML - z anglického HyperText Markup Language - značkovací jazyk pro vytváření webových stránek.

³HTTP - z anglického HyperText Transfer Protocol - internetový protokol pro přenos dat mezi klientem a serverem.

⁴SGML - z anglického Standard Generalized Mark-Up Language - univerzální značkovací metajazyk.

⁵WYSIWYG - z anglického What you see is what you get - touto zkratkou se označují editory, u kterých je verze dokumentu zobrazená na obrazovce vzhledově totožná s výslednou verzí dokumentu.

⁶UNIX je operační systém.

první prohlížeč Microsoftu Internet Explorer.

Web se začínal stávat velmi populární a na konci roku 1993 bylo známo již kolem několika set webových serverů. Dnes odhady ukazují, že je na světě více než 165 miliónů serverů a více než 45 miliard indexovatelných webových stránek [9].

2.2. HTTP Protokol

Žádná z výše popsaných událostí by nenastala bez HTTP protokolu, proto i jeho vývoj je velmi důležitý a protože byl nedílnou součástí historie, je třeba se mu i náležitě věnovat v obecné historii Internetu.

Jak již bylo zmíněno, pro přenos dat mezi klientem a serverem se používá protokol HTTP. Klientem je nejčastěji webový prohlížeč a serverem bývá nejčastěji webový server. Protokol je bezstavový a pracuje na principu požadavek/odpověď (Request/Response). Uživatel pomocí klienta zašle požadavek na server. Požadavky mohou být zaslány pomocí různých dotazovacích metod, nejčastěji to jsou metody GET a POST. V případě použití metody GET se formulářová data stávají součástí URL a tělo požadavku je vždy prázdné. V případě použití metody POST se data předávají většinou v těle dotazu. Server obdrží požadavek a zpracuje jej. Výsledek vrátí uživateli v odpovědi.

Protokol během let prošel následujícím vývojem:

První verze HTTP 0.9 byla publikována v roce 1991. Požadavek mohl být zaslán pouze pomocí metody GET s jedním parametrem. Jako odpověď ze serveru byl poslán přímo vyžádaný dokument HTML bez hlavičky. V případě, že nastala chyba (například dokument nebyl nalezen), vracela se chybová hláška pouze ve formátu HTML dokumentu.

Další verzí byla HTTP 1.0. Tato verze byla rozšířena o další metody HEAD a POST a při přenosu se navíc používají také HTTP hlavičky. Hlavičky slouží pro přenos různých doplňujících informací (Content-Type⁷, Date⁸, atd...).

Poslední dnes používanou verzí je HTTP 1.1. U předchozích verzí se vždy navazovalo nové spojení mezi klientem a serverem, toto navazování spojení však bylo časově náročné. Proto se v této verzi spojení po vyřízení požadavku ihned neuzavírá, ale umožňuje jej použít pro přenos více požadavků. Tato verze byla opět rozšířena o další metody OPTIONS, PUT, DELETE, TRACE a CONNECT.

⁷HTTP hlavička Content-Type udává typ zasílaných dat. Tento typ může nabývat hodnot např.: text/html, image/gif, application/octet-stream.

⁸HTTP hlavička Date udává okamžik vytvoření zprávy.

2.3. Historický vývoj WWW stránek a aplikací

Původním typem WWW stránek byly tzv. statické stránky. Statickými stránkami rozumíme HTML dokumenty, které jsou na serveru uloženy přesně v takové podobě, v jaké jsou poté přeneseny a zobrazeny uživateli.

Dalším typem WWW stránek byly dynamicky generované HTML dokumenty. Stránky již nejsou uloženy na serveru, ale jsou dynamicky generovány podle zaslání požadavku na server. Hlavní úlohu zde má tzv. aplikační server, který zpracovává dotaz od uživatele a za pomoci vlastního programovacího jazyka (např. PHP⁹, Java¹⁰, ASP.NET¹¹, atp.) a databáze vygeneruje HTML výstup, který je poté odeslán jako odpověď uživateli. Skriptování tedy probíhá na straně serveru.

V roce 1995 přichází firma Netscape s novým prohlížečem, který má v sobě integrovaný programovací jazyk LiveScript, později přejmenovaný na JavaScript. JavaScript má sloužit jako doplněk k jazyku HTML. Tímto je odstartována nová éra webových stránek, skriptování již neprobíhá pouze na straně serveru, ale může probíhat i na straně klienta. Tento jazyk prošel do dnešní doby obrovským vývojem. Zmínila bych snad jen některá významná období vývoje. Příchod prohlížeče Netscape Navigator 3.0 s novou implementací JavaScriptu verze 1.2. V té době vstupuje do hry firma Microsoft a do prohlížeče Internet Explorer 3.0 integruje programovací jazyk JScript, který vychází z JavaScriptu. Jazyky byly téměř totožné, ale nebyly zcela kompatibilní. Vývoj pokračuje dále a Netscape i Microsoft se snaží vylepšovat jak JavaScript tak JScript, až nakonec do tohoto souboje zasahuje mezinárodní standardizační asociace ECMA¹² v roce 1997, která standardizuje verzi JavaScriptu a tato standardizovaná verze se nazývá ECMAScript. V dnešní době je aktuální verzí JavaScript 1.9 a tato verze je například integrována v prohlížeči Mozilla Firefox 3.5. a aktuální verzí JScriptu je 5.8, která je integrována v prohlížeči Internet Explorer 8.0. Díky neustálé standardizaci se udržuje rozdíl mezi těmito jazyky na v celku přijatelné úrovni, ale i přes to stále existují určité implementační rozdíly, které programátorům stěžují jejich práci na vývoji WWW aplikací. Web se tedy díky JavaScriptu stává více dynamickým.

Všechny předchozí WWW aplikace používají tzv. synchronní přístup komunikace klient/server.

V roce 1996 přichází Microsoft s novou verzí webového prohlížeče Internet Explorer, který obsahuje nový element IFRAME. IFRAME byl rámeček, který byl vložený do WWW stránky, ve kterém byla zobrazena další WWW stránka. Vývojáři začali IFRAME využívat následujícím způsobem, nastavovali IFRAMu nulovou velikost, rámeček tedy nebyl vidět a při nějaké akci (například klepnutím

⁹PHP je rekurzivní zkratkou pro Hypertext Preprocessor původně Personal Home Page - skriptovací programovací jazyk.

¹⁰Java je objektově orientovaný programovací jazyk, který vyvinula firma Sun Microsystems.

¹¹ASP.NET - z anglického Active Server Pages je součástí .NET Frameworku pro tvorbu webových aplikací a služeb.

¹²ECMA - European Computer Manufacturers Association.

na tlačítko), byla tato událost zachycena pomocí JScriptu a v rámci byl poté poslán dotaz na webový server, který vrátil odpověď. Tato odpověď se zobrazila v neviditelném rámci, ze kterého opět pomocí JScriptu byl výsledek zobrazen uživateli. Již v té době se tedy mohlo využívat asynchronní komunikace se serverem. Další verze Netscape Navigator již také obsahuje nový element pod názvem Layer, který má podobný účel jako IFRAME u Internet Exploreru. Jak je vidět v té době jsou největšími rivaly na poli webových prohlížečů právě Internet Explorer a Netscape Navigator. V roce 1998 představuje Microsoft novou technologii nazvanou Remote Scripting. Tato technologie byla založena na běhu Java Appletu¹³. Pomocí tohoto JavaAppletu docházelo ke komunikaci klinta se serverem, přičemž odpovědi byly zpracovávány pomocí JavaScriptových funkcí. V roce 1999 vychází webový prohlížeč Internet Explorer 5.0, ve kterém je využíván nový objekt XMLHttpRequest, který umožňoval asynchronní volání serveru. Tento objekt zprostředkoval komunikaci server-klient pomocí protokolu HTTP. Postupně je tento objekt implementován i v ostatních prohlížečích a začíná se plně využívat ve webových aplikacích. Právě na tomto principu je postavena technologie AJAX¹⁴. A byl to právě AJAX, který stál u zrodu Web 2.0 aplikací a RIA aplikací, které pracují s asynchronním voláním serveru.

2.4. Web 2.0

Pojem Web 2.0 se poprvé objevuje v roce 1999, Darcy DiNucci jej použila v článku "Fragmented future". Píše v něm:

"Web, jak ho známe teď, který se jako statický text načte do okna prohlížeče, je jen zárodek webu, který přijde. První záblesky Webu 2.0 se již začínají objevovat a my sledujeme, jak se toto embryo začíná vyvíjet. Web bude chápán ne jako obrazovky plné textu a grafiky, ale jako prostředí, jako éter, jehož prostřednictvím dochází k interaktivitě. Objeví se na obrazovce počítače, na televizním přijímači, na palubní desce, na mobilním telefonu, na herní konzoli, a možná, že i na vaší mikrovlnné troubě."

Toto byl však jen nástin budoucnosti webu. Tento pojem se dostal do povědomí široké veřejnosti v roce 2004, kdy Tim O'Reilly a zástupci MediaLive International zvažovali název připravované konference a nakonec dospěli k názvu Web 2.0. Tento název měl symbolizovat směr, kterým se webové aplikace ubíraly.

Co vlastně přesně definujeme pod pojmem Web 2.0?

Pokud se budeme snažit najít přesnou definici, neuspějeme. Tento pojem přesnou definici nemá. V některých článcích dokonce můžeme najít, že tento pojem je pouze nafouknutá bublina - reklamní trik. Většina zdrojů se však spojuje v hlavní myšlence. Web 2.0 není pojmem technickým či technologickým, jedná se spíše o nový přístup k vytváření obsahu a sdílení informací.

¹³Java Applety byly jednoduché programky, které se stahovaly z webového serveru a spouštěly se uvnitř WWW stránky na klientském počítači.

¹⁴AJAX - z anglického Asynchronous JavaScript and XML.

Charakteristickými rysy jsou:

- Uživatel již není pasivním konzumentem, stává se aktivním spolutvůrcem obsahu.
- Sociální fenomén – otevřená komunikace, decentralizace autorit, sdílení a znovuvyužití informací.
- Více organizovaný a roztříděný obsah s propracovanější hyperlinkovou strukturou.
- Bohatý uživatelský zážitek.

Nejsou to rysy bezpodmínečně aplikovatelné na každou web 2.0 aplikaci. Každá aplikace, která je označena za web 2.0 aplikaci může splňovat pouze některý z výše zmíněných bodů.

Příklady stránek, které vyhovují některým z charakteristických rysů, jsou:

<http://www.wikipedia.org/> - webová encyklopedie, jejíž obsah je tvořen dobrovolnými přispěvateli.

<http://www.facebook.com/> - sociální síť, sloužící ke komunikaci a sdílení dat.

<http://www.youtube.com/> - portál pro sdílení videí.

<http://blog.respekt.ihned.cz/> - aplikace obsahující příspěvky a názory uživatelů.

Určitě stojí za zmínku často se vyskytující pojem Mashup, který také spadá do slovníku pojmů Webu 2.0. Mashup je webová aplikace kombinující obsah z více zdrojů do jednoho celku. Nejoblíbenějšími zdroji pro tvorbu Mashupu jsou GoogleMaps, Flickr, Amazon a YouTube. Příkladem takovéto aplikace může být <http://3dtour.cz/> - aplikace prezentující zajímavá města na světě.

3. RIA

3.1. Úvod

Určitě nejsem sama, kdo viděl mnoho stránek na webu s nálepkou RIA aplikace. Nebo alespoň články popisující budoucnost internetu, ve kterých se o této zkratce zmiňují.

Ria je zkratka pro Rich Internet Applications, do češtiny by se tento název dal volně přeložit jako Bohaté internetové aplikace. Stejně jako Web 2.0 také definice pro RIA není přesně nadefinována.

Co si tedy pod názvem RIA přesně představit? RIA aplikace jsou takové webové stránky, které se chováním přibližují tradičním desktopovým aplikacím. Charakteristickými rysy jsou:

- Silné spojení s webem – obecně není podmínkou, aby RIA aplikace musela běžet pouze ve webovém prohlížeči, ale k tomu, aby správně aplikace fungovala, potřebuje komunikovat s nějakým vzdáleným serverem. Příkladem může být například aplikace eBayDesktop.
- Aplikace nevyžaduje žádnou nebo pouze minimální instalaci.
- Uživatelské rozhraní – uživatelské rozhraní by se mělo blížit tradičním desktopovým aplikacím, jak ve smyslu vzhledu, tak i v použití klávesových zkratk a rychlou odezvou.
- Aplikace by měla být multiplatformní.

Proč se vlastně RIA aplikace stávají čím dál tím více oblíbené? Ačkoliv vytváření webových aplikací je mnohem komplikovanější a mají více omezení než obyčejné desktopové aplikace, uživatelé mají tyto aplikace v oblibě. Uživatelé si obecně neradi instalují nové programy a instalaci se snaží pokud možno vyhnout. Často se u instalace stává, že si uživatel neví rady a potřebuje pomoc. Toto u RIA aplikací zcela odpadá. Uživatel jednoduše spustí prohlížeč a aplikace je okamžitě připravena k použití. Uživatel může aplikaci spustit na kterémkoliv počítači. Jak bylo zmíněno v charakteristických rysech, aplikace bývají multiplatformní, takže si uživatel nemusí lámat hlavu, že momentálně nepoužívá svůj počítač v práci, ale tuto aplikaci může klidně využít doma, či i v internetové kavárně.

3.2. RIA Technologie

Při vytváření RIA aplikací máme možnost výběru technologie. Není pouze jediná správná technologie, která by se dala použít, ale je jich hned několik. Pojdme se tedy seznámit s tím, na které technologie se zaměříme.

V této diplomové práci se zaměřím hlavně na lídry na pomyslném trhu RIA technologií. Zúžíme tedy tuto skupinu pouze na technologie, které jsou nejčastěji používané pro vytváření RIA aplikací. Do tohoto výběru nebudeme brát také v potaz technologie sloužící pro vytváření RIA aplikací pro mobilní telefony.

Některé RIA technologie potřebují mít pro správný běh nainstalovaný zásuvný modul¹⁵. Zásuvným modulem budeme rozumět běhové prostředí pro konkrétní RIA aplikaci. Podle přístupu k zásuvným modulům si RIA technologie rozdělíme do dvou skupin. První skupina bude obsahovat technologie, které pro svůj běh nepotřebují mít nainstalovaný zásuvný modul. Druhou skupinou budou ty technologie, které zásuvné moduly potřebují. Do první skupiny budeme řadit pouze technologii AJAX, což je v této chvíli jediná implementace RIA, jež by teoreticky měla běžet v každém moderním prohlížeči. V tom má technologie AJAX nespornou výhodu. Tyto aplikace jsou ihned uživatelům přístupné. Na druhou stranu však tato technologie nemá tak obrovské možnosti, které se nabízí technologiím z druhé skupiny. Do druhé skupiny budeme řadit technologie Adobe Flash, Adobe Flex, JavaFx a Microsoft Silverlight. Všechny tyto technologie potřebují mít nainstalovaný zásuvný modul, což může některým uživatelům připadat omezující a je to často jediný důvod, proč danou technologii nepoužívají. Někteří uživatelé například nemají oprávnění si doinstalovat tyto zásuvné moduly, a proto se jim danou aplikaci nepodaří nikdy otevřít. Proto pokud vytváříme nějakou aplikaci, měli bychom mít na paměti hlavně to, pro jaké uživatele tuto aplikaci vytváříme.

3.3. Multimédia

K tomu abychom mohli do analýzy zařadit pohled z hlediska podpory multimédií, musíme si nejdříve nadefinovat, co budeme mezi multimédia řadit. Multimediálním obsahem budeme rozumět vizuální a audio prostředky. Mezi vizuální prostředky budeme řadit rastrovou grafiku, vektorovou grafiku, 3D grafiku (geometrické modely a objekty) a animace. Mezi audio prostředky zařadíme jakékoliv zvukové stopy. Společným vizuálním a audio prostředkem jsou videa, která v multimédiích nesmíme opomenout.[1]

Počítače ukládají a zpracovávají obrazové informace pomocí dvou základních způsobů. Prvním způsobem je použití rastrové grafiky a druhým způsobem je použití vektorové grafiky.

