

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## PŘEDPOVĚĎ POČASÍ – MASHUP

BAKALÁŘSKÁ PRÁCE

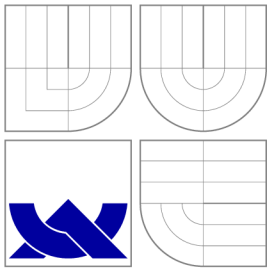
BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

RADEK GAJDUŠEK

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## PŘEDPOVĚĎ POČASÍ – MASHUP

WEATHER FORECAST – MASHUP

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

RADEK GAJDUŠEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. RUDOLF KAJAN

BRNO 2010

## **Abstrakt**

Cílem bakalářské práce „Předpověď počasí – mashup“ je návrh a implementace aplikace, shromažďující data z více zdrojů ve formátu XML za použití technologie Silverlight. Práce obsahuje kromě popisu jednotlivých fází vývoje aplikace také objasnění pojmu mashup, rozebírá Web 2.0 a definuje základní pojmy z oblasti předpovědi počasí. Poměrně velký prostor je věnován technologii Silverlight a jejímu srovnání s hlavním konkurentem na trhu. Součástí práce je také zhodnocení dosažených výsledků společně se zamyslením nad možným dalším vývojem aplikace.

## **Abstract**

The goal of bachelor's thesis "Weather forecast – mashup" is the design and implementation of application, which gathers data from multiple sources in XML format using Silverlight technology. In addition to the description of the individual stages of application development the thesis also contains the clarifying of the term mashup, analyses Web 2.0 and defines the basic terms of weather forecast. Attention is paid to Silverlight and its comparison with the main competitor in the market. The thesis also includes the evaluation of the achieved results, together with thinking about the future possible application development.

## **Klíčová slova**

Předpověď počasí, Mashup, Silverlight, Web 2.0, XAML

## **Keywords**

Weather forecast, Mashup, Silverlight, Web 2.0, XAML

## **Citace**

Radek Gajdušek: Předpověď počasí – mashup, bakalářská práce, Brno, FIT VUT v Brně, 2010

# Předpověď počasí – mashup

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Rudolfa Kajana a uvedl všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Radek Gajdušek  
19. května 2010

## Poděkování

Rád bych poděkoval svému vedoucímu Ing. Rudolfu Kajanovi za jeho odborné vedení, rady a připomínky, díky kterým se podařilo práci úspěšně dokončit. Dále bych rád poděkoval organizacím, které poskytly svá data pro potřeby mé práce, konkrétně společnosti Meteo-centrum.cz za poskytnutí dat s předpovědí počasí pro Českou republiku, panu Ing. Davidu Královi z Českého hydrometeorologického ústavu za poskytnutí dat s výstrahami pro Českou republiku a v neposlední řadě také společnosti IN-POČASIE za poskytnutí dat s předpovědí počasí pro Slovensko. Nerad bych zapomněl také na mou rodinu, bez jejichž podpory bych práci nedokončil, proto i jim patří velký dík.

© Radek Gajdušek, 2010.

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



# Obsah

<b>1</b>	<b>Úvod</b>	<b>3</b>
<b>2</b>	<b>Silverlight</b>	<b>4</b>
2.1	Představení technologie	4
2.1.1	XAML	5
2.2	Historie platformy	5
2.3	Srovnání s technologií Adobe Flash	7
2.4	Vývojové nástroje	8
2.4.1	Microsoft Visual Studio 2008	9
2.4.2	Microsoft Expression Blend 3	9
<b>3</b>	<b>XML</b>	<b>11</b>
3.1	Charakteristika jazyka	11
3.2	Vývoj jazyka	12
3.3	Použití XML	12
<b>4</b>	<b>Mashup</b>	<b>14</b>
4.1	Objasnění pojmu a příklady aplikací	14
4.2	Web 2.0	15
4.3	Souvislost mashup aplikací s Web 2.0	16
4.4	Budoucnost mashup aplikací	16
4.5	Mashup editory	16
4.5.1	Yahoo pipes	16
4.5.2	OpenKapow RoboMaker	17
<b>5</b>	<b>Předpověď počasí</b>	<b>19</b>
5.1	Základní pojmy	19
5.2	Historie předpovídání počasí	19
5.3	Druhy vydávaných předpovědí	20
<b>6</b>	<b>Analýza</b>	<b>22</b>
6.1	Zadání práce a motivace	22
6.2	Zmapování dostupných zdrojů	23
6.2.1	České zdroje	23
6.2.2	Slovenské zdroje	24
6.2.3	Zahraniční zdroje	25
6.2.4	Zhodnocení zdrojů	26