V rastrové neboli bitmapové grafice je celý obraz tvořen pomocí jednotlivých bodů (pixelů), které jsou uspořádány do mřížky. Každý bod má přesně určenou polohu a barvu. Každý rastrový obraz je tedy definován svou velikostí (šířkou a

¹⁵Zásuvný modul, jinak nazývaný plug-in.

výškou), rozlišením (hustotou barevných bodů) a barevnou hloubkou (počet možných barev, kterých může každý bod nabývat). Bitmapová grafika se vyznačuje vysokou paměťovou náročností, proto existují kompresní metody pro rastrové obrazy, které umožňují snížit tuto paměťovou náročnost. Nejznámějšími formáty na webu, které jsou podporovány v nejrozšířenějších webových prohlížečích, jsou PNG¹⁶, GIF¹⁷ a JPEG¹⁸.

Vektorová grafika používá k popisu obrazu základní geometrické útvary, jako jsou: body, přímky, mnohoúhelníky a křivky. Jednou z obrovských výhod použití vektorové grafiky je možnost zmenšení nebo zvětšení obrazu bez ztráty kvality. S podporou vektorové grafiky na webu je situace mírně komplikovaná. Formátem, který by měl být v dohledné době podporovaný všemi nejrozšířenějšími prohlížeči je SVG¹⁹. Tento formát je založený na XML a na jeho vývoji pracuje konsorcium W3C. Společnost Microsoft se v minulosti pokusila vyvinout svůj vlastní formát VML (Vektor Markup Language), který je však podporovaný pouze ve webových prohlížečích Internet Explorer od verze 5 a proto se tento formát nestal příliš rozšířeným. RIA technologie většinou používají své vlastní formáty pro vektorovou grafiku.

Díky výše zmíněným způsobům můžeme nadefinovat, co budeme označovat pod pojmem animace. Animaci můžeme chápat jako pohybuující se obraz. Tento pohybuující se obraz se skládá z jednotlivých obrazů (snímků), rychle se měnících za sebou, díky kterým nastává zdánlivá iluze pohybu.

Audio a video asi není potřeba představovat, přesto se u nich může objevit pár nepřesných pojmů, o kterých bych se ráda v rychlosti zmínila. U videa/audia rozlišujeme formáty a kodeky. Formát videa/audia si můžeme představit jako kontejner pro data a informace popisující tyto data. Data v kontejneru mohou být jak soubory, tak datový tok. U videa se data mohou skládat například z titulků, několika zvukových stop pro jednotlivé jazyky a samotného videa. Informace popisující data jsou uloženy v hlavičce, příkladem těchto informací může být například název kodeku, nebo rozlišení v případě videa, atp. U jednotlivých formátů se však kontejner liší. Protože samotná data jsou paměťově náročná, používají se na jejich komprimaci a dekomprimaci právě zmíněné kodeky (kodér-dekodér). Příkladem video kodeků je např.: H.264, VP8 nebo Theora. Z audio kodeků je to např.: Ogg, MP3, Wave. Může se stát, že kodek a formát mohou mít stejný název, což může být matoucí. [2]

Multimediální obsah je v dnešní době na Internetu velice rozšířený. Jedním příkladem za všechny může být již jednou zmíněný YouTube, portál pro sdílení videí. Multimediální obsah je dnes často využíván například pro vzdělávání, na pracovních poradách, prezentacích nových produktů, atp.

¹⁶PNG - z anglického Portable Network Graphics.

¹⁷GIF - z anglického Graphics Interchange Format.

¹⁸JPEG - z anglického Joint Photographic Experts Group.

¹⁹SVG - z anglického Scalable Vector Graphics.

3.4. Metoda analýzy

Analýzu jednotlivých technologií budeme provádět následujícím způsobem. Každá technologie bude rozepsána pomocí těchto následujících bodů:

- Úvod – první seznámení s technologií a její historický vývoj.
- Princip – jaké vývojové prostředí je potřebné, na jakém principu daná technologie pracuje, výhody použití této technologie pro vývojáře.
- Podpora v multimédiích – jaké prostředky pro tvorbu multimediálního obsahu jsou v dané technologii nabízeny.
- Příklad – reálné příklady, kde je tato technologie využita.

3.5. Adobe Flash

3.5.1. Úvod

Určitě se každý z nás již někdy v životě setkal s reklamním bannerem nebo hrou vytvořenou pomocí Adobe Flash (dále Flash). Většina z nás ho má dodnes spojený s nejrůznějšími animacemi a reklamními bannery. Z Flashe se však postupem času vyvinula silná platforma, která umožňuje vytvářet nejrůznější interaktivní aplikace (výukové programy), kreslené filmy a v neposlední řadě také bohaté webové aplikace. Flash spadá do druhé skupiny technologií, tedy k těm které pro správný běh aplikací potřebuje mít v prohlížeči nainstalovaný zásuvný modul tzv. Flash Player. Momentálně tuto platformu vlastní společnost Adobe.

Pojďme se podívat na historické mezníky této platformy. Za zakladatele této platformy můžeme považovat Jonathana Gaya. Již od útlého věku se zajímal možnostmi, jak na počítačích vytvářet grafické objekty. Pomocí Pascalu vytvořil svůj první grafický editor. V roce 1993 založil společně s Charlie Jacksonem firmu nazvanou FutureWave Software, jejímž produktem byl grafický editor SmartSketch. Tento editor byl vyvíjen pro operační systém Go. Vývoj operačního systému GO byl však v roce 1994 zastaven a tím se trh pro SmartSketch uzavřel. V té době nastává rozšiřování Internetu a webových stránek. Díky odezvě uživatelů na produkt SmartSketch, se vlastníci firmy rozhodli vytvořit software sloužící k vytváření animací pro webové stránky. V roce 1996 tedy vznikl nástupce SmartSketch nazvaný FutureSplash Animator. Ve stejném roce dostala firma dvě obrovské zakázky pro společnosti Microsoft a Disney. V prosinci 1996 společnost Macromedia koupila firmu FutureWave za 500 tisíc dolarů a FutureSplash Animator byl přejmenovaný na Macromedia Flash 1.0.

Již v červnu 1997 vychází Macromedia Flash 2.0. Tato verze obsahovala knihovny objektů, podporu stereo zvuku a barevné transformace. Až do roku

2000 vydává každý rok firma Macromedia novou verzí Flashe. Postupně vylepšují výkon, přidávají streamování MP3, pohybové transformace, skriptovací jazyk ActionScript, atd. . . 4. března 2002 je v San Francisku uvedena nová verze Macromedia Flash pod názvem Flash MX. Tato verze je významná přidáním podpory videa a UI komponent. O dva roky později vychází verze Macromedia Flash MX 2004, spolu s ní byl představen Action Script 2.0 a Flash Lite, který umožňuje rozšíření Flashe na mobilní zařízení. Tato verze vychází ve dvou typech: základní a profesionální. Profesionální se od základní liší nástroji pro řízení datových toků, nástroji pro podporu týmové produktivity a také například šablonami pro tvorbu aplikací (Flash Slide Presentation a Flash Form application) [3]. V roce 2005 vychází Flash Macromedia 8. Mezi největší novinky této verze patří například speciální grafické efekty, velmi dobrý kodek pro komprimaci videa, podpora alfa kanálu u videa, lepší zobrazení textu na obrazovce a zlepšení podpory pro mobilní zařízení. Ve stejném roce společnost Adobe Systems Inc. kupuje společnost Macromedia za 3,4 miliardy dolarů [10]. Macromedia Flash je přejmenován na Adobe Flash. Adobe za dobu, kterou vlastní Flash, stihl do dnešní doby vydat tři nové verze. Aktuální verzí je Adobe Flash Professional CS5. Dnes je Flash světovým standardem pro tvorbu interaktivních multimediálních aplikací pro web. Zásuvný modul (Flash Player) pro běh aplikací vytvořených ve Flashi je rozšířen na 99% všech počítačů připojených k Internetu.

Rok	Verze
1993	FutureWave Software
1996	Macromedia Flash 1.0 (dříve FutureSplash Animator)
1997	Macromedia Flash 2.0
1998	Macromedia Flash 3.0
1999	Macromedia Flash 4.0
2000	Macromedia Flash 5.0
2002	Macromedia Flash MX
2004	Macromedia Flash MX 2004
2005	Macromedia Flash 8
2007	Adobe Flash CS3
2008	Adobe Flash CS4
2010	Adobe Flash CS5

Tabulka 1. Historický přehled vývoje Adobe Flash

3.5.2. Popis

Jak již bylo zmíněno, tato technologie patří do druhé skupiny, k správnému běhu aplikací potřebuje mít v prohlížeči nainstalován zásuvný modul. Tento zásuvný modul se nazývá Flash Player. Je to vlastně běhové prostředí pro aplikace vytvořené ve Flashi. Při každém vydání nové verze Flashe byl také vydán nový Flash Player. Momentálně je tedy verze Flash Player 10. Tento zásuvný modul má velikost 2,7MB pro Windows, 5,5MB pro Mac OS a 4,7MB pro Linux.

Adobe Flash CS5 je nástroj pro vytváření interaktivního obsahu, který je poté zobrazen ve Flash Playeru. Není určen přímo pro vývojáře, ale spíše je orientován na designéry. Základem práce s Flashem je tzv. časová osa. Časová osa udává, co se ve kterém okamžiku má zobrazit. Soubory s grafikou, texty a multimédií jsou uloženy v souborech s koncovkou .fla. Od verze Adobe Flash CS5 mají vytvořené soubory FLA interní formát XFL²⁰. Tento nový formát je použitý i v jiných nástrojích od Adobe (např. InDesign). Je tedy možné začít práci v aplikaci InDesign a poté ji dokončit v Adobe Flash CS5.

Od verze Macromedia Flash 5 je integrován nový skriptovací jazyk ActionScript. Dnes je již ve třetí verzi a vyvinul se v plnohodnotný objektově orientovaný jazyk. Díky tomuto jazyku můžeme nadefinovat chování aplikace. Aplikace tedy může správně reagovat na interakci uživatele, ne jen zobrazovat předem dané věci.

Pro publikování aplikace musíme vytvořit výstupní soubor ve formátu SWF²¹. Soubor SWF jsou vlastně zkompileované verze souborů FLA. Informace v tomto souboru jsou značně komprimované, aby měl soubor co nejmenší možnou velikost. Formát SWF představuje otevřený standart podporovaný i jinými aplikacemi.

3.5.3. Podpora v multimédiích

Podpora grafiky:

- Rastrová – Podporovanými formáty pro rastrovou grafiku jsou: BMP, GIF, JPG, PNG, TIFF²², FXG²³.
- Vektorová – Ve Flashi lze využít nástrojů Rectangle, Oval, Polystar Tool a Pen k vytvoření vektorové grafiky. Výhodou je možnost převedení rastrových obrázků do vektorových.

²⁰XFL z anglického Extensible Flash.

²¹SWF z anglického ShockWave Flash.

²²TIFF je zkratkou pro Tag Image File Forma je jeden z formátů pro ukládání rastrové počítačové grafiky

²³FXG je anglickou zkratkou pro Flash XML Graphics. Tento formát byl vytvořen vytvořen firmou Adobe Systems Inc.

- 3D – Flash podporuje tzv. perspektivní 3D grafiku, které lze dosáhnout pomocí transformací (natočení, posunutí). Všechny 3D transformace se odehrávají ve vztahu ke třem osám ozn. X, Y a Z.
- Animace – Animace jsou ve Flashi založené na sekvenčním procházení snímků časové osy (tzv. animace frame-to-frame). Novinkou ve Flashi je nástroj Bone tool, který umožňuje proložit objekt kostrou. Díky této kostře lze vytvořit velice reálné animace.

Přehrávání audia – Zvuky se ve Flashových aplikacích dají rozdělit na dva typy: zvuky pro události a zvukové streamy. Zvuky pro události se musí stáhnout celé před jejich přehráním. Zvukové streamy se začínají přehrávat, jakmile je stažen dostatek dat. Podporovanými formáty zvukových souborů jsou: ASND (Windows nebo Macintosh), WAV (pouze Windows), AIFF (pouze Macintosh), MP3 (Windows nebo Macintosh).

Přehrávání videa – Video lze ve Flashové aplikaci používat třemi způsoby. První způsob se hodí pro videa, která mají velmi malou velikost. Video je přímo vloženo do souboru SWF. Další dva způsoby již video nekládají do souboru SWF, ale jsou uloženy na serveru. Druhým způsobem je mít video uložené na webovém serveru, z kterého je postupně stahováno. Posledním způsobem je mít video uloženo na Flash Media Serveru, který zajišťuje plynulé přehrávání streamovaného videa. Flash podporuje tři typy kodeků: Sorenson Spark, ON2 VP6 a H.264. Nejnovějším je H.264, který byl do přehrávače Flash Playeru přidán ve verzi Flash CS4 Professional. Formát F4V používající tento kodek nabízí výrazně lepší poměr kvality a datového toku, než předchozí zmíněné kodeky. Pro ovládání videa lze použít komponentu FLVPlayback. Samozřejmě lze vytvořit i vlastní přehrávač pomocí ActionScriptu.

3.5.4. Výhody a nevýhody

Výhody

- Vyzrálá technologie, vyvíjí se již 15 let.
- Rozšířenost – na konci roku 2010 byl Flash Player rozšířen na 99% počítačů připojených k internetu.
- Plná podpora multimédií.
- Multiplatformní.

Nevýhody

- Podpora SEO²⁴ - je podporovaný pouze v některých vyhledávačích (Google, Yahoo, atd...).

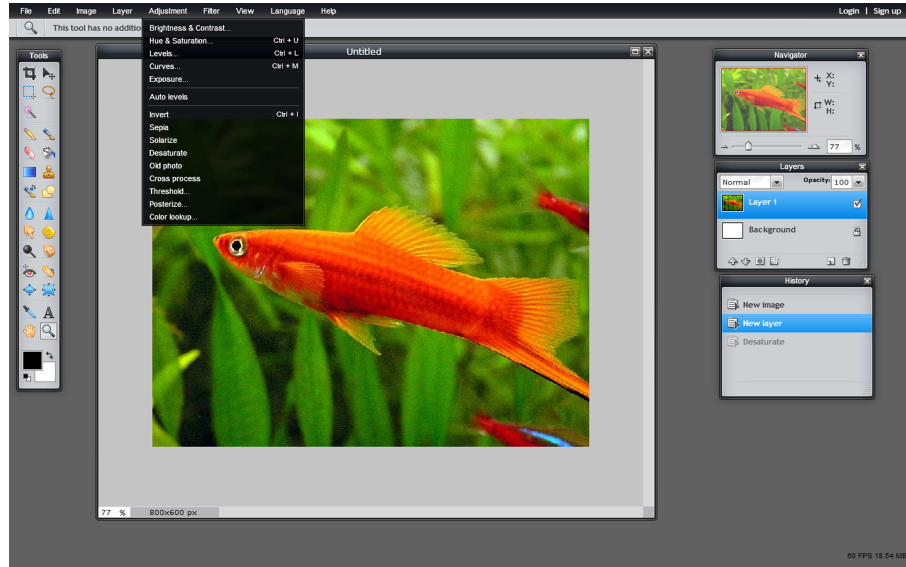
²⁴SEO z anglického Search Engine Optimization, neboli optimalizace pro vyhledávače.

- Není vhodný pro vývoj rozsáhlých webových aplikací.

3.5.5. Příklad

Příkladů webových aplikací v této technologii můžeme nalézt opravdu mnoho. Vybrala jsem grafický editor, který můžete nalézt na adrese: <http://www.pixlr.com/editor/>

Musím říct, že jsem byla velice příjemně překvapená, kolik možností tento editor nabízí. Připadalo mi až neskutečné, že věci, které jsem dělala v Adobe Photoshopu lze udělat i pomocí webové aplikace.



Obrázek 1. Aplikace Pixlr

3.6. Adobe Flex

3.6.1. Úvod

Vlastníkem technologie Adobe Flex(dále jen Flex) je společnost Adobe Systems Inc. Tato technologie je určena primárně k vývoji bohatých webových aplikací. V době svého vzniku byla již zavedenou technologií na trhu Macromedia Flash (nyní Adobe Flash). Flash byl však primárně určen pro designéry a není stavěný pro vývoj rozsáhlých internetových aplikací. Technologii Flex řadíme k technologiím z druhé skupiny. Pro běh aplikací vytvořených pomocí Flexu je zapotřebí mít nainstalován zásuvný modul Flash Player, stejně jako tomu je u technologie Adobe Flash. Flex je zařazen mezi produkty společnosti Adobe, které na vývoji aplikací mohou spolupracovat, tento balík nástrojů se nazývá Adobe Creation Suite.

Historie technologie Flex sahá až do roku 2003. V té době společnost Macromedia pracuje na projektu označeném kódovým jménem Royale [5]. V březnu 2004 je uvolněna první verze tohoto frameworku pod oficiálním názvem Macromedia Flex. Flex 1.0 byl představen jako nová cesta pro vytváření bohatých internetových aplikací. Aplikace vytvořené ve Flexu běžely v jakémkoli prohlížeči, v kterém byl nainstalován zásuvný modul Flash Player 7. Používá dva jazyky MXML²⁵, který slouží k popisu prezentačního rozhraní aplikace a ActionScript 2.0, který se používá hlavně k definici chování aplikace. První verze byla vyvíjena jako J2EE aplikace, která je schopná zkompileovat MXML a ActionScript jako Flashovou aplikaci (výsledkem je vytvoření souboru SWF). Flex je v době své první verze placeným frameworkem, kde licence pro jeden počítač stojí 12000 dolarů. V listopadu téhož roku vychází verze Flex 1.5, která sebou nese novinky, jako jsou komponenty pro vizualizaci dat, větší možnosti pro stylování a skinování a zlepšení výkonnosti.

V červnu 2006 byla uvolněna verze Flex 2.0. V té době již není vlastníkem společnost Macromedia, ale Adobe Systems Inc. Tato verze byla ve vývoji Flexu velmi významná ze dvou důvodů. Prvním důvodem je změna licenčních podmínek, protože Flex se stává open-source projektem. Zmíněný kompilátor a zdrojové kódy pro stovky uživatelsky bohatých ovládacích prvků jsou v této verzi nabízeny zdarma. Uživatel mohl aplikaci vyvíjet v libovolném zvoleném editoru a poté spustit kompilaci z příkazové řádky a to vše zcela zdarma. Druhým důvodem byla integrace ActionScript 3.0., který se stal plnohodnotným objektově orientovaným jazykem [6].

Rok	Verze
2004	Macromedia Flex 1.0
2006	Adobe Flex 2.0
2008	Adobe Flex 3.0
2010	Adobe Flex 4.0

Tabulka 2. Historický přehled vývoje Adobe Flex

3.6.2. Popis

Jak již bylo zmíněno v úvodní části, Flex patří do skupiny technologií, které pro správný běh aplikací potřebují mít nainstalovaný zásuvný modul. Pomocí tohoto frameworku lze vyvíjet, jak webové, tak desktopové aplikace. V případě, že je aplikace spuštěna ve webovém prohlížeči, je běhovým prostředím Flash

²⁵MXML je zkratkou pro Macromedia XML.

Player. V případě, že je aplikace nainstalována v počítači je běhovým prostředím Adobe AIR²⁶ Desktop.

Flex je vyvíjen jako Open Source. Vše, co vývojář potřebuje k vývoji, je obsaženo v Flex SDK²⁷. Součástí Flex SDK je například: sada tříd a komponent, debugger, kompilátory, atd... [7] Vývojář má tedy možnost využít k vývoji Flex SDK s vybraným editorem, nebo využít placeného nástroje Adobe Flash Builder (dříve Adobe Flex Builder), který vychází ve dvou verzích Standart a Premium. Tyto nástroje mají hned několik výhod: není potřeba kompilovat aplikaci pomocí příkazové řádky, obsahuje okno s náhledem pro design aplikace, šablony aplikací, atd...

Jak již víme, při vývoji se využívá dvou jazyků MXML a ActionScript. MXML je značkovací jazyk pro uživatelské rozhraní. Je založen na XML²⁸, a proto je zdrojový kód vždy validním XML. Při vytváření uživatelského rozhraní využíváme předefinovaných komponent (Button, Label, TextInput, atd...). Vzhled jednotlivých komponent v jazyce MXML lze ovlivnit pomocí atributů, nebo pomocí externích stylů, které jsou velice podobné CSS²⁹. Druhým zmíněným jazykem byl ActionScript, který je z části velice podobný JavaScriptu. Tato podobnost nebyla zvolena náhodně, pro vývojáře se totiž jedná o model vývoje, který důkladně znají. ActionScript má všechny rysy objektově orientovaného jazyka. Vše, co je napsané pomocí MXML, lze napsat pomocí ActionScriptu. Mohli bychom říct, že jazyk MXML je pouze syntaktickou podobou jazyka ActionScript. Této skutečnosti je využíváno při kompilaci.

Při kompilaci je celá aplikace zkompileována do jednoho souboru SWF. Flex obsahuje dva kompilátory mxmclc - kompilátor aplikace a compc - kompilátor komponent. Pomocí kompilátoru compc kompilujeme zdroje, které naše aplikace využívá. Například můžeme zkompileovat komponenty a třídy do SWC souborů nebo do RSL³⁰ a poté staticky nebo dynamicky přilinkovat tyto knihovny k aplikaci. Kompilátor mxmclc se používá ke kompilaci MXML, ActionScriptu, SWC a RSL souborů, které jsou zkompileovány do výsledného souboru SWF, který můžeme publikovat na webovém serveru. V případě, že ke kompilaci používáme Flash Builder vše probíhá automaticky [8].

Protože Flex je pouze klientskou technologií, musíme při jeho použití pro získávání dat ze serveru využít také serverové technologie, jako je například PHP, ASP.NET nebo Java. Pro komunikaci se serverem můžeme využít objektů HTTP-Service, Webservice nebo RemoteObject. Máme možnost využít Adobe BlazeDS nebo Adobe LiveCycle Data Services, pro zjednodušení komunikace se serverem.

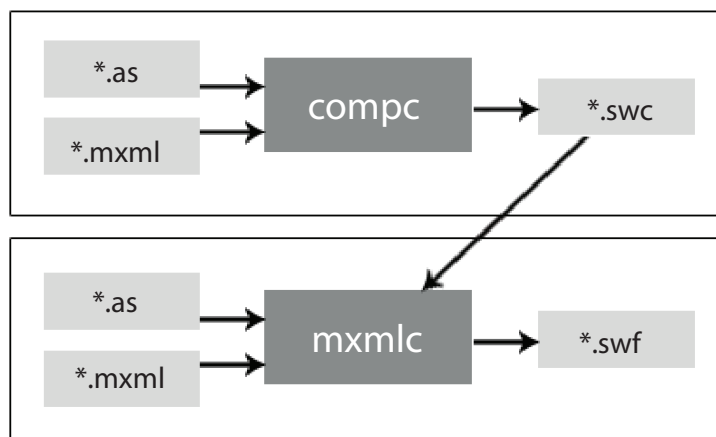
²⁶Adobe AIR je zkratkou pro Adobe Integrated Runtime.

²⁷Flex SDK je zkratkou pro Flex Software Development Kit.

²⁸XML - z anglického Extensible Markup Language je obecný značkovací jazyk, který byl vyvinut a standardizován konsorciem W3C.

²⁹CSS - z anglického Cascading Style Sheets neboli kaskádové styly jsou jazyk pro popis způsobu zobrazení stránek.

³⁰RSL je zkratkou pro Runtime Shared Libraries.



Obrázek 2. Průběh kompilace

3.6.3. Podpora v multimédiích

Mezi Flexem ve verzi 3 a ve verzi 4 nastala výrazná změna. Ve Flexu 1-3 se používaly pro grafiku Halo komponenty. Ve Flexu 4 máme na výběr, můžeme použít buď Halo komponenty, nebo nové Spark komponenty, které jsou navrženy tak, aby byly rychlejší (jsou založené na jiném konceptu vykreslování) a lépe se s nimi pracovalo. Všeobecně se doporučuje používat Spark komponenty.

Podpora grafiky:

- Rastrová – Flex podporuje následující formáty obrázků GIF, JPEG, PNG, FXG³¹. Pro zobrazení obrázku se používá Spark komponenta BitmapImage z balíčku spark.primitives, u kterého se zadává například atribut Source (umístění obrázku).
- Vektorová – Flex má podporu vektorové grafiky. V balíčku spark.primitives najdeme komponenty pro vektorovou grafiku, např.: Ellipse (kruh, elipsa), Line (linka), Path (cesta), Rect (obdélník, čtverec), atd...
- 3D – Flex má podporu perspektivní 3D grafiky. Obsahuje 3D transformace jako je Move3D, Rotate3D, Scale3D, atd... Tyto transformace jsou obsaženy v balíčku spark.effects.
- Animace – Balíček spark.effects obsahuje objekty pro tvorbu animací.

Přehrávání audia - Flash player podporuje následující kodeky: ADPCM, MP3, HE-AAC.

Přehrávání videa – Pro přehrávání videa obsahuje Flex komponentu VideoDisplay, která je opět obsažena mezi Spark komponentami. Tato komponenta

³¹FXG je zkratkou pro Flash XML Graphic.

obsahuje např. atributy: Source(umístění videa), autoPlay(automatické spuštění přehrávání), atd... Flash player podporuje následující kodeky: Sorenson Spark, On2 VP6, H.264. A následující formáty: FLV, F4V, MPEG-4.

3.6.4. Výhody a nevýhody

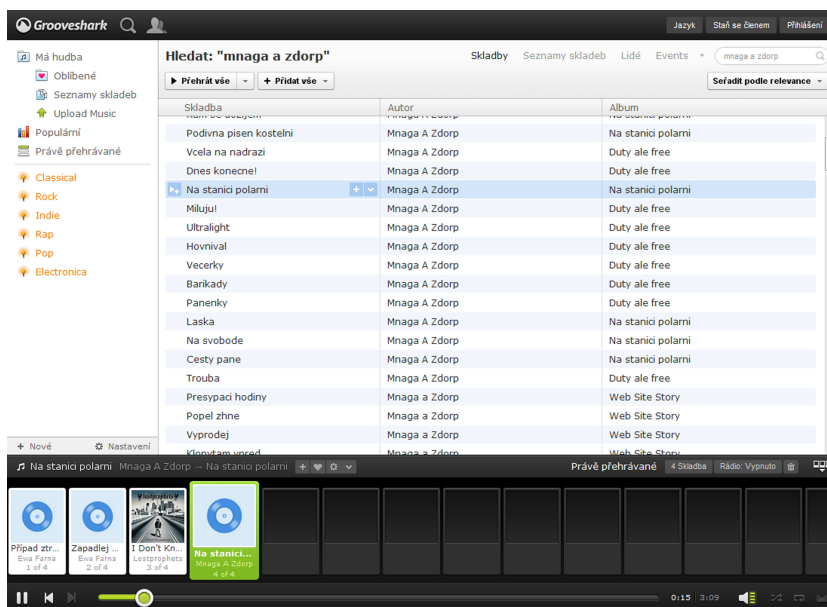
- Rozšířenost – na konci roku 2010 byl Flash Player rozšířen na 99% počítačů připojených k internetu.
- Plná podpora multimédií.
- Multiplatformní.

Nevýhody

- Podpora SEO - je podporovaný pouze v některých vyhledávačích (Google, Yahoo, atd...).

3.6.5. Příklad

Jako příklad reálné aplikace jsem vybrala webovou stránku, která umožňuje přehrávání nejrůznějších audio písní a poslech rádií. Uživateli je nabídnuta široká škála možností od vyhledávání písní podle jména skladatelů, ukládání písní do oblíbených, či zakoupení vybraných písní, atd... Aplikaci lze nalézt na adrese: <http://listen.grooveshark.com/>



Obrázek 3. Aplikace Grooveshark

3.7. Ajax

3.7.1. Úvod

Pojem AJAX (Asynchronous JavaScript and XML) není konkrétní technologií. Spíše by se dalo říct, že je to skupina technologií, které jsou použity dohromady za určitým cílem. Cílem je vytváření interaktivní webové aplikace. AJAX patří do první skupiny technologií, kde není potřeba žádný nainstalovaný zásuvný modul, což je jeho nesporná výhoda oproti ostatním technologiím.

Za počátky AJAXu lze považovat první pokusy použití asynchronního volání. V roce 1996 představuje společnost Microsoft nový prohlížeč Internet Explorer 3.0. Tento prohlížeč obsahuje element IFrame. IFrame byl vlastně vložený rámec ve stránce. V tomto rámci mohla být zobrazena další stránka ve stránce na straně klienta. Nevýhodou bylo, že tento element obsahoval pouze prohlížeč Internet Explorer 3.0. O rok později tedy v roce 1997 společnost Netscape sice vydává nový prohlížeč Netscape Navigator 4.0, který obsahuje nový element se stejnými vlastnostmi jako má IFrame, bohužel mu však dává název Layer. Pro vývojáře to znamenalo další práci navíc, protože svůj kód museli upravovat pro každý prohlížeč zvlášť.

Vývoj pokračuje a v roce 1998 představuje Microsoft novou technologii nazvanou Remote Scripting. Tato technologie byla založena na běhu java appletu na klientském počítači, který dokázal na pozadí komunikovat s webovým serverem, kde vstupy a výstupy byly zpracovávány pomocí JavaScriptových funkcí. Tato technologie funguje v prohlížeči Internet Explorer 4.0 a později i v Netscape Navigator 4.0.

V roce 1999 vychází nový prohlížeč Internet Explorer 5.0, ve kterém je implementován ActiveX Object s MSXML.XMLHTTP, dnes ho známe pod názvem XMLHttpRequest. Dále tento objekt budeme nazývat XMLHttpRequest. Tento objekt umožňoval komunikaci mezi webovým serverem a klientem pomocí protokolu HTTP. Objekt XMLHttpRequest umožňoval přenášet data pomocí formátu XML, výsledky byly zpracovány pomocí JavaScriptu. Funkcionalita tohoto objektu byla zajištěna díky objektu ActiveX, který podporoval pouze prohlížeč Internet Explorer. Postupem času se objekt XMLHttpRequest dostává i do ostatních prohlížečů.

První opravdová zmínka o AJAXu se objevuje až v dubnu 2005 v článku od Jesse James Garretta pod názvem AJAX: New Approach to Web Applications (AJAX: Nový přístup k webovým aplikacím). Hovoří zde o rozdílnosti desktopových a webových aplikací. Tento rozdíl má zaplnit právě AJAX, díky nové možnosti asynchronní komunikace klienta a serveru. V článku také poznamenává, že AJAX není novou technologií, ale že jde o balíček několika technologií, které díky vzájemnému propojení vytváří velice silný nástroj pro tvorbu interaktivních webových aplikací.