<b>7</b>	<b>Návrh</b>	<b>27</b>
7.1	Funkcionalita aplikace . . . . .	27
7.1.1	Výběr zdrojů . . . . .	28
7.1.2	Zobrazované informace . . . . .	29
7.1.3	Stavy oblačnosti, výstražné informace a varování . . . . .	29
7.2	Grafické uživatelské rozhraní . . . . .	31
7.3	Zřízení domény a blogu . . . . .	31
<b>8</b>	<b>Implementace</b>	<b>33</b>
8.1	Aplikace . . . . .	33
8.1.1	Inicializace a načtení dat . . . . .	34
8.1.2	Zobrazení předpovědi a výstrah . . . . .	35
8.1.3	Vyhledávání pro konkrétní město . . . . .	35
8.1.4	Ostatní části . . . . .	36
8.2	Grafické uživatelské rozhraní . . . . .	36
8.2.1	Ikonky oblačnosti a výstrah . . . . .	37
8.2.2	Mapky států s vyobrazením krajů . . . . .	38
8.3	Ostatní detaily . . . . .	38
8.4	Problémy při implementaci . . . . .	39
8.4.1	Vyhledávání v celoobrazovkovém režimu . . . . .	39
8.4.2	Bezpečnostní politika . . . . .	39
<b>9</b>	<b>Testování</b>	<b>40</b>
9.1	Předpověď počasí . . . . .	40
9.2	Výstražné informace a varování . . . . .	42
9.3	Vyhledávání předpovědi . . . . .	43
9.4	Ostatní funkcionalita . . . . .	45
<b>10</b>	<b>Závěr</b>	<b>47</b>
<b>A</b>	<b>Čestné prohlášení</b>	<b>50</b>
<b>B</b>	<b>Zdrojový kód skriptu</b>	<b>51</b>
<b>C</b>	<b>Ukázka XAML</b>	<b>52</b>
<b>D</b>	<b>Obsah CD</b>	<b>53</b>

# Kapitola 1

## Úvod

Předmětem této bakalářské práce je vytvoření webové aplikace, která bude zobrazovat předpověď počasí a výstrahy pro Českou a Slovenskou republiku na několik dní dopředu. Aplikace má používat zdroje ve formátu XML a bude naprogramována za použití technologie Silverlight.

Práce je rozdělena na dvě pomyslné části, z nichž první je zaměřena spíše teoreticky a uvádí čtenáře mimo jiné do problematiky tvorby aplikací shromažďujících data z více zdrojů na jednom místě, tzv. mashupů, zatímco druhá část popisuje jednotlivé fáze vývoje softwarového produktu.

Na začátku teoretické části je představena technologie Silverlight a její hlavní komponenty. V této kapitole nechybí také srovnání s hlavním konkurentem na trhu. Následuje kapitola věnující se značkovacímu jazyku XML. Obsahem čtvrté kapitoly je představení pojmu mashup, jeho souvislost s Web 2.0 a autorovo zamýšlení nad budoucností mashup aplikací. Teoretickou část práce uzavírá kapitola týkající se předpovědi počasí, která se zabývá historií a základními pojmy z tohoto odvětví.

Ve druhé části, která popisuje již samotnou tvorbou aplikace, se jako první nachází kapitola věnující se analýze zadání práce a dostupných datových zdrojů na internetu. Následuje kapitola o návrhu aplikace, nejprve z pohledu funkcionality a poté je věnován prostor grafickému uživatelskému rozhraní. Cyklus vývoje aplikace je zakončen kapitolou popisující testování hotového programu, v níž jsou podrobně popsány jednotlivé testy, které byly provedeny za účelem ověření správnosti fungování aplikace.

Závěrečná kapitola obsahuje zhodnocení dosažených výsledků práce autorem a zamýšlení nad dalším možným vývojem aplikace.

## Kapitola 2

# Silverlight

Tato kapitola seznamuje čtenáře s technologií Silverlight od firmy Microsoft a představuje její možnosti. Nechybí také srovnání této technologie s jejím hlavním konkurentem na trhu, kterým je Flash od firmy Adobe. V závěru kapitoly jsou zmíněny vývojové nástroje úzce související s vývojem Silverlight aplikací.

### 2.1 Představení technologie

Interaktivitu webových stránek výrazně zvyšuje technologie Silverlight od firmy Microsoft, která je podmnožinou technologie Windows Presentation Foundation (WPF).

WPF obohacuje klasické formulářové aplikace, tzv. *WinForms aplikace*<sup>1</sup>, o možnost přidání propracovanější grafiky a dává tak vzniknout bohatým grafickým aplikacím. Do příchodu WPF renderoval grafickou stránku aplikace procesor a grafická karta zůstávala nevyužitá, zatímco nyní je grafika renderována prostřednictvím grafické karty, což významně ulehčuje procesoru, který může uspořené prostředky použít jiným způsobem. Tato technologie se objevila poprvé s příchodem operačního systému Windows Vista a v současné době je na ní kompletně postaveno nové Visual Studio 2010. WPF pracuje se značkovacím jazykem XAML, který je více popsán v podkapitole 2.1.1 [14].