3.7.2. Popis

Jak bylo řečeno výše, nejedná se pouze o jednu technologii, ale o skupinu technologií. Jsou to technologie HTML (nebo XHTML), CSS, DOM, JavaScript a XMLHttpRequest. Základním stavebním prvkem je XMLHttpRequest. Tento objekt bohužel prozatím nebyl standardizován, ale konsorcium W3C ho již má zařazený mezi novými kandidáty na standardizaci. Hlavním rozdílem mezi prohlížeči bylo, že Internet Explorer měl funkcionalitu tohoto objektu implementovanou jako ActiveX objekt. Ostatní prohlížeče měly tento objekt implementovaný, jako vestavěný objekt v JavaScriptu pod názvem XMLHttpRequest. S příchodem Internet Exploreru 7 je použití objektu XMLHttpRequest ve všech majoritních prohlížečích stejné.

Vytváření požadavku probíhá v následujících krocích:

1. Nejprve se vytvoří instance objektu XMLHttpRequest.
2. Pro obsluhu změny stavu asynchronní odpovědi ze serveru se nastaví vlastnost `onreadystatechange`. Tato vlastnost objektu umožňuje specifikovat funkci, která se zavolá při každé změně stavu požadavku (aktuální stav lze získat pomocí atributu `readyState`).
3. Poté se vytváří požadavek pomocí metody `open`, která bere tři parametry – metodu komunikace (GET, POST, HEAD a další), adresu URL, na kterou má být požadavek odeslán a posledním parametrem je typ požadavku asynchronní nebo synchronní, tento parametr může nabývat hodnot `true` nebo `false`, někdy se také zadává jméno a heslo, které slouží k autentizaci na serveru.
4. Odeslání samotného požadavku se provádí pomocí metody `send`. Metoda bere jeden parametr. V případě, že byla vybrána metoda komunikace GET, je tento parametr `null`. V případě, že byla metoda komunikace POST, obsahuje tento parametr tělo HTTP požadavku.
5. Požadavek je odeslán na server.

Po odeslání je při každé změně stavu požadavku volána obsluha události `onreadystatechange`. Podle aktuálních vlastností požadavku je možné zjistit, v jakém stavu se právě posílání nachází a případně jaký je výsledný HTTP kód serveru. Ve chvíli, kdy je vlastnost `readyState` rovna hodnotě 4, je možné začít odpověď zpracovávat.

Odpověď může být zaslána v následujících formátech:

- XML – Jak písmeno X ve zkratce AJAX značí, v původním návrhu bylo použití formátu XML pro odesílání a přijímání odpovědi. Tato odpověď se následně na klientovi musela parsovat, což v mnoha případech bylo velice

pracné a zdlouhavé. Důvodem proč padla volba v původním návrhu na XML bylo díky univerzálnosti tohoto značkovacího jazyka a snaze o rozšíření transformace na klientovi pomocí XSLT³².

- JSON³³ – Je to formát pro výměnu dat, který se stal na webu velice oblíbeným. Tento formát dat je odvozený z JavaScriptu, takže není potřeba složitěho parsování odpovědi, jak tomu je u formátu XML. Data jsou prezentovány jako objekty v JavaScriptu, takže je možné je použít s minimální režii ihned.
- HTML – Jedním ze způsobů je odesílání a přijímání odpovědi ve formátu HTML. Tento způsob má určitou výhodu, pomocí jediného přiřazení vlastnosti DOM objektu můžeme překreslit část stránky a nemusíme se zabývat žádným parsováním nebo vytvářením DOM objektu. Toto přiřazení je uskutečněno pomocí použití vlastnosti innerHTML elementu, který však není standardizován a jeho použití nemusí ve všech prohlížečích fungovat. Můžeme však využít JQuery framework, který nám také dovoluje přiřadit elementu řetězec, který obsahuje HTML elementy.
- Text – Samozřejmě se dá odpověď zaslat, jako obyčejný jednoduchý textový řetězec. V případě, že očekáváme krátké a jednoduché odpovědi ze serveru, je tento způsob dostačující, pokud však očekáváme složitější odpovědi, je zapotřebí naprogramovat parser těchto odpovědí, což může být dost pracné.

Existuje celá řada Ajax Frameworků, které se snaží práci vývojářům ulehčit. Zmínila bych zde několik nejznámějších, se kterými se v dnešní době můžeme setkat.

JQuery – v dnešní době asi nejznámější JavaScriptový Framework. Byl vydán Johnem Resigem v lednu 2006. Umožňuje jednoduchou práci s DOM elementy (modifikaci, vytváření, atd...), obsahuje podporu AJAXu, manipulaci s CSS a mnoho dalšího. Jednou z částí tohoto frameworku je knihovna JQuery UI, která slouží pro implementaci pokročilejších efektů a animací. JQuery klade důraz na jednoduchost, rychlost a přehlednost a navíc je tento Framework dostupný zdarma.

GWT³⁴ – tento framework je takzvaným řešením ”všechno v jednom”³⁵. Framework obsahuje vše, co by mohl vývojář ke svému vývoji webové aplikace potřebovat. Výhodou je, že vývojář může psát v Javě, jak klientskou, tak i serverovou

³²XSLT z anglického eXtensible Stylesheet Language Transformations slouží k převodům zdrojových dat ve formátu XML do libovolného jiného požadovaného formátu, nejčastěji HTML.

³³JSON z anglického JavaScript Object Notation.

³⁴GWT z anglického Google Web Toolkit.

³⁵Framework GWT je tzv. all-in-one.

část. Klientská část je tvořena ze znovupoužitelných UI komponent (renderovaných jako HTML) a logiky, která je pomocí Java kompilátoru přeložena do JavaScriptu.

3.7.3. Podpora v multimédiích

Z hlediska multimédií má Ajax omezené možnosti. Jeho možnosti jsou dány technologiemi, ze kterých se skládá. Abychom tyto možnosti rozšířili, vezmeme v úvahu chystané verze HTML 5.0 a CSS3.

Podpora grafiky:

- Rastrová – pomocí tagu `` můžeme do HTML stránky vložit obrázek. Většina prohlížečů podporuje obrázky ve formátech GIF, JPG, PNG. Novinkou v HTML 5 je element `<Canvas>`, který umožňuje dynamicky vkládat rastrovou grafiku. Využití tohoto elementu může být vhodné například při vytváření dynamických grafů podle vstupů od uživatele.
- Vektorová – vektorovou grafiku lze do stránky vkládat pomocí formátu SVG, který je založený na XML. Slouží k popisu dvojrozměrné vektorové grafiky. Základními elementy jsou `svg` (základní tvar), `rectangle` (obdélník), `circle` (kruh), `ellipse` (elipsa), `line` (jednoduchá čára), `polyline` (lomená čára), `polygon` (mnohoúhelník) a `text`. Je vyvíjen pod záštitou konsorcia W3C ve spolupráci s firmami jako jsou Apple, IBM, Microsoft, atd... Nevýhodou je, že ne všechny prohlížeče formát SVG v tuto chvíli podporují. Některé k správnému zobrazení potřebují mít nainstalovaný zásuvný modul (např. Internet Explorer 8) od firmy Adobe.
- 3D – částečná podpora perspektivního 3D se bude vyskytovat v HTML5 s využitím elementu `<canvas>`. Na adrese <http://ajaxian.com> je možné vidět některé příklady pokusů zabývajících se 3D grafikou. Jedním příkladem je vytvořený 3D grafický editor. Pomocí jednoduchých krychliček je možné nakreslit 3D tvar a potom se na tento element dívat z různých stran (adresa <http://www.3dtin.com/>).
- Animace – Určitý typ animací je možné použít již dnes. Velkým posunem v tomto ohledu byl framework JQuery, který nabízí uživateli přidat různé animační efekty na HTML elementy. Pomocí JQuery Frameworku však není možné dělat žádné složité animace, je limitován technologiemi, které používá. Samozřejmě různé animace lze naimplementovat pomocí JavaScriptu a CSS.

Přehrávání audia momentálně v HTML 4.0 nelze spustit bez nainstalování zásuvných modulů od jiných technologií. HTML 5.0 nabízí pro tuto činnost element `<audio>`, který obsahuje atributy `src` (pro zadání cesty k souboru), `loop`

(znovu přehrávání), autoplay (automatické spuštění přehrávání), atd... Problém dnešních dnů jsou hlavně samotné audio formáty, ne všechny prohlížeče podporují stejné audio kodeky. Částečnou podporu mají kodeky: Ogg, MP3, Wave.

Stejně jako u audia, tak i video momentálně v HTML 4.0 nelze spustit bez nainstalování zásuvných modulů jiných technologií. HTML 5.0 však již tuto možnost nabízí, nachází se v něm element <video>. Stejně jako u audia i u videa najdeme atribut src pro zadání cesty k videu. Dalšími atributy jsou poster (nastavení obrázku v době načítání videa), loop (znovu přehrávání), autobuffer (automatické načítání), atd... Také zde se bohužel v dnešní době setkáváme s problémem kodeků.

3.7.4. Výhody a nevýhody

Mezi hlavní výhody AJAXu patří:

- Nepotřebuje mít nainstalovaný zásuvný modul.
- Rozšířenost – Ajax bude fungovat ve všech webových prohlížečích, ve kterých je povolený JavaScript.
- Podpora SEO.
- Multiplatformní.

Mezi nevýhody patří:

- Absence podpory některých multimédií.

3.7.5. Příklad

Vzhledem k tomu, že AJAX je dnes na Internetu oblíbenou implementační technologií, lze nalézt mnoho pěkných příkladů s touto technologií. Nakonec jsem zvolila webovou aplikaci, která ukazuje mapu zadlužení světa. Pomocí mapy lze jednoduše najet kurzorem na jednotlivé státy a zjistit jejich aktuální zadlužení. Aplikaci lze nalézt na adrese: <http://buttonwood.economist.com/content/gdc>



Obrázek 4. Aplikace Global Debt Comparison

3.8. JavaFX

3.8.1. Úvod

Platforma JavaFX pochází od společnosti Sun Microsystems. Java na Internetu ztrácela na své atraktivitě a oblibě, a proto se společnost Sun Microsystems rozhodla také zaujmout pozici ve světě RIA aplikací a její odpovědí byla právě platforma JavaFX. V dnešní době již Sun Microsystems odkoupila firma Oracle, takže bychom vlastně měli mluvit o společnosti Oracle ve spojení s JavaFX. Tato platforma umožňuje vyvíjet aplikace, jak pro webové prohlížeče, tak pro mobilní zařízení a televizory. Momentálně společnost pracuje na rozšíření pro auto navigaci.

Počátečním nápadem pro vytvoření této platformy bylo vytvořit interpretovaný skriptovací jazyk pro snadnější tvorbu uživatelského rozhraní. Tento jazyk se nazýval F3³⁶. Časem se z tohoto jazyku vyvinula celá platforma, jak pro vývojáře, tak pro designéry. Její první představení proběhlo v květnu roku 2007 na konferenci pro vývojáře nazvané JavaOne³⁷. Opět v květnu dalšího roku (2008), oznámila společnost Sun Microsystems, že má v plánu do konce roku uvolnit JavaFX pro webové prohlížeče a desktopy a na jaře dalšího roku pak pro mobilní zařízení. V prosinci tedy opravdu Sun Microsystems uvolňuje první verzi platformy JavaFX. Během doby od první zmínky o jazyku F3 k vytvoření celé platformy

³⁶F3 z anglického Form Follow Function.

³⁷JavaOne je konference konaná každý rok od roku 1996, na které vývojáři diskutují o Java technologiích.

JavaFX uběhly tři roky. JavaFX zaznamenává během své historie několik důležitých milníků. Jednak byl vytvořen kompilátor, který celý kód přeložil do byte kódu běhového prostředí Javy. Další verze JavaFX 1.1 byla vydána 12. února roku 2009, přináší sebou podporu pro vytváření aplikací na mobilní zařízení. Verze 1.2. byla vydána již v květnu téhož roku s opravami některých dřívějších chyb a se zlepšením podpory skinování jednotlivých prvků. Poslední verzí je JavaFX 1.3, která byla vydána v roce 2010. Tato verze obsahuje nové ovládací prvky, podporu pro TV emulátory, nové vlastní písmo nazvané Amble, také přichází se zlepšením výkonnosti běhu programu atd... Momentálně se očekává příchod JavaFX 2.0.

Rok	Verze
2007	JavaOne
2008	JavaFX 1.0
2009	JavaFX 1.1
2009	JavaFX 1.2
2010	JavaFX 1.3

Tabulka 3. Historický přehled vývoje JavaFX

3.8.2. Popis

Základem pro spuštění aplikací napsaných v JavaFX je mít nainstalované běhové prostředí JRE³⁸. Při vytváření nové aplikace, je aplikace překompilována do Java byte kódu. Je to vlastně univerzální přenositelný kód, který je spuštěn v běhovém prostředí virtuálního stroje (JVM – Java Virtual Machine). Tato architektura má velikou výhodu, jednotlivé aplikace se nemusí tvořit jednotlivě pro různé operační systémy, ale stačí ji vytvořit pouze jednou a měla by pracovat na jakémkoliv operačním systému, který má nainstalovaný JRE. V případě spuštění aplikace ve webovém prohlížeči, je webový prohlížeč jen jakýmsi spouštěcím mechanismem nového procesu, který je dále nezávislý na daném webovém prohlížeči. S tím souvisí i jedna z vlastností aplikací JavaFX, kde si uživatel může doslova přetáhnout aplikaci z webového prohlížeče na plochu, kde se mu automaticky vytvoří zástupce pro danou aplikaci.

Platforma JavaFX se skládá z těchto komponent:

- JavaFX SDK – obsahuje kompilátor pro jazyk JavaFX Script a nástroje běhového prostředí. Dále obsahuje knihovny pro webové služby, uživatelské prvky (UI controls), grafiku a média pro vytváření aplikací jak pro desktop, tak i pro mobilní zařízení a webové prohlížeče.

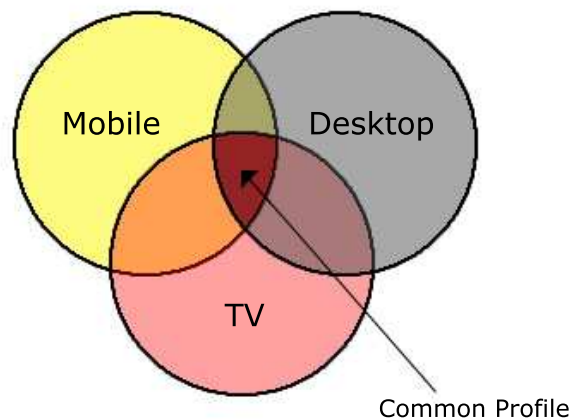
³⁸JRE z anglického Java Runtime Environment.

- NetBeans IDE pro JavaFX – vývojové prostředí pro vytváření, prohlížení a ladění JavaFX aplikací.
- JavaFX Production Suite – sada nástrojů a zásuvných modulů ulehčující práci pro webové designéry. Vývojáři JavaFX se nesnaží o to, aby se designér musel učit nové grafické prostředí, jak je tomu například u technologie Silverlight, pro kterou Microsoft vyvinul nástroj Blend. Naopak si uvědomují, že mnoho designéru pracuje s nástroji od firmy Adobe Systems Inc., jako jsou Photoshop nebo Illustrator a proto vyvinuli zásuvný modul, pomocí kterého se dají tyto grafické návrhy exportovat do formátu FXZ, který je kompatibilní s JavaFX. Druhou částí této sady je množina nástrojů konvertující SVG grafiku do kódu jazyka JavaFX Scriptu. Komprimovaný soubor FXZ obsahuje nejméně jeden soubor FXD. FXD je vlastně textový soubor, ve kterém je obsažen popis jednotlivých grafických prvků. Dále obsahuje všechny zdroje, jako jsou obrázky, fonty, atd...

JavaFX aplikace se vytváří pomocí již zmíněného staticky typovaného deklarativního skriptovacího jazyka JavaFX Script. Tento jazyk byl speciálně vyvinut pro tuto platformu. Údajně je 5krát úspornější než samotný programovací jazyk Java. Syntaxe jazyka je velice podobná skriptovacímu jazyku JavaScript. Zdrojové kódy jednotlivých souborů se ukládají s koncovkou `.fx`. Jednou z hlavních výhod JavaFX je možnost znovupoužitelnosti kódu, který je napsaný přímo v Javě. Tuto výhodu mohou využít například vývojáři, kteří chtějí pouze změnit GUI aplikace, přičemž logiku aplikace by rádi ponechali beze změny. Díky tomu lze také využít všechny knihovny, které jsou určeny pouze pro Javu. Jak již bylo zmíněno, na této platformě se dají vyvíjet aplikace pro desktop, mobilní zařízení a televizory. JavaFX se tedy dělí na tzv. profily, kde každý profil je určen pro vývoj jiných typů aplikací. Profil je vlastně sada API, které jsou pro vytváření daného typu aplikace k dispozici. Mezi profily tak existuje jakýsi průnik základních funkcí, který je společný pro všechny typy aplikací, tento profil se nazývá Common Profile. V případě, že programátor využívá ve své aplikaci pouze možnosti z tohoto Common Profile, může svoji aplikaci jednoduše zkompileovat pro všechny zmíněné typy aplikací.

Aplikaci napsanou v JavaFX lze napsat následujícími způsoby:

- Applet – JavaFX aplikace je vložena přímo do HTML kódu pomocí JavaScriptu jako Java Applet. Vytvoření Java Appletu je v NetBeans IDE prováděno automaticky. Nejprve se vytvoří komprimovaný soubor `.jar` a následně je vygenerován JNLP soubor a jednoduchá HTML stránka, ve které je applet použit. Pokud uživatel vstoupí na danou webovou stránku, aplikace je spuštěna jako součást webové stránky ve webovém prohlížeči.
- Java Web Start – aplikaci lze spustit, jako standardní desktopovou aplikaci



Obrázek 5. Průnik profilů

- Mobilní aplikace – daná aplikace je zobrazena, jako by byla spuštěna na mobilním zařízení.

3.8.3. Podpora v multimédiích

Platforma JavaFX obsahuje řadu knihoven pro podporu multimédií.

Podpora grafiky:

- Rastrová – JavaFX podporuje obrázky ve formátu PNG, JPG, BMP a GIF. Pro vložení obrázku je třeba použít balíček `javafx.scene.image`. Pokud je potřeba zobrazit některý z dalších rastrových formátů, je třeba použít externí knihovny JAI³⁹.
- Vektorová – JavaFX obsahuje podporu vektorové grafiky. Existuje více možností jak tuto vektorovou grafiku vytvořit, ať už pomocí balíčku `javafx.scene.shape`, nebo pomocí nástrojů JavaFX Production Suite. Vektorovou grafiku lze vytvořit například v Adobe Illustratoru a poté převést do akceptovatelného formátu pro JavaFX. Stejně tak můžeme mít vektorový grafický návrh vytvořený ve formátu SVG.
- 3D – V JavaFX lze implementovat tzv. perspektivní 3D grafiku. Pomocí různých efektů na daný element, lze vytvořit iluzi 3D elementu. Sun však do budoucna počítá s plnou podporou 3D grafiky.
- Animace – JavaFX obsahuje balíček `javafx.animation`. S jeho pomocí lze vytvářet animace jednotlivých elementů. Animace jsou vytvářeny na základě časové osy (Timeline) a klíčových snímků (framů).

³⁹JAI z anglického Java Advanced Imaging.

Pro přehrávání videa a audia, je vytvořen balíček `javafx.scene.media`, který obsahuje třídu `Media`. Ta je přímo určena pro práci s videem a audiem. Obsahuje například vlastnosti pro délku (`duration`), umístění zdroje (`source`), atd... JavaFX podporuje následující video kodeky: On2 VP6, WMV, H.264 (na některých operačních systémech se podpora může lišit) a tyto audio kodeky: MP3, MIDI a WAF⁴⁰.

3.8.4. Výhody a nevýhody

Mezi hlavní výhody platformy JavaFX patří:

- Multiplatformní – aplikace je spustitelná na kterémkoliv operačním systému (Windows, MacOS, Linux), kde běží JRE.
- Využití pouze sady API Common Profile, možnost zkompileovat aplikaci pro Desktop, Web, mobilní zařízení a televizor.
- Plná podpora multimédií.
- Lze využít jakoukoliv knihovnu nebo třídu napsanou pro jazyk Java.
- Vývojové prostředí je nabízeno zdarma. Klíčové elementy jsou nabízeny jako Open Source.

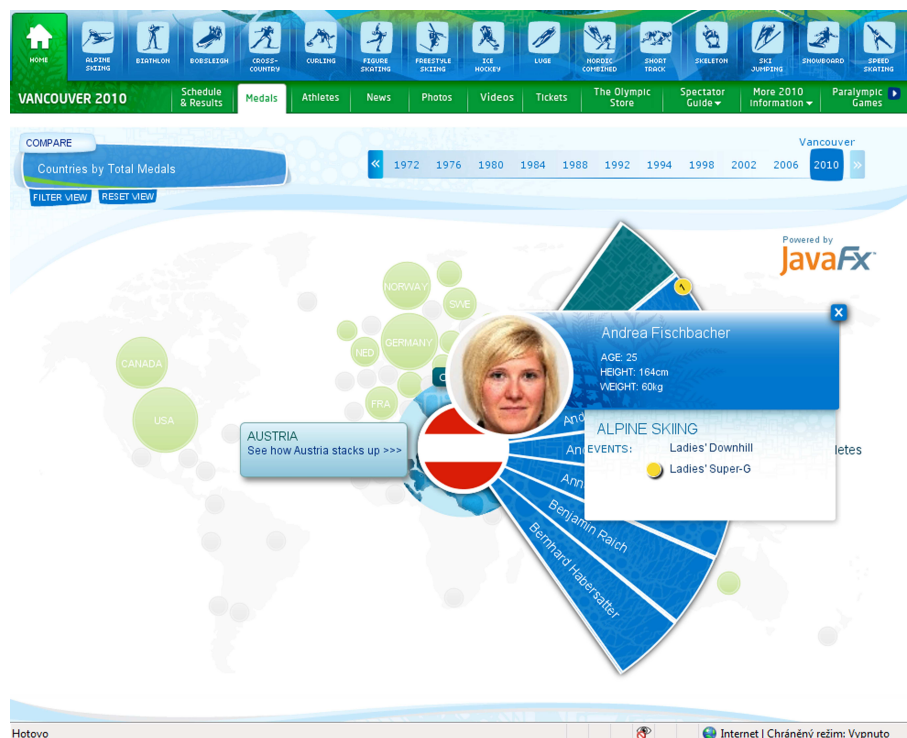
Nevýhodami jsou:

- Velice mladá technologie.
- Absence podpory SEO.
- Rozšířenost – na konci roku 2010 byla rozšířena na 80% počítačů připojených k internetu s mírně klesající tendencí.
- Nemá podporu tlačítka zpět ve webovém prohlížeči.

3.8.5. Příklad

Příkladem použití této technologie může být aplikace: <http://www.vancouver2010.com/>. Kde v sekci `Medals/Geo/View` můžeme spustit JavaFX aplikaci. Aplikace byla vytvořena pro zimní olympijské hry ve Vancouveru, které se konaly na začátku roku 2010.

⁴⁰WAF z anglického Waveform Audio Format.



Obrázek 6. Aplikace Vancouver 2010

3.9. Microsoft Silverlight

3.9.1. Úvod

Tato technologie pochází z dílny společnosti Microsoft. Silverlight vznikl za účelem konkurovat do té doby neohrožené technologii Adobe Flash. Společnost Microsoft popisuje svůj produkt následovně:

”Microsoft Silverlight je nejmodernější technologie pro internetové prohlížeče. Je to platforma určená pro tvorbu dynamického online obsahu a interaktivní práce s ním. Kombinuje text, vektorovou i bitmapovou grafiku, animace a video.”

Má opravdu Microsoft pravdu, je Silverlight nejmodernější technologií pro webové prohlížeče? Od doby svého vzniku prošel dlouhou cestu a v dnešní době je jedním z předních hráčů na poli RIA technologií. Momentálně lze tuto technologii využít pro tvorbu webových aplikací a aplikací pro Windows Phone 7. Silverlight, stejně jako předchozí technologie, patří do druhé skupiny, tedy do skupiny technologií, které pro správný běh aplikací potřebují mít nainstalovaný zásuvný modul.

Technologie Silverlight evolučně vychází z WPF⁴¹ platformy pro tvorbu desk-

⁴¹WPF z anglického Windows Presentation Foundation.

topových aplikací, který je obsažený v .NET Frameworku od verze 3.0. Silverlight byl nejdříve vyvíjen pod názvem WPF/E⁴². Tento název napovídá o veliké podobnosti obou technologií. Stejně jako WPF je i Silverlight podmnožinou .NET Frameworku. Často se píše, že Silverlight je vlastně podmnožinou WPF, přestože na začátku vývoje Silverlightu toto tvrzení mohlo být pravdivé, v dnešní době tomu již tak není. Obě technologie mají své výhody a nevýhody. V některých článcích o budoucnosti vývoje WPF a Silverlightu je možné se dozvědět, že je to právě Silverlight, který možná způsobí zánik technologie WPF.

První verze Silverlightu byla vydána v září roku 2007 – základem je nový značkovací jazyk XAML⁴³, který slouží k definování prezentačního rozhraní aplikace. Je založený na XML. Aplikační logika a obsluha událostí (například stisknutí tlačítka uživatelem) je implementována pomocí JavaScriptu. Zároveň s touto verzí je představena alfa verze 1.1.

Verze 2.0 byla vydána přibližně o rok později, do té doby byla tato verze označována jako 1.1. Přejít na verzi 2.0 znamenal obrovský skok ve vývoji. Pro aplikační logiku se již nadále nemusel používat jazyk JavaScript, ale mohly se v plné míře využívat .NET jazyky (např. C#). Změnilo se také samotné sestavení aplikace. Soubory v XAMLu jsou společně s kódem na pozadí zkompileovány do .NET assembly a zkomprimovány do souboru s koncovkou .xap. Mimo těchto novinek tato verze přináší novou množinu ovládacích prvků (TextBox, CheckBox, Slider, Grid, DataGrid, atd...) a zlepšení komunikace s webovými službami.

V červenci roku 2009 byla představena verze 3.0. Samozřejmě i tato verze sebou přináší řadu novinek. Jednou z hlavních novinek byla možnost streamování videa ve vysokém rozlišení. Aplikace již není omezena pouze na webový prohlížeč a uživatel má možnost si aplikaci nainstalovat do operačního systému (tzv. režim Out Of Browser). Dalšími novinkami jsou například podpora 3D nebo podpora SEO, atd...

Aktuální verzí je verze 4.0, která byla představena 18. listopadu 2009 na události Professional Developers Conference v Los Angeles a oficiálně vydána 15. dubna 2010. Verze obsahuje podporu pro webové kamery a mikrofony a plnou podporu tisku. Přibyly i nové ovládací prvky, kterých je již více než 60.

V roce 2011 se očekává vydání verze 5.0, která již byla představena na události Silverlight Firestarter. Tato verze by měla obsahovat nové funkce pro přehrávání multimediálního obsahu, využití procesoru grafické karty (GPU) pro realizaci 2D a 3D obsahu, podporu dálkových ovladačů atd...

Jak je vidět Microsoft se opravdu snaží o co největší prosazení této technologie a investuje do ní hodně peněz a úsilí.

⁴²WPF/E z anglického Windows Presentation Foundation/Everywhere.

⁴³XAML z anglického Extensible Application Markup Language.

Rok	Verze
2007	Silverlight 1.0
2008	Silverlight 2.0
2009	Silverlight 3.0
2010	Silverlight 4.0

Tabulka 4. Historický přehled vývoje Silverlight

3.9.2. Popis

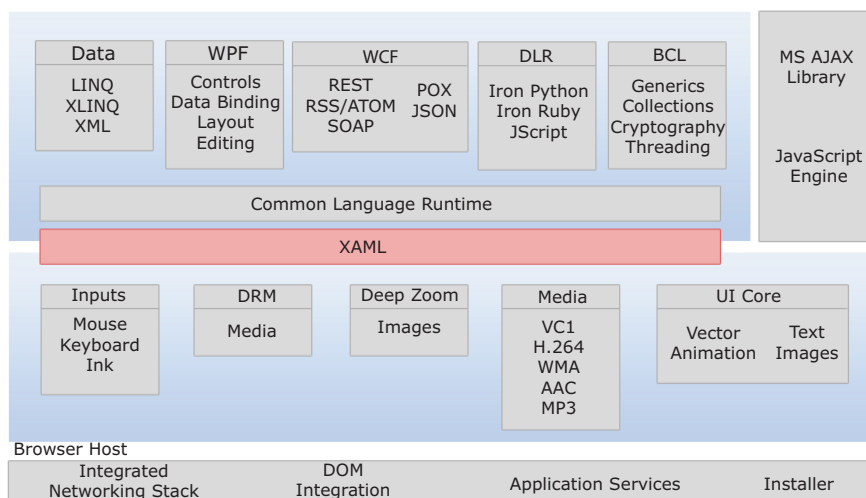
Každý uživatel, který si chce spustit aplikaci vytvořenou pomocí Silverlightu, si musí nejprve nainstalovat zásuvný modul, který slouží jako běhové prostředí pro aplikaci. Zásuvný modul má 6MB. Platformu Silverlight můžeme rozdělit na dvě části. První částí je prezentační vrstva (Core presentation framework), která obsahuje všechny komponenty a služby k vytváření uživatelského rozhraní a interakce s uživatelem. Při vytváření uživatelského rozhraní je možné využít animace, renderování vektorové a bitmapové grafiky, textový výstup a prezentaci multimediálního obsahu v různých formátech. Pod interakcí s uživatelem je možné si představit např. události vytvořené pomocí myši či klávesnice. Druhou částí je .NET Framework pro Silverlight, což je podmnožina .NET Frameworku. V podmnožině jsou obsaženy například komponenty a knihovny zahrnující integraci dat, práci se sítí, knihovny základních tříd, garbage collection a common language runtime (CLR), atd... Může se stát, že v aplikaci budou použity i knihovny, které v tomto .NET Frameworku pro Silverlight nejsou. V tom případě, jsou tyto knihovny přibaleny k aplikaci a stáhnuty do prohlížeče.

Jednou z nevýhod Silverlightu je, že není tak zcela multiplatformní. Problém nastává u platformy Linux. Pro tuto platformu vzniká open source projekt pod názvem Moonlight, který zpřístupňuje uživatelům Linuxu aplikace vytvořené pro běhové prostředí Microsoft Silverlight. Momentálně je ve verzi 2.0. Společnost Microsoft na vývoji spolupracuje se společností Novell od roku 2007.

Jak už bylo zmíněno v úvodu, značkový jazyk XAML se používá pro definování prezentačního rozhraní aplikace. Soubory psané pomocí jazyka XAML jsou označeny koncovkou .xaml. Kód na pozadí, například kód pro obsluhu událostí, se vkládá do souborů s koncovkou .cs (v případě, že je kód psaný v jazyce C#).

Microsoft vyvinul pro designéry speciální nástroj Microsoft Expression Blend, ve kterém mohou vytvářet grafické návrhy a animace. Designér může jednoduše natahat ovládací prvky, které potřebuje na pracovní plochu a příslušný XAML kód je při tomto procesu vytvářen automaticky. Díky tomuto nástroji je v mnoha ohledech zefektivněna práce a komunikace mezi designéry a vývojáři.