Silverlight rozšiřuje možnosti webového prohlížeče o možnost práce s vektorovou grafikou a použití multimediálního obsahu. Jedná se o multiplatformní plug-in pro webový prohlížeč podobný konkurenční technologii firmy Adobe zvané Flash. Flash je stejně jako Silverlight multiplatformní zásuvný modul pro prohlížeč, používaný především k tvorbě animací, vytváření reklamních bannerů a her.

Instalace zásuvného modulu je velmi jednoduchá a probíhá v prostředí internetového prohlížeče. V případě, že stránka na níž uživatel vstoupil používá technologii Silverlight a prohlížeč nemá nainstalován plug-in pro Silverlight, je uživatel vyzván k jeho instalaci.

Silverlight aplikace je v podstatě pracovní a interaktivní zobrazovací prostor běžící na klientovi ve webovém prohlížeči, zatímco veškerá aplikační logika je provozována na serveru. Technologie Silverlight neklade žádné nároky na server, ten pouze poskytuje prohlížeči na základě běžného GET požadavku XAP soubor<sup>2</sup>, což je binární soubor se zkompilovanou aplikací a je zobrazován prostřednictvím webového prohlížeče klientovi.

---

<sup>1</sup><http://msdn.microsoft.com/en-us/library/dd30h2yb.aspx>

<sup>2</sup><http://pietschsoft.com/post/2008/03/Silverlight-Anatomy-of-an-XAP-file.aspx>

Platformu Silverlight tvoří tyto dvě základní části:

### 1. Prezentační vrstva

Tato vrstva obsahuje komponenty pro generování uživatelského rozhraní a interakci s uživatelem. Lze díky ní vykreslovat vektorovou a bitmapovou grafiku, textový výstup, animace nebo multimediální obsah v různých formátech. Rovněž obsahuje prostředky pro obsluhu událostí generovaných pomocí myši a klávesnice.

Zde lze zařadit DOM API pro Silverlight nebo speciální značkovací jazyk XAML, o němž se dozvíte více dále.

### 2. Instalační a aktualizací komponenta

Jedná se o vrstvu, která má na starosti automatické aktualizace technologie Silverlight a instalaci nových verzí.

V současné době Silverlight podporuje všechny nejčastěji používané prohlížeče, rovněž funguje na zařízeních s operačním systémem Windows Mobile a v neposlední řadě také na operačních systémech unixového typu, pro které je určena opensource implementace zvaná *Moonlight*.

#### 2.1.1 XAML

Pro potřeby technologie Silverlight a Windows Presentation Foundation byl vyvinut speciální značkovací jazyk, jehož základem se stal populární jazyk XML. Tento jazyk odděluje programovou část od uživatelského rozhraní a činí tak kód zapsaný v tomto jazyce velmi čitelným [8].

Pro představu bych rád uvedl ukázkou jednoduchého XAML souboru, který jsem převzal z webu<sup>3</sup>. Tato část kódu zobrazí na stránce tučně modrou barvou větu „Hello world!“.

```
<Window x:Class="HelloWorld.Window1"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  Title="HelloWorld" Height="300" Width="300">

  <Label Background="LightBlue" FontWeight="Bold" FontSize="20">Hello World!</Label>
</Window>
```

Pro snadnější editaci XAML souborů byl vytvořen firmou Microsoft nástroj Expression Blend (v aktuální verzi 3), který je součástí balíčku Expression Studio. Na internetu je bezplatně ke stažení také jeho 30-ti denní zkušební verze. Více bude tento nástroj popsán v podkapitole 2.4.2, která se věnuje vývojovým nástrojům spojených s technologií Silverlight.

## 2.2 Historie platformy

První verze technologie Silverlight byla představena firmou Microsoft v září roku 2007. Tehdy vyšly dvě verze – finální verze Silverlight 1.0 a společně s ní alfa verze 1.1. Po roce vývoje verze 1.1 došlo k jejímu přejmenování na Silverlight 2.0.

<sup>3</sup><http://www.netstudent.cz/%C4%8C1%C3%A1nky/tabid/56/articleType/ArticleView/articleId/53/1-WPF-Hello-XAML-World.aspx>

Zásadní rozdíl mezi Silverlight 1.0 a 2.0 byl v tom, že první verze využívala jako programovací jazyk jen Javascript, zatímco druhá umožňovala využívat v plné míře .NET jazyky mezi něž patří mimo jiné C# a Visual Basic. Obliba této technologie rychle stoupala a za necelé dva roky od jejího vzniku bylo na světě zaregistrováno více jak 10 000 aplikací. Samotná firma Microsoft Silverlight používá pro většinu svých webů. Za zmínku stojí také to, že Silverlight byl použit televizní společností NBC k online přenosům olympijských her z Pekingu v roce 2008, kdy tyto přenosy shlédlo přes 52 milionů návštěvníků [9].

Poté následovala verze 3.0, která přinesla mnoho vylepšení, mezi nejvýznamnější se řadí možnost běhu aplikace na klientském počítači mimo prohlížeč nebo možnost streamování videí v úplné HD kvalitě. Posledně zmiňovanou komponentu využila také Česká televize, která během Zimní olympiády ve Vancouveru na 6 online kanálech vysílala přenosy právě v HD rozlišení.