Pro vývoj aplikací v Silverlightu 4.0 je nejvhodnějším vývojovým nástrojem Visual Studio 2010. Tento vývojový nástroj je bohužel placený, stejně jako ná-



Obrázek 7. Architektura Silverlight

stroj pro designéry Microsoft Expression Blend. Je možné se ale setkat i s vývojovými nástroji, které jsou nabízeny zdarma. Je to například vývojové prostředí Eclipse4SL⁴⁴, které také umožňuje tvorbu aplikací v Silverlightu.

Kompilace a integrace do webové stránky. Výsledná aplikace je zkomprimována do souboru s koncovkou .xap. Jde v podstatě o ZIP archív. Důvodem komprimace souborů je dosažení co nejmenší velikosti výsledného souboru. V archívu jsou obsaženy všechny části aplikace jako jsou knihovny, zdroje (např. obrázky), soubor AppManifest (popis aplikace a vstupní bod aplikace), atd... Aplikaci napsanou v Silverlightu lze integrovat do webové stránky následujícími způsoby:

- integrace pomocí (X)HTML což je nejvhodnější způsob nasazení aplikace. V případě, že uživatel nemá nainstalovaný Silverlight, snadno mu můžeme nabídnout alternativní obsah.
- integrace pomocí JavaScriptu. K vložení aplikace do HTML stránky se použije Microsoftem zveřejněný skript SilverLight.js. Vše je však závislé na povolení skriptovacího jazyka JavaScript, dokonce i zobrazení alternativního obsahu.
- ASP.NET Silverlight Server Control – podobný způsob jako předchozí, opět se používá SilverLight.js skript. Vývojář se však o nic nemusí starat, do stránky jej automaticky emituje ScriptManager.

⁴⁴Eclipse4SL z anglického Eclipse for Silverlight.

3.9.3. Podpora v multimédiích

Silverlight obsahuje všechny základní prvky pro práci s multimédií. V následujících řádcích se budu snažit popsat, které prvky se k tomu dají využít a které formáty jsou podporovány.

Podpora grafiky:

- Rastrová - Silverlight podporuje pouze obrázky ve formátu PNG a JPEG.
- Vektorová - jazyk XAML je koncipovaný vektorově. Obsahuje základní tvary pro vektorovou grafiku, kterými jsou Line (linka), Ellipse (elipsa, kružnice), Rectangle (obdélník), Polygon (mnohoúhelník), Polyline (spojitá linka). Pro vytváření složitějších tvarů se používá System.Windows.Shapes.Path, který pomocí prvku Geometry spojuje prvky v jeden celek. Tyto tvary lze následně vyplňovat barvami, nebo různě transformovat.
- 3D - od verze 3.0 je možné implementovat perspektivní 3D grafiku. Tzn. že u každé komponenty lze aplikovat určité efekty, pomocí kterých vypadá komponenta jako prostorová (pootočení, převrácení, atd...).
- Animace - Základem pro animace u Silverlightu je tzv. animační model. Hlavní jednotkou je čas (narozdíl od Flashe, který používá snímek). Vývojář nadefinuje počáteční a koncové podmínky a Silverlight sám řeší, jak těchto podmínek dosáhnout.

Přehrávání audia a videa – je zprostředkováno pomocí elementu System.Windows.Controls.MediaElement. Tento prvek obsahuje vlastnost Source, která se nastaví na adresu URL média, dále obsahuje události, které umožňují ovlivňování přehrávání videa, např. Play, Pause, Stop, atd... Tento element je založen na jádru programu Windows Media Player. V Silverlightu lze tedy přehrávat všechny následující formáty: WMV1, WMV2, WMV3, WMV4, WMVA, WMVC1, ASX, WMA 7, WMA 8, WMA 9, MP3, MP4, DRM a podpora kodeků H.264, AAC.

3.9.4. Výhody a nevýhody

Mezi hlavní výhody technologie Silverlight patří

- Výborná podpora multimédií.
- Aplikace mohou jet v režimu "Out of Browser".
- Podpora SEO.
- Plná podpora tisku.

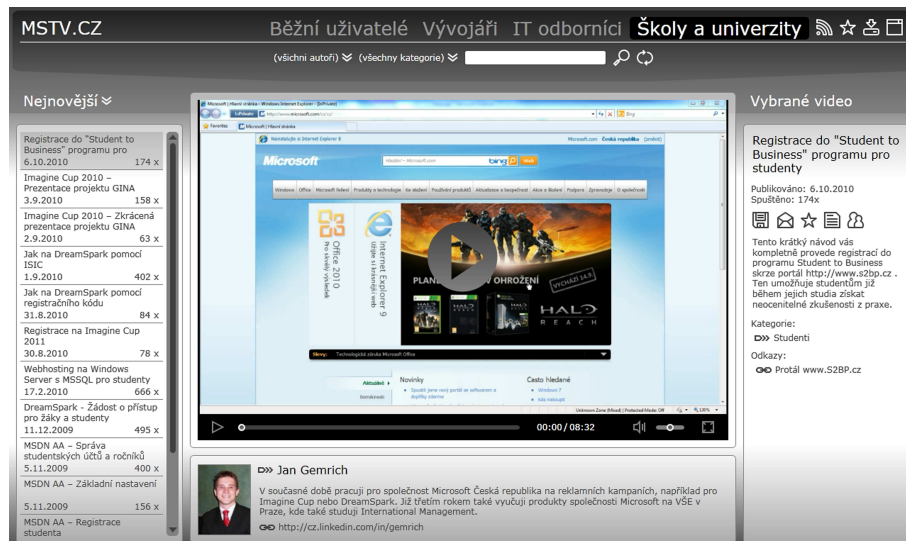
- Podpora webové kamery a mikrofону.
- Podpora tlačítka zpět ve webovém prohlížeči.

Nevýhodami jsou:

- Malá rozšířenost - na konci roku 2010 byl rozšířen na 60% počítačů připojených k Internetu s rostoucí tendencí.
- Není zcela multiplatformní.

3.9.5. Příklad

V této technologii lze najít mnoho reálných aplikací. Vybrala jsem MSTV, která byla vytvořena v roce 2008. Tato aplikace je určena pro odborníky IT a lze ji nalézt na adrese: <http://www.mstv.cz/>



Obrázek 8. Aplikace MSTV

4. Implementovaná RIA Aplikace

4.1. Úvod

Při uvažování jaký projekt je nejvhodnější pro demonstraci RIA aplikací jsem se rozhodla zaměřit na jednu technologii a v ní vytvořit větší skutečně použitelnou aplikaci.

První otázkou bylo, v jaké technologii aplikaci vytvořit. Pro mne byla volba snadná. Vzhledem k tomu, že jsem zvyklá pracovat ve vývojovém prostředí Visual Studia od Microsoftu a Silverlight se stal jednou z předních technologií na trhu RIA aplikací, rozhodla jsem se zvolit technologii Silverlight.

4.2. Základní popis systému

Cílem aplikace je vytvořit multimediální knihovnu. Pracovní název této aplikace je Video-Stop a jak sám název napovídá, hlavním účelem aplikace je prohlížení a přidávání videí. Aplikaci mohou využívat jak přihlášení, tak nepřihlášení uživatelé. Přihlášený uživatel má samozřejmě více možností než nepřihlášený. Nepřihlášený uživatel má tyto možnosti:

- Přehrávat videa.
- Vyhledávat videa.
- Hodnotit videa.
- Vytvořit si zástupce aplikace na ploše.

Přihlášený uživatel má výše zmíněné možnosti a k tomu navíc:

- Psát komentáře k videu.
- Psát si poznámky k videu a možnost si tyto poznámky vytisknout.
- Přidat si video do oblíbených.
- Pracovat se skupinami – Skupiny zde slouží ke kategorizaci oprávnění přístupu k videím. Uživatel může jak vytvářet skupiny, tak se připojovat nebo rušit již vytvořené skupiny.
- Přidat nové video.
- Upravit informace v uživatelského účtu.
- Zobrazit si statistiku o videu (kolik lidí se na video podívalo, kolikrát bylo hodnoceno, atd...).

4.3. Použité nástroje

4.3.1. Expression Blend 4.0

Při vývoji aplikace jsem použila speciální designérský nástroj pro vývoj aplikací v Silverlightu. Tento nástroj se nazývá Expression Blend. Za jeho pomoci jsem vytvořila uživatelské rozhraní aplikace. Zjednodušil mi práci s animacemi a nastavováním vizuálního vzhledu elementů.

4.3.2. Fiddler

Výborný nástroj pro zachycení HTTP provozu. Pomocí něj lze zobrazit HTTP požadavek a následně odpověď. Nástroj jsem využila mnohokrát při ladění aplikace.

4.3.3. FFMpeg

FFMpeg je open source nástroj pro práci s audio a video soubory. Umožňuje například konverzi formátů videa, streamování digitálního zvuku a obrazu, atd... V aplikaci je využit tento nástroj pro vytváření obrázků, které jsou zobrazeny, jako náhledy pro videa.

4.3.4. IIS 7

IIS 7 je webový server vyvinutý společností Microsoft. Při prvním publikování aplikace jsem chtěla využít bezplatnou hostingovou službu Aspone. Nastaly však nepředvídatelné potíže a já byla nucena hledat příčiny těchto potíží. Při tomto ladění jsem využila lokálního IIS 7 nainstalovaného na mém počítači.

4.3.5. Internet Explorer 8, Google Chrome 8, Firefox 3.5

Aplikaci jsem ladila pro všechny majoritní prohlížeče, proto má ve všech stejný vzhled a funkčnost.

4.3.6. Microsoft SQL Management Studio

Nástroj vyvinutý společností Microsoft pro práci s databázovými servery MS SQL. Pomocí tohoto nástroje byla vytvořena databáze, kterou aplikace používá.

4.3.7. Visual Studio 2010 Ultimate

Po celou dobu byla aplikace vyvíjena v tomto vývojovém prostředí. Visual Studio 2010 Ultimate je komplexní nástroj pro vývoj jak webových, desktopových, tak mobilních aplikací, ladění, atd... Vzhledem k tomu, že jsem dříve pracovala

s prostředím Visual Studio 2008, nebyl pro mě přechod na novou verzi nástroje nijak obtížný.

4.4. Použité technologie

4.4.1. ASP.NET 4.0

ASP.NET je součástí .NET Frameworku a slouží ke tvorbě webových aplikací a služeb. Je nástupcem technologie ASP (Active Server Pages).

4.4.2. C# 4.0

C# je vysokoúrovňový objektově orientovaný programovací jazyk vyvinutý firmou Microsoft zároveň s platformou .NET Framework. Byl založen na jazycích C++ a Java (a je tedy nepřímým potomkem jazyka C, ze kterého čerpá syntaxi).

4.4.3. Linq (Language Integrated Query)

Linq je dotazovací jazyk umožňující uživateli získávat informace z jakéhokoliv datového zdroje. Tento nový dotazovací jazyk se poprvé objevuje v Frameworku 3.5. Umožňuje jak získávání informací, tak modifikaci těchto informací (vkládání, mazání, editaci). V aplikaci je použit tzv. Linq To SQL, což je implementace Linqu pro Microsoft SQL Server 2000 a vyšší (v tomto případě 2008 R2) a Linq to Object.

4.4.4. Microsoft Silverlight 4.0

Viz. kapitola Microsoft Silverlight.

4.4.5. MS SQL 2008 R2 Express/Web Edition

Databázový server vyvinutý společností Microsoft. Pro vývoj na svém počítači jsem použila omezenou verzi Express. Na serveru, kde je momentálně aplikace publikována je použita verze MS SQL 2008 R2 Web Edition.

4.4.6. Silverlight Toolkit

Tento balíček se skládá z komponent a nástrojů pro vývoj aplikací v Silverlightu. Silverlight Toolkit je veden jako Open Source projekt, na kterém spolupracují hlavně vývojáři z Microsoftu.

V aplikaci jsou využity například tyto komponenty: BusyIndicator – komponenta se používá k informování uživatele o vytíženosti aplikace. Příkladem může být vyslání požadavku na uložení informací, zobrazení nových informací atd. . .

ChildWindow – komponenta zobrazující nové okno nad hlavním oknem, přičemž zamezuje interakci s modálním oknem.

DockPanel – je podobný komponentám StackPanel nebo Grid. Všechny tyto komponenty slouží k rozvržení stránky. DockPanel má však více možností než samostatný Grid nebo StackPanel. DockPanel je použit pro rozdělení hlavní stránky na části jako hlavička, menu a obsah.

WrapPanel – speciální panel s možností orientace (Horizontal, Vertical), používá se pro správné zarovnání vnitřních kontrolů.

4.4.7. Silverlight 4 Tools

Silverlight 4 Tools je doplněk pro Visual Studio 2010. Tento doplněk umožňuje využití tzv. WCF RIA Services. Tyto služby usnadňují implementaci vícevrstvých řešení pro Silverlight aplikace. Výhodou použití těchto služeb je možnost využití validace na straně serveru.

4.5. Analýza aplikace

4.5.1. Prezentační vrstva

Prezentační vrstva projektu je vytvořena pomocí šablony Silverlight Navigation Application. Tato šablona umožňuje navigaci pomocí tlačítek Zpět a Vpřed integrovaných v prohlížeči, což bývá častým problémem u aplikací, které používají asynchronní komunikaci se serverem.

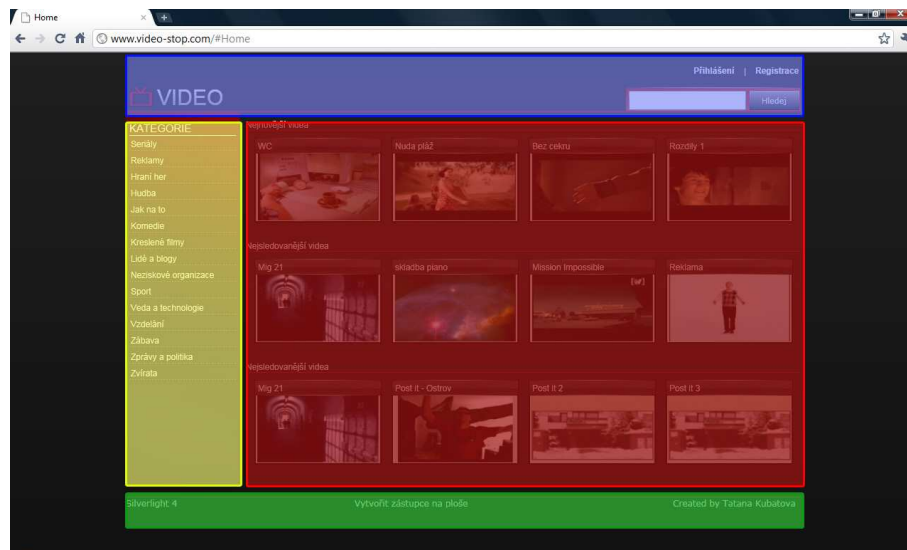
Navigace je zprostředkována pomocí prvku Frame, který je součástí System.Windows.Controls. Do tohoto prvku jsou načítány jednotlivé stránky podle daného URI⁴⁵. App.xaml obsahuje takzvaný UriMapper, který vyhledá k danému URI XAML stránku (Page). Každá XAML stránka obsahuje metodu OnNavigatedTo. V případě, že uživatel naviguje na URI, které je přiřazeno této XAML stránce, je vyvolána právě tato metoda.

Prezentační vrstva obsahuje jednu hlavní stránku MainPage.xaml, která je rozdělena na části pomocí prvku DockPanel obsaženém v Silverlight Toolkitu. Podle interakce uživatele jsou v těchto částech vykreslovány různé UserControly nebo XAML stránky. Dříve než přistoupím k jednotlivým částem stránky, ráda bych upozornila ještě na další prvky, které se na stránce vyskytují:

- BusyIndicator – prvek obsažený v Silverlight Toolkitu, pomocí kterého je zobrazena aktuální vytíženost aplikace. Tomuto prvku byl změněn vzhled pomocí přiřazeného stylu.

⁴⁵URI z anglického Uniform Resource Identifier je řetězec znaků, který slouží k přesné specifikaci zdroje informací.

- MediaElement – element obsažený v knihovně System.Windows.Controls, který slouží k zobrazení videa - v případě, že uživatel stisknul tlačítko Full Screen.



Obrázek 9. Rozdělení stránky

Na obrázku je vidět následující rozvržení: modře je hlavička, žlutě je navigační menu, zeleně je patička a červeně je obsahová část. V následujícím textu jsou popsány jednotlivé prvky podrobněji.

Hlavička obsahuje vždy:

- SearchControl - user control pro vyhledávání videí
- LoginContainerControl – user control, který zobrazuje stav a možnosti pro uživatele (Přihlášení, Registrace, Odhlášení).

Prvky, které se zobrazují pouze podle interakce uživatele:

- MenuUserControl - zobrazen pouze v případě, že je uživatel přihlášen
- AddFileControl – zobrazen pouze v případě, že uživatel uploaduje video
- RegistrationControl – tento prvek je obsažen v elementu ChildWindow (součástí Silverlight Toolkitu), prvek je zobrazen pouze při kliknutí na odkaz "Registrace"
- LoginControl – tento prvek je obsažen v elementu ChildWindow (součástí Silverlight Toolkitu), a je zobrazen pouze při kliknutí na odkaz "Přihlásit"

Navigační menu obsahuje pouze CategoryControl s odkazy na jednotlivé kategorie. K tomuto prvku je vytvořena animace, která je spuštěna při prvním načtení stránky.

Patička obsahuje pouze prvky pro vypisování textu a odkazu pro instalaci aplikace na desktop.

Obsahová část je tvořena pomocí již zmíněného prvku Frame. V této části se tedy dynamicky vykreslují jednotlivé XAML stránky. Seznam jednotlivých XAML stránek a jejich úloha:

Home – Úvodní stránka zobrazující náhodně vybraná, nejsledovanější a nejlépe hodnocená videa. Tato stránka využívá prvku SetPreviewsControl(kontejner pro jednotlivé náhledy videí) a PreviewControl(náhled videa).

ViewVideo – Stránka umožňující přehrávání videa. Tato stránka obsahuje následující user controly:

- RateControl – prvek obstarávající hodnocení videa
- TextEditorControl – textový editor, pro uložení poznámek od uživatele. Zdrojový kód editoru je převzat z webových stránek Microsoftu [http://msdn.microsoft.com/en-us/library/ff426926\(vs.95\).aspx](http://msdn.microsoft.com/en-us/library/ff426926(vs.95).aspx)
- InsertURL – okno pro vložení linku
- PrintPreview – okno s náhledem pro tisk
- AddMessageControl – prvek pro přidávání komentářů
- MessagesControl – prvek pro zobrazení jednoho komentáře
- MessageContainerControl – kontejner pro komentáře
- NoticeStyleText – prvek pro zobrazení upozorňovací zprávy
- PlayerControl – přehrávač videa

AddVideo – Tato stránka volá kód, který umožňuje upload videa (zdrojový kód se nachází v adresáři FileUpload). Tento kód jsem převzala z webové stránky <http://www.silverlightshow.net/items/uploading-multiple-files-with-silverlight-4.0.aspx>. Princip uploadu: soubor je rozdělen do stejných částí (chunks) a postupně je pomocí HttpRequestu uploadován na server. Kód umožňuje upload více videí najednou.

EditVideo – Stránka umožňující editovat video. Tato stránka využívá BaseEditVideoControl user control.

StatisticMyVideo – Stránka s náhledem statistik pro dané video.

ViewMyVideos - Stránka zobrazující poslední přidaná videa uživatelem.

ViewSortedVideos – Stránka používaná pro výpis oblíbených, vyhledávání, atd... Tato stránka využívá PreviewControl(náhled videa) user control.

EditOptions – Stránka umožňující editovat informace o uživateli. Tato stránka využívá EditUserControl user control.

ViewGroups – Stránka umožňující práci se skupinami. Tato stránka využívá následujících user controlů: AllGroupsControl, MyGroupsControl, JoinGroupControl, AddGroupControl.

AllMyVideos – Stránka, ve které jsou zobrazeny všechny videa, která byly přidány uživatelem.

UnauthorizePage – Stránka, která se zobrazí v případě, že uživatel zadal URI, na které nemá mít přístup.

4.5.2. Vrstva pro přístup k datům a logická vrstva

Pro přístup k datům jsem využila dotazovacího jazyku Linq To SQL. Logická vrstva je implementována pomocí WCF RIA Služeb a generických handlerů.

Je využito následujících generických handlerů:

CreateThumbnail.ashx – Po každém ukončení uploadu videa je zavolán tento handler pro vytvoření obrázku pro dané video.

DeleteHandler.ashx – V případě, že uživatel vybere možnost smazání videa, je zavolán tento handler, který vymaže dané soubory z disku.

UploadHandler.ashx – Tento Handler slouží k uploadu nového videa na server od uživatele.

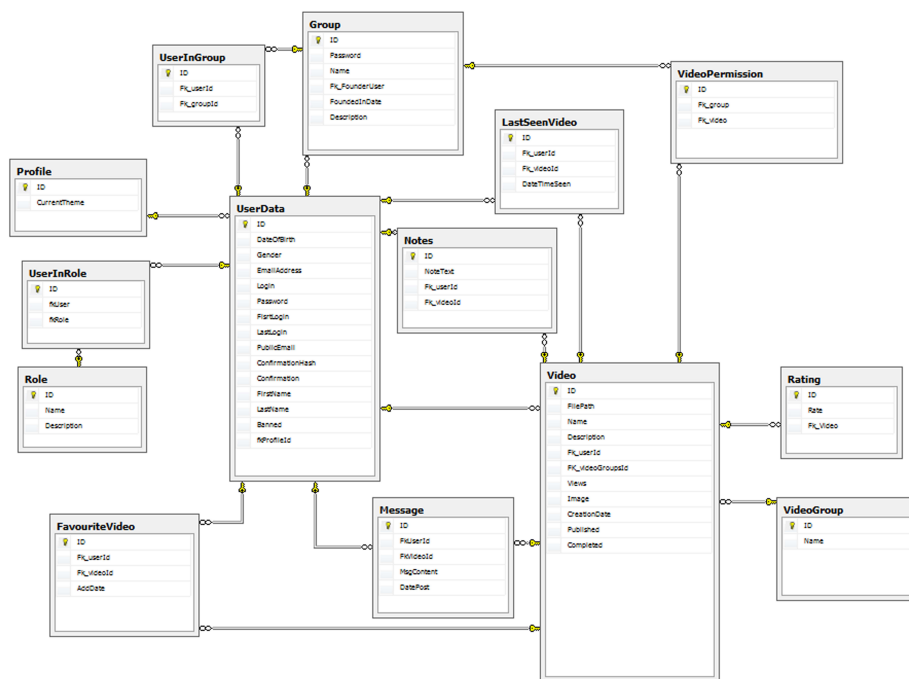
Je využito následujících WCF RIA Služeb:

UserInformationService – Tento servis slouží k navrácení informací o uživateli (přihlášení, atd...) a vytváření nových uživatelů.

VideoService – Tento servis slouží k navrácení informací, vytváření a mazání všeho, co souvisí s videi a skupinami.

4.5.3. Návrh databáze

Jako databázový server byl použit MSSQL 2008 R2. Databázové schéma je tvořeno z 15 tabulek. V databázi je používána referenční integrita. Není využito žádných uložených procedur, funkcí ani triggerů.



Obrázek 10. Schéma databáze

Popis jednotlivých tabulek:

FavouriteVideo – Tabulka sloužící k ukládání oblíbených videí.

Group – Tabulka s uživatelskými skupinami.

LastSeenVideo – Tabulka, která ukládá poslední shlédnutá videa daného uživatele.

Message – Tabulka s komentáři pro dané video.

Notes – Tabulka sloužící k uložení poznámek uživatele k určitému videu.

Profile - Tabulka s jednotlivými profily uživatelů. Momentálně obsahuje pouze jeden profil s názvem Basic. Tato tabulka byla přidána s přihlédnutím k budoucímu vývoji aplikace.

Rating – Tabulka sloužící k ukládání hodnocení videa.

Role – Tabulka s jednotlivými rolemi uživatelů. Momentálně obsahuje pouze jednu roli s názvem User. Tabulka byla přidána s přihlédnutím k budoucímu vývoji aplikace.

UserData – Tabulka uchovávající data o uživateli.

UserInGroup – Vazební tabulka mezi UserData a Group. Značí, zda je daný uživatel přidán do skupiny.

UserInRole – Vazební tabulka mezi tabulkami User a Role. Značí jakých rolí uživatel nabývá.

Video – Tabulka uchovávající informace o jednotlivých videích.

VideoGroup – Tabulka s jednotlivými video kategoriemi.

VideoPermission – Vazební tabulka mezi tabulkami Video a Group. Značí, zda je dané video přidáno do dané skupiny. V případě, že je video ve skupině, pouze uživatelé v dané skupině mají právo video shlédnout.

AddVideoFile – Tabulka nemá nyní využití, ale byla v původním návrhu aplikace a proto jsem se jí rozhodla ponechat. Tabulka má sloužit pro ukládání informací o souborech, které jsou navázány k danému videu. Základní ideou bylo nasdílení nejrůznějších dokumentů ke stažení k danému videu, např. prezentace.

4.6. Uživatelská dokumentace

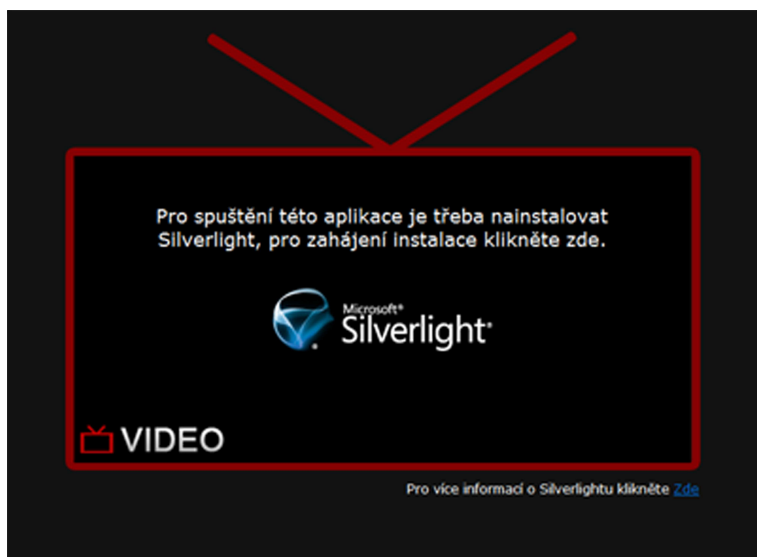
Tato kapitola slouží jako uživatelská dokumentace k případové studii.

4.6.1. Spuštění aplikace

Aplikace se spouští ve webovém prohlížeči. Přičemž technologie použité v aplikaci jsou podporovány v těchto prohlížečích: IE7/IE8, Firefox 3, Goo-

gle Chrome na těchto operačních systémech Windows XP, Windows Vista a Windows 7. V ostatních prohlížečích nebo na ostatních operačních systémech je možné, že aplikace nebude správně fungovat. (Podrobné informace naleznete zde http://en.wikipedia.org/wiki/microsoft_silverlight)

Do adresního řádku prohlížeče je třeba zadat webovou adresu: www.video-stop.com, na které se aplikace nachází. V případě, že v prohlížeči není nainstalován plug-in pro Silverlight 4 objeví se následující informativní obrazovka.



Obrázek 11. Obrazovka - není nainstalován plug-in pro Silverlight

Po kliknutí na zobrazené upozornění je zahájena instalace Silverlightu. Tato instalace může trvat několik minut a po nainstalování plug-inu je třeba restartovat webový prohlížeč.



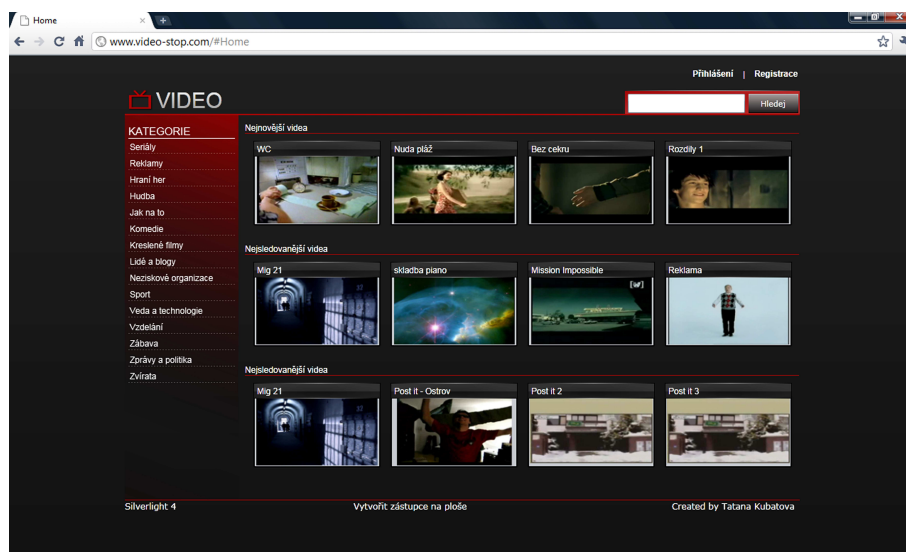
Obrázek 12. Instalace

Aplikace se do prohlížeče nejprve musí stáhnout, což může chvíli trvat. Samotná doba stažení závisí na rychlosti konkrétního Internetového připojení. Na obrazovce je zobrazen informační ukazatel o tom, kolik procent aplikace již bylo staženo. Po dotážení celé aplikace se objeví úvodní stránka.

4.6.2. Úvodní stránka

Úvodní stránku je možné vizuálně rozdělit na čtyři části:

- Hlavička – logo aplikace, link pro přihlášení a registraci, vyhledání. Po přihlášení uživatele je zobrazeno také administrační menu
- Navigace pro jednotlivé video kategorie
- Obsahová část stránky
- Patička s možností vytvoření zástupce na ploše.



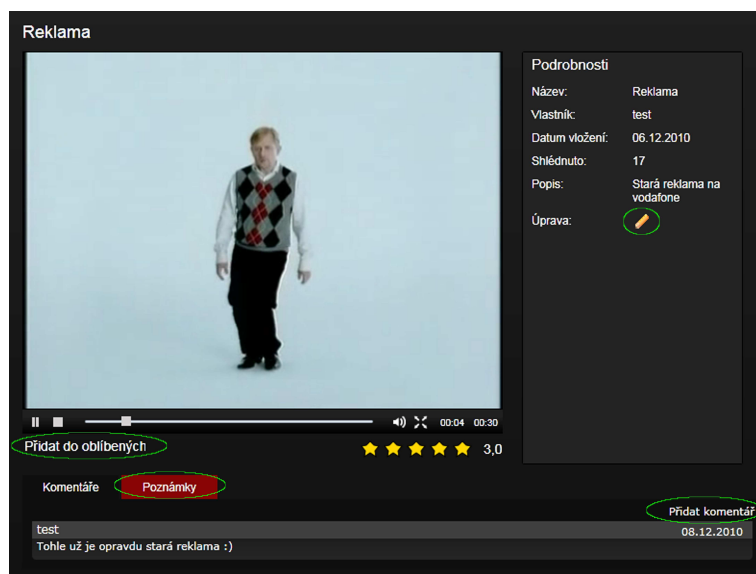
Obrázek 13. Úvodní stránka

4.6.3. Zobrazení stránky pro přehrání videa

Při výběru některého z videí je uživatel přesměrován na stránku, která umožňuje video shlédnout. Na této stránce se nachází přehrávač videa, informace o videu, hodnocení videa a komentáře od uživatelů.