Nejnovější verze Silverlightu vyšla v dubnu tohoto roku a nese označení 4.0. Každá nově vydaná verze obsahuje možnosti a technologie všech verzí předchozích.

Na závěr povídání o historii této platformy bych rád uvedl malý výčet dalších zajímavých novinek, které (kromě výše uvedených) přinesly jednotlivé verze:

### **Verze 1.0**

- Podpora přehrávání videa ve formátu WMV a zvuku ve formátu MP3.
- Podpora 2D vektorové grafiky a animací.
- Podpora technologie AJAX.
- XML parser podporující formát XAML.

### **Verze 2.0**

- Rozsáhlá podpora síťových služeb.
- Pokročilá ochrana obsahu pomocí Silverlight DRM.
- Pokročilá podpora skinovatelnosti a šablon.

### **Verze 3.0**

- Podpora hardwarové akcelerace, 3D transformací.
- Zlepšení animací a grafické efekty na úrovni pixelů.
- Datový formulář a Save File dialog.

### **Verze 4.0**

- Podpora mikrofonu a webkamery.
- Možnosti tisku přímo z aplikace.
- Podpora prohlížeče Opera a Chrome.

## 2.3 Srovnání s technologií Adobe Flash

Pro srovnání možností technologie Silverlight s jiným produktem na trhu jsem zvolil jejího největšího konkurenta, technologii Flash od známe softwarové firmy Adobe. Ve srovnání se zaměřím nejprve na webové varianty obou technologií a poté se zmíním o možnostech přenesení webové aplikace na desktopy.

Obě technologie se velmi významným způsobem zaměřují na přehrávání videa v prohlížeči, umožňují přehrávání HD videa v celoobrazovkovém režimu, podporují hardwarovou akceleraci a rozložení výkonu na více jader procesoru. Samozřejmostí je také schopnost přizpůsobení přehrávání videa na základě vytíženosti datové linky a aktuálního výkonu počítače, na němž daná aplikace běží. O ochranu přehrávaných videí před kopírováním a kontrolu jejich užívání se stará DRM modul, který je implementován v obou technologiích.

Dalším zajímavou oblastí pro srovnání jsou animace. Zde je práce s technologií Silverlight jednodušší. Je třeba pouze definovat počáteční, koncové podmínky a délku animace, transformaci a ostatní nutnosti jsou pak v režii Silverlightu. Ve Flashi je naopak nutné pracovat s maticemi a počítat pozice objektů v čase, neboť Flash používá sekvenční procházení snímku [1].

Z hlediska jazyků, v nichž je možné aplikace vyvíjet, nese Flash jistá omezení. Jediným jazykem, v němž lze psát aplikace je ActionScript, což je jazyk stvořený přímo firmou Adobe. Silverlight aplikace naproti tomu lze psát jak v kompilovaných jazycích skupiny .NET (C#, Visual Basic), tak i interpretovaných jakými jsou například jazyky IronPython nebo IronRuby. V tomto ohledu si myslím, že Silverlight jasně převyšuje Flash a díky tomu si získal přízeň lidí, kteří již některý z těchto jazyků ovládají a nechtějí se učit nový jazyk.

O komplexnější srovnání jsem se pokusil v tabulce 2.1, kde lze najít podporované audio a video formáty u obou technologií [11]. Rovněž se v tabulce vyskytují formáty obrázků, které lze při vývoji aplikací použít. Pro zajímavost je v tabulce dále uvedena velikost pluginu. Srovnání bylo provedeno pro verze Silverlight 3.0 a Adobe Flash 10.

Obě technologie také disponují možnostmi přenesení aplikace z webového prohlížeče na počítač a vytvořit tak desktopovou aplikaci. Microsoft pro svou technologii používá označení Silverlight OOB<sup>4</sup>, zatímco Adobe pro svůj produkt používá název Adobe AIR<sup>5</sup>. Instalace aplikace Adobe Air se velmi podobá klasické instalaci programu v prostředí operačního systému, zatímco instalaci Silverlight OOB<sup>6</sup> aplikace bych přirovnal svým průběhem k instalaci zásuvného modulu do prohlížeče. V oblasti desktopových aplikací přenesených z webu jasně vítězí produkt Adobe AIR nad Silverlight OOB. Desktopové varianty obou produktů disponují prakticky stejnými možnostmi jako jejich webové varianty, především Adobe AIR však navíc přináší další užitečné funkce, typické pro desktopové aplikace. Hlavní rozdíly mezi oběma technologiemi jsem se pokusil shrnout do tabulky 2.2.

Z obou srovnání je vidět, že technologie Silverlight je oproti Flashi velmi mladá, ale postupně Flash dohání, v některých případech z minulosti dokonce přišla jako první s funkcemi, kterými ani konkurence nedisponovala, jako například bylo zmiňované DRM či přehrávání videa v HD kvalitě. V tu chvíli museli vývojáři v Adobe zapracovat a nedostatky rychle implementovat, aby udrželi se Silverlightem krok. Osobně si myslím, že konkurence je v tomto směru velmi dobrá a tím, kdo na ní nakonec vydělá budou vývojáři a v neposlední řadě také uživatelé. Domnívám se, že Flash prohodil náskok, který oproti Silverlightu měl a ten nyní určuje hlavní směr vývoje.

<sup>4</sup><http://www.microsoft.com/SILVERLIGHT/silverlight/demos/OOB/default.html>

<sup>5</sup><http://www.adobe.com/cz/products/air/>

<sup>6</sup>zkratka OOB vznikla z anglického slova Out Of Browser (mimo prohlížeč)



	<b>Silverlight</b>	<b>Flash</b>
Jazyky pro vývoj	.NET jazyky IronRuby, IronPython, ...	pouze ActionScript
Formát ukládání tvarů	textový formát (XAML)	binární (SDK)
Přízpusobení přehrávání	ano (Smooth Streaming)	ano (Dynamic Streaming)
Hardwarová akcelerace	ano	ano
Podpora vícejadrových procesorů	ano	ano
Ochrana obsahu DRM	ano	ano
HD video ve fullscreen	ano	ano
Animace	časová osa	práce s maticemi
Podporované video formáty	WMW, MP4	SWF, F4V
Podporované audio formáty	WAW, WMA, MP3	AAC, MP3
Podporované formáty obrázků	PNG, JPEG	GIF, PNG, JPEG
Podporované prohlížeče	Internet Explorer, Firefox, Safari, Chrome, Opera	Internet Explorer, Firefox, Safari, Chrome, Opera
Velikost plug-inu	4,8 MB	1.9 MB

Tabulka 2.1: Srovnání Silverlight a Flash.

	<b>Silverlight OOB</b>	<b>Adobe AIR</b>
Přístup k lokálním datům uživatele	ne	ano
Vestavěná SQL databáze	ne	ano
Aktualizace	automatická	nutno naprogramovat
Notifikace u systémových hodin	ne	ano
Spouštění více oken	ne	ano

Tabulka 2.2: Hlavní rozdíly mezi Silverlight OOB a Adobe AIR

V oblasti „desktopových aplikací“ vede z mého pohledu jednoznačně produkt Adobe AIR, je však otázkou, zda Microsoft zapracuje na nedostatecích a přinese s novou verzí Silverlightu vylepšení nebo se vydá opačnou cestou než Adobe a zvolí vlastní přístup k webovým aplikacím přeneseným na desktopy.

## 2.4 Vývojové nástroje

Vývoj Silverlight aplikací ulehčují a zefektivňují nástroje firmy Microsoft. Velkou výhodou níže uvedených vývojových nástrojů je jejich provázanost mezi sebou, což je pro vývojáře velmi přívětivé. Na tyto nástroje bude v této části nahlíženo jako na pomocníky vývojářů, a proto se pokusím nastínit jejich přínos při vývoji Silverlight aplikací a připojím své vlastní postřehy při práci s těmito nástroji.



### 2.4.1 Microsoft Visual Studio 2008

Pro vývoj Silverlight aplikací v Microsoft Visual Studio 2008 je potřeba mít nainstalován Service Pack 1 a dále doplněk Silverlight Tools 3.0. Po instalaci zmíněných komponent je Visual Studio plně připraveno pro tvorbu aplikací pomocí technologie Silverlight. Zmíněná verze Visual Studio je postavena na .NET Framework 3.5.

Samotné vývojové prostředí není nutné podle mě nijak dalece představovat, neboť je většině programátorů dobře známé nebo o něm alespoň slyšeli. Za zmínku stojí snad jen funkce automatického doplňování kódu zvaná *IntelliSense*, která podle mého názoru velmi usnadňuje práci, funguje spolehlivě a často pomáhá také s odhalením chyb.

Při vývoji Silverlight aplikace, která je předmětem mé bakalářské práce, budu používat studentskou licenci Microsoft Visual Studio 2008 dostupnou na serveru MSDNAA<sup>7</sup>.

V měsíci dubnu 2010 vyšla nová verze s označením Microsoft Visual Studio 2010, která přináší mimo jiné nový .NET Framework 4.0.

### 2.4.2 Microsoft Expression Blend 3

Microsoft Expression Blend je grafické vývojové prostředí pracující se značkovacím jazykem XAML, o němž již byla řeč v podkapitole 2.1.1. Tento nástroj slouží k vytvoření a úpravě prezentační vrstvy webových aplikací vyvíjených pomocí technologie Silverlight nebo WPF. Jedná se o nástroj pro zpracování grafické části aplikace, proto není nezbytně nutné aby designér byl zároveň programátorem. Designér pouze navrhne vzhled celé aplikace a aplikační logika je implementována programátorem.