Nepřihlášený uživatel má pouze možnost video přehrát a ohodnotit.

V případě, že je uživatel do aplikace přihlášen, získává rozšířené možnosti. Může přidat k videu svůj komentář, udělat si poznámky k videu nebo si video přidat do svých oblíbených. V případě, že je vlastníkem videa, má možnost se dostat na stránku pro úpravu videa. Tyto možnosti jsou vyznačené níže na obrázku pomocí zeleného zvýraznění.



Obrázek 14. Stránka pro přehrávání videa

Ovládání přehrávače videa

Přehrávač videa obsahuje základní prvky a informace o videu. Je navržen tak, aby stačil na základní manipulaci s videem. Obsahuje prvky: Play, Pause, Stop, ovládání hlasitosti, zobrazení celé obrazovky, aktuální a celkový čas videa.



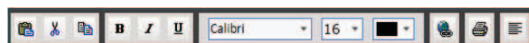
Obrázek 15. Přehrávač videa

Zapisování komentáře k videu

Uživatel má možnost přidat svůj komentář k videu a tento komentář vidí i nepřihlášení uživatelé. Komentář se přidává kliknutím na odkaz "Přidat komentář" v pravé dolní části okna.

Zapisování poznámek k videu

Uživatel má možnost zapsat si poznámky k videu a poté si poznámky vytisknout k čemuž slouží jednoduchý editor nabízející základní možnosti, jako je: změna písma, vkládání textu, zarovnání, atd. . .

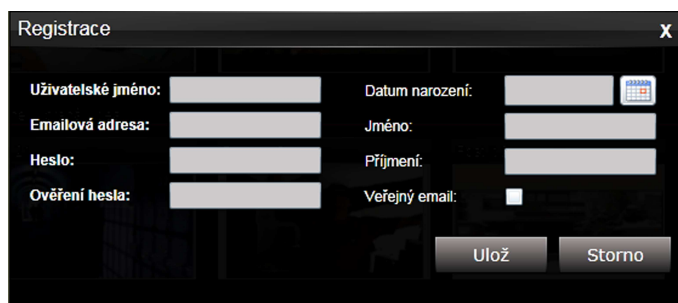


Obrázek 16. Editor pro psaní poznámek

4.6.4. Registrace

Každý uživatel má možnost se do systému kdykoliv zaregistrovat a získat tak škálu dalších možností, jako např. přidávání dalších videí, komentování, atd. . .

Pro zaregistrování je třeba kliknout na link obsažený v hlavičce s názvem Registrace. Aplikace zobrazí registrační formulář, kde je třeba vyplnit povinné položky (uživatelské jméno, emailovou adresu, heslo a ověření hesla). Ostatní nepovinné položky není třeba zadávat pokud si tak uživatel nepřeje. V případě, že některá z položek je nesprávně vyplněna, aplikace nás na to upozorní a my máme možnost tuto položku opravit. Pokud máme vše správně vyplněno, stiskneme tlačítko "Ulož". V případě, že registrace proběhla v pořádku, formulář pro registraci je skryt a uživatel je přihlášen do systému.



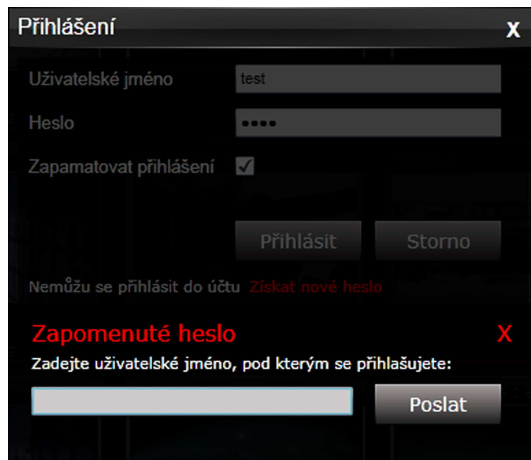
Obrázek 17. Registrační formulář

4.6.5. Přihlášení

Jediný způsob, kterým může uživatel přidávat a upravovat údaje o videích je, pokud je přihlášený do systému. Předpokládejme, že uživatel již má vytvořený uživatelský účet.

Pro přihlášení je třeba kliknout na link obsažený v hlavičce s názvem "Přihlášení". Aplikace zobrazí přihlašovací formulář. Uživatel vyplní položky pro uživatelské jméno a heslo a má možnost také zaškrtnout možnost k zapamatování těchto položek. Pokud tak učiní, při dalším přihlášení již není třeba tyto položky vyplňovat. V případě, že uživatel heslo zapomene, může požádat o nové heslo,

které mu bude posláno na email zadaný při registraci. V případě, že některé informace byly chybně zadané, aplikace na ně upozorní.



Obrázek 18. Přihlašovací formulář

4.6.6. Administrační Menu

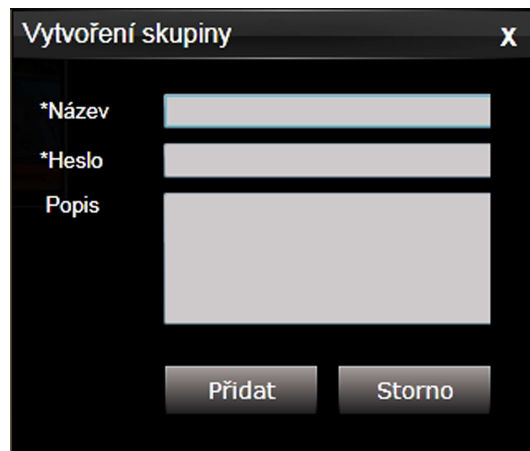
Poté, co se uživatel přihlásí, aplikace zobrazí v hlavičce administrační menu. Jednotlivé položky tohoto menu jsou:

Oblíbené - Uživatel má možnost zobrazit si všechna videa, která má momentálně přidány do oblíbených. Má možnost tyto videa spustit nebo jednotlivá videa ze seznamu odebrat.

Editace účtu - Uživatel má možnost editovat svůj účet – editovat datum narození, jméno, příjmení, veřejnost emailu nebo změnit heslo.

Skupiny - V této položce menu má uživatel možnost pracovat se skupinami. Při kliknutí na tuto položku se uživateli zobrazí submenu s následujícími položkami:

- Nejnovější videa – seznam nejnovějších videí přidanych do skupin. Uživatel má ihned přehled o tom, která videa byla nově přidána do skupin.
- Vytvořit skupinu - při výběru této položky submenu se uživateli zobrazí formulář pro vytvoření nové skupiny. Pro vytvoření nové skupiny stačí vyplnit Název a Heslo, má také možnost vyplnit i nepovinnou položku, která slouží k popisu skupiny. Pomocí zadaného jména a hesla si poté mohou ostatní uživatele tuto skupinu připojit.



Obrázek 19. Vytvoření nové skupiny

- Připojit skupinu – při výběru této položky submenu se uživateli zobrazí formulář pro připojení nové skupiny, která byla vytvořena jiným uživatelem. Pro připojení ke skupině stačí vyplnit pouze Název a Heslo. Po připojení skupiny uživatel může vidět všechna videa, která jsou přidána do této skupiny a také on jako člen skupiny má možnost do této skupiny přispívat novými videi.



Obrázek 20. Připojení skupiny

- Mé skupiny – tato položka submenu nabízí uživateli pohled na všechny skupiny, které sám vytvořil. Má možnost zobrazit si všechna videa, která jsou do skupiny přidána, zobrazit si členy skupiny, editovat skupinu nebo může skupinu ze seznamu odebrat.
- Připojené skupiny – tato položka submenu nabízí uživateli pohled na všechny skupiny, které si v průběhu času připojil. Má možnost zobrazit si seznam všech videí přidávaných do skupin, nebo jednotlivé skupiny ze seznamu odebrat.

Mé videa - V poslední položce menu má uživatel možnost přidávat a editovat přidaná videa. Při kliknutí na tuto položku se uživateli zobrazí submenu s následujícími položkami:

- Poslední přidaná videa – při výběru této položky submenu jsou uživateli zobrazena poslední videa, které on sám přidal.
- Přidat video – tato položka submenu nabízí uživateli přidat nové video. Přidání videa má dvě omezení: video může mít maximální velikost 30MB (video bylo omezeno na tuto velikost z důvodu malého prostoru na hostingovém serveru) a musí být ve formátu WMV. Pokud bylo video zařazeno do fronty, objeví se v hlavičce prvek informující o ukládání videa na server. Uživatel může sledovat vývoj ukládání videa a po uložení videa má možnost přejít přímo do editační stránky. Video není ihned publikováno veřejnosti, uživatel musí odsouhlasit na editační stránce, že ho chce publikovat. V případě, že video nebylo v pořádku uloženo, je o této skutečnosti uživatel informován.
- Seznam videí – uživateli je zobrazen seznam všech jeho doposud přidaných videí. Má možnost podívat se na statistiky k jednotlivým videím (kolik uživatelů video shlédlo, kolik ohodnotilo video, atd...) Může přejít na editační stránku videa – změnit název, popis videa, publikovat video, nebo přidat(odebrat) video do nějaké skupiny. V neposlední řadě má samozřejmě také možnost video smazat.

4.6.7. Vytvoření zástupce na ploše

Uživatel má možnost vytvořit si zástupce této aplikace na ploše. Tato možnost je nabídnuta v patičce aplikace. V případě, že uživatel otevře pomocí tohoto zástupce aplikaci, aplikace je zobrazena jako obyčejná desktopová aplikace. Uživatel má také možnost tohoto zástupce odebrat kliknutím pravým tlačítkem na aplikaci a vybrat možnost "Odebrat tuto aplikaci".

Závěr

V rámci této diplomové práce jsem se snažila přehledně vysvětlit pojem RIA a uvést základní technologie pro tvorbu bohatých internetových aplikací. Každá technologie má své výhody a nevýhody. Při vytváření aplikace, bychom měli vždy zvážit, jaké požadavky má zákazník na výsledný produkt a podle toho pečlivě vybrat technologii, která se k tvorbě produktu hodí nejvíce.

Dále jsem vytvořila aplikaci, která demonstruje použití jedné vybrané technologie. Vybrala jsem si technologii Silverlight od společnosti Microsoft. Vytvořenou aplikací je multimediální knihovna, která umožňuje, jak shlédnout, tak přidávat nová videa. Snažila jsem se přidat také funkčnost, která by se dala využívat na školách, či univerzitách, např. vytváření skupin, nebo přidávání poznámek k videu s následným tiskem. V dalším vývoji by se aplikace mohla rozšířit například o přidávání titulků do videa, nebo přidáním učebních materiálů k výukovému videu.

Conclusions

In this Msc. thesis I tried to make clear what means term RIA and present main technologies for creating Rich Internet Applications. Each technology has some advantages and some disadvantages. When we decide to create new rich internet application, we should think about what requirements does customer have and we should carefully choose technology, that is suited for this application.

In the next part of my thesis, I have created application, that demonstrates the usage of selected of one RIA technology. I chose Microsoft Silverlight technology. Application which I created is a multimedia library. This application allows user to watch video, add new videos, etc. I tried to add functionality, that can be useful for schools or universities, e.g. to create groups or add notes about video and print these notes. Next stage of the development process could focus on adding subtitles or adding documents to educational videos, which can be downloaded.

Reference

- [1] Dostál, Jiří. *Multimedia, hypertext a hypermedia teaching aids – a current trend in education*. Elektronická publikace, 2009.
- [2] Sládek, Jan. *Webdesignérův průvodce po HTML5 - pohyblivé obrázky, 2010*. <http://zdrojak.root.cz/>
- [3] <http://interval.cz/clanky/macromedia-flash-mx-2004/>
- [4] <http://cs.wikipedia.org/>
- [5] Joshua Noble, Todd Anderson, Garth Braithwaite, Marco Casario, Rich Tretola. *Flex 4 Cookbook*. O'Reilly Media Inc., 2010.
- [6] Anthony Franco, The EffectiveUI Team *Flex Early Evaluation: Assessing Flex and Your Project Needs*. O'Reilly Media Inc., 2007.
- [7] <http://www.adobe.com/>
- [8] <http://help.adobe.com/>
- [9] <http://digiweb.ihned.cz/c1-24524330>
- [10] <http://www.adobe.com/aboutadobe/invrelations/adobeandmacromedia.html>
- [11] Lacko, Ľuboslav. *Silverlight 4*. Elektronická publikace, 2010.
- [12] Lecrenski, Nick. *Silverlight 4 Problem - Design - Solution*. Wiley Publishing Inc., Indianapolis, 2010.
- [13] Jim Clarke, Jim Connors, Eric Bruno. *JavaFX Developing Rich Internet Applications*. Prentice Hall, 2009.
- [14] <http://javafx.com>
- [15] Dana Moore, Raymond Budd, Edward Benson. *Professional Rich Internet Applications: AJAX and Beyond*. Wrox, 2007.
- [16] Schaeffer, Mark. *Adobe Flash CSS Professional 100 nejlepších postupů*. Computer Press a.s., Brno, 2009.
- [17] Johnson, Steve. *Adobe Flash Professional CS5 on Demand*. Johnson Perpection Inc., 2010.

A. Obsah příloženého CD

Součástí diplomové práce je CD, na kterém se nachází vytvořená aplikace a dokumentace. CD má následující adresářovou strukturu:

`bin/`

Adresář obsahuje ZIP archív, který je nachystán pro nasazení aplikace na webový server. Dále tento adresář obsahuje skripty pro vytvoření databáze.

`doc/`

Adresář obsahuje dokumentaci diplomové práce ve formátu PDF a ZIP archív s názvem `tex`, který obsahuje vše, co je nutné pro úplné vygenerování nového PDF souboru dokumentace (zdrojový text dokumentace, vložené obrázky, atd...).

`src/`

Adresář obsahuje všechny zdrojové kódy programu včetně skriptů pro vytvoření nové databáze. Dále obsahuje instalace programů, které je nutné nainstalovat pro spuštění aplikace ve vývojovém prostředí Visual Studio 2010.

`readme.txt`

Textový soubor `readme.txt`, obsahuje instrukce pro nasazení webové aplikace na webový server, včetně požadavků pro její provoz. Dále obsahuje webovou adresu, na které je aplikace nasazena pro testovací účely a pro účel obhajoby práce.

Zadání diplomové práce

Autor: Bc. Taťána Kubátová

Vedoucí: RNDr. David Skoupil

Název: Rich Internet Applications

Název: Rich Internet Applications

Požadavky:

- Analýza problematiky RIA.
- Popis základních technologií tvorby RIA (Javascript, AJAX, Macromedia Flash, Microsoft Silverlight, ActiveX ...).
- Popis podpory multimédií v RIA.
- Vytvoření rich internet aplikace na platformě ASP.NET demonstrující nasazení vybraných technologií.

Literatura

- Dana Moore, Raymond Budd, Edward Benso: Professional Rich Internet Applications: AJAX and Beyond, Wrox 2007, ISBN 978-0470082805
- Adam Nathan: Silverlight 1.0 Unleashed, Sams 2007, ISBN 978-0672330070.
- Microsoft MSDN [online] , last revision 2002 [cit 2004-05-25]. Dostupné z WWW: <http://msdn.microsoft.com>
- Andrew Troelsen, C# a .NET 2.0 profesionálně, Zoner Press, 2006, ISBN 80-86815-42-0
- Simon Robinson, K. Scott Allen, Ollie Cornes, Jay Glynn, Zach Greenvoss, Burton Harvey, Christian Nagel, Morgan Skinner, Karli Watson: C# Programujeme profesionálně, Computer Press 2005
- Chris Sells, C# a WinForms - programování formulářů Windows, Zoner Press, 2005, ISBN 80-86815-25-0