Práce s tímto nástrojem je podle mého názoru velmi intuitivní. Veškeré ovládání probíhá metodou *drag-and-drop*, kdy designér vybírá ze seznamu dostupných ovládacích prvků vhodný prvek a umísťuje jej na plátno. Na plátně má pak možnost seskupovat prvky do několika různých skupin, které umožňují definovat vlastnosti pro všechny prvky skupiny najednou. Pro každý prvek nebo skupinu prvků umístěných na plátně lze dále definovat velké množství atributů, lze například nastavit barvu písma, pozadí, rozměry objektu, způsob ohraničení, průhlednost, efekty, apod. Všechny nedefinované vlastnosti objektu jsou uloženy v příslušném XAML souboru. Ukázkou prostředí Microsoft Expression Blend 3 lze nalézt na obrázku 2.1.

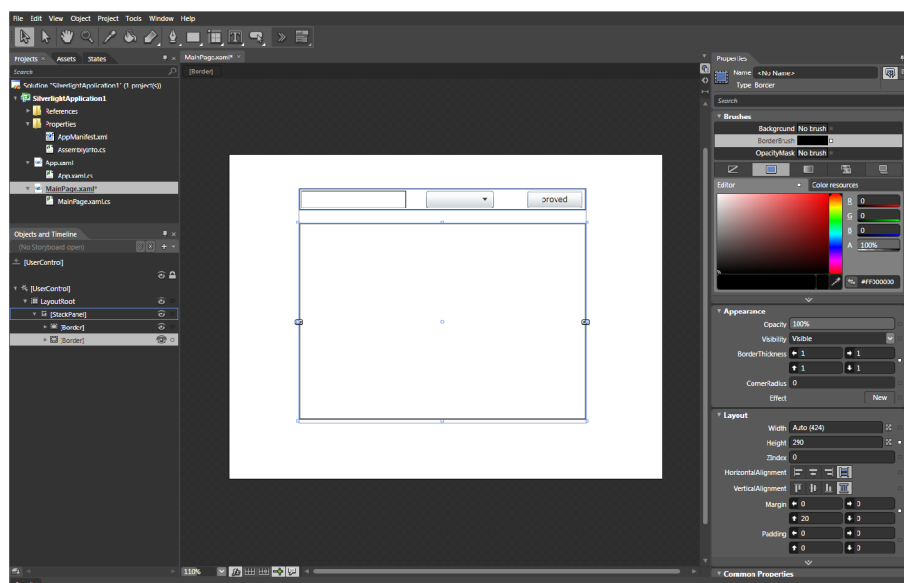
Velkým přínosem poslední verze je funkcionality nazvaná *Sketch Flow*. Tato funkce umožňuje vytvářet a modelovat stránky včetně navigace mezi nimi. Při představení grafického návrhu aplikace zákazníkovi může tento aspekt hrát velmi významnou roli a každý zadavatel projektu tuto možnost jistě rád uvítá. Za zmínku dále stojí možnost přímého importu obrazových souborů typu Adobe Photoshop nebo Adobe Illustrator. Samozřejmostí je možnost přepínání mezi tvorbou designu aplikace a editováním XAML souboru s plnou podporou *Intellisense* [10].

Expression Blend 3 lze stáhnout ze stránek Microsoftu<sup>8</sup>, ke stažení je však pouze trial verze. Plnou verzi lze zakoupit jen jako součást balíčku vývojových nástrojů pod označením Expression Studio 3<sup>9</sup>, který obsahuje kromě Blendu další 3 nástroje – Expression Web 3, Expression Design 3 a Expression Encoder 3. Při vývoji Silverlight aplikace, která je předmětem mé bakalářské práce budu používat studentskou licenci Expression Studio 3 staženou z webu MSDNAA.

<sup>7</sup>[http://msdn61.e-academy.com/elms/Storefront/Home.aspx?campus=vut\\_fit](http://msdn61.e-academy.com/elms/Storefront/Home.aspx?campus=vut_fit)

<sup>8</sup><http://www.microsoft.com/expression/try-it/default.aspx#PageTop>

<sup>9</sup>[http://www.microsoft.com/expression/products/Studio\\_Overview.aspx](http://www.microsoft.com/expression/products/Studio_Overview.aspx)



Obrázek 2.1: Ukázka grafického rozhraní aplikace Microsoft Expression Blend 3.

V současné době Microsoft uvolnil ke stažení betaverzi s označením Expression Blend 4, která přináší velmi silnou podporu vývoje aplikací pro komunikátory Windows Phone 7. Samozřejmě by měla být také podpora Silverlight 4 a WPF 4.

# Kapitola 3

## XML

Následující kapitola se věnuje značkovacímu jazyku XML, popisuje jeho základní rysy a vývoj. Na konci kapitoly lze nalézt příklady vhodného i nevhodného použití tohoto jazyka.

### 3.1 Charakteristika jazyka

Jazyk XML (Extensible Markup Language) je rozšířitelný značkovací jazyk, schválený konsorciem W3C roku 1998. Při jeho vytváření si členové skupiny W3C stanovili celkem 10 cílů, které lze najít na jejich webu<sup>1</sup>. Mezi nejzajímavější patří tyto:

- XML by mělo být přímo použitelné na internetu.
- Dokumenty XML by měly být pro člověka snadno čitelné a dostatečně jasné.
- Vytváření dokumentů XML by mělo být snadné.
- Počet volitelných částí XML by měl být naprosto minimalizován, v ideálním případě by měl být nulový.

Jazykem XML lze popsat libovolný typ dokumentu. Základní vlastnosti tohoto jazyka demonstruje následující příklad, který implementuje seznam knih obsahující jedinou knihu.

```
<?xml version="1.0"?>
<inventar>
  <kniha typ="odborna">
    <nazev>Predpoved pocasi</nazev>
    <autor>Radek Gajdusek</autor>
    <cena>
      <dolar>10.00</dolar>
      <kc>190.00</kc>
    </cena>
    <stran>235</stran>
  </kniha>
</inventar>
```

Základní stavební jednotkou každého XML dokumentu jsou elementy. Příkladem takového elementu je například `<inventar>`. Jedná se o tzv. root element, všechny další jsou pak jeho potomky. Elementy mohou dále obsahovat atributy, příkladem elementu s atributem je `<kniha typ="odborna">`, který nese informaci o tom, že kniha patří do kategorie

---

<sup>1</sup><http://www.w3.org/TR/2008/REC-xml-20081126/#sec-origin-goals>

odborných. Atributy musí být vždy ohraničeny z obou stran uvozovkami. Neexistuje však žádné jasně definované pravidlo, které by určovalo, za jakých okolností je vhodnější použít atribut elementu nebo samostatný podelement elementu `<kniha>`.

Na uvedeného příkladu je vidět jedna ze základních vlastností jazyka XML a to ta, že dokument vytvořený v tomto jazyce tvoří hierarchickou strukturu. Výhodou jazyka XML je, že nedefinuje žádnou pevnou množinu značek (jako je tomu například u jazyka HTML), ale dává možnost vývojářům a autorům dokumentů definovat vlastní elementy, které pro danou úlohu právě potřebují. Na druhou stranu je gramatika XML dokumentů velmi striktní a vyžaduje, aby všechny elementy měly počáteční a ukončovací značku a byly správně zanořeny.

Povolené značky pro daný typ dokumentu jsou definovány pomocí definice typu dokumentu (DTD<sup>2</sup>). Tato specifikace dokumentu definuje pravidla pro místo a způsob použití jednotlivých značek. Dokumenty jsou pak porovnávány oproti DTD a je určována jejich platnost, případně neplatnost vzhledem k referenčnímu DTD. Firma Microsoft vyvinula svůj vlastní formát pro definici typu dokumentu, který nese označení XSD<sup>3</sup>. Tento formát je mírně komplikovanější co se týče zápisu a čitelnosti, ale v zásadě se jedná jen o jinou notaci sloužící ke stejnému účelu. Pokud se dokument neshoduje s porovnávanou definicí, označujeme jej za neplatný vzhledem k referenčnímu DTD, resp. XSD. V současné době se používají obě definice XML dokumentů.

## 3.2 Vývoj jazyka

Počátek vývoje tohoto jazyka je datován do roku 1996. Je následníkem jazyka SGML, který našel uplatnění v americké armádě a vládě USA pro své efektní způsoby správy rozsáhlých dokumentací. Z tohoto jazyka se vyvinul jazyk HTML, který je podmnožinou SGML a nedisponuje tedy všemi jeho možnostmi. Tento jazyk je výborný pro vytváření webových stránek, ale pro jakékoliv další uplatnění, jako například přenos dat mezi nekompatibilními databázemi, zcela nevhodný. Proto začala skupina okolo Johna Bosaka pracovat na „odlehčené“ verzi SGML, která spatřila světlo světa roku 1998 jako první verze XML 1.0. Tento jazyk získal ihned velkou popularitu mezi vývojáři a je používán dodnes.

Postupně vznikly další standardy odvozené od standardu XML 1.0 jako XML namespaces (oddělené bezkonfliktní použití značek v dokumentu), jazyk pro stylové šablony XSLT (XSL Transformations). Později vzniklo také jednotné rozhraní (tzv. API) pro přístup k XML dokumentům z programů psaných v jazycích Java, Javascript a C++. Jelikož nebyla API jednotlivých analyzátorů kompatibilní mezi sebou bylo vytvořeno jednoduché API pro XML zvané SAX, které se bylo později roku 2000 rozšířeno verzí SAX2 [7].

Vývoj pokračuje dále směrem rozšiřování specifikace a vzniká jazyk XHTML, který je redefinicí jazyka HTML a měl by tento jazyk do budoucna nahradit. Hierarchii jazyků rodiny SGML znázorňuje obrázek 3.1.

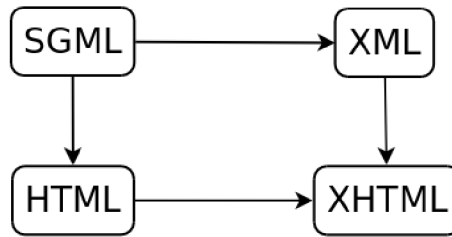
## 3.3 Použití XML

Tento jazyk je multiplatformní a univerzální, což umožňuje jeho použití pro široké spektrum účelů a bez omezení na různých platformách. Mnoho aplikací využívá pro čtení dokumentu XML analyzátor, který dokument rozdělí na jednotlivé elementy, části a atributy a předá

---

<sup>2</sup>DTD – Document Type Definition

<sup>3</sup>XSD – XML Schema Definition



Obrázek 3.1: Schéma vývoje jazyků rodiny SGML.

je dále aplikačnímu programu. Pokud analyzátor narazí na porušení pravidel jazyka XML, ohlásí tuto chybu aplikaci a čtení je ukončeno.

Mnoho lidí se domnívá, že XML je programovací jazyk, což není pravda. Pro XML neexistuje žádný kompilátor, který by ze vstupního souboru generoval spustitelný kód. Tento značkovací jazyk nefunguje ani jako přenosový síťový protokol. Data ve formátu XML lze přenášet po síti, ale jen za použití HTTP, FTP nebo jiného síťového protokolu. Rovněž jazyk XML není databází a neměl by být jako databáze používán. Lze však XML data do databáze ukládat, ale databáze není v žádném případě dokumentem v XML.

V dnešní době je tento jazyk hojně používán v aplikacích, které distribuují informace dalším skupinám uživatelů. Příkladem je i má bakalářská práce, která čerpá data s předpovědí počasí ze zdrojů společností zabývajících se předpovědí. Nachází však uplatnění i u hudebníků, kteří jej využívají ke znázornění notového zápisu – MusicXML [16]. Pod označením Office Open XML používá jazyk XML pro uložení dokumentů také kancelářská sada Microsoft Office od firmy Microsoft. Jedinou oblastí, ve které nelze XML použít jsou multimediální data (fotografie, zvukové záznamy, videa, ...).

## Kapitola 4

# Mashup

Obsahem kapitoly je seznámení s pojmem mashup a aplikacemi odvozenými od tohoto pojmu. Dále je věnován prostor novému směru v oblasti internetu a webových aplikací, který se označuje jako Web 2.0 a je uvedena jeho souvislost s mashup aplikacemi. Kapitola obsahuje také zamyšlení nad budoucností těchto aplikací. V závěru jsou uvedeny dva editory, pomocí nichž lze tvořit mashup aplikace a autorovy postřehy při práci s nimi.

### 4.1 Objasnění pojmu a příklady aplikací

Slovo mashup původně označovalo hudební nahrávku, která se skládala ze dvou nebo více písní spojených dohromady. Toto označení se postupem času rozšířilo do oblasti počítačů, kde je synonymem pro webovou stránku nebo aplikaci, která kombinuje data z více než jednoho zdroje a kumuluje je na jednom místě [5].

Jednou z nejpopulárnějších mashup aplikací je Google Maps. Tato aplikace shromažďuje na jednom místě obrovské množství údajů, které jsou zobrazovány na mapě světa. Uživatel má tak například možnost vyhledávat nejbližší restaurace nebo zastávky MHD. Tato webová aplikace rovněž umožňuje zobrazit fotografie daného místa nebo uskutečnit virtuální procházku zvoleným městem.

Další zajímavou mashup aplikací je Weather Bonk. Jedná se o webovou stránku, která používá jako podklad právě mapy z Google Maps. Tato aplikace zobrazuje předpověď počasí na mapě světa doplněnou o možnost zobrazení aktuálního počasí pomocí dostupných online kamer.

Poslední aplikací, o které bych se rád více zmínil, je Flash Earth. Jak již název napovídá, tato stránka používá technologii Flash a shromažďuje mimo několika různých map světa také pravidelně aktualizované snímky z družic NASA s možností zoomování v těchto záběrech.

Uvedené aplikace se nacházejí na těchto adresách:

- Google Maps  
<http://maps.google.com>
- Weather Bonk  
<http://www.weatherbonk.com/>
- Flash Earth  
<http://www.flashearth.com/>



Na internetu lze nalézt nespočet dalších mashupů. Výše uvedené aplikace byly vybrány na základě statistiky oblíbenosti u uživateli dostupné na této stránce [13]. Uvedený zdroj obsahuje rozsáhlou databázi mashup aplikací včetně internetových odkazů s možností jejich filtrování na základě kategorií.

## 4.2 Web 2.0

Poprvé se označení Web 2.0 výrazněji objevilo v září roku 2005, kdy Tim O'Reilly publikoval svůj článek „What is web 2.0“ (Co je web 2.0) a jako první jasně definoval rozdíly mezi v tu chvíli již starými a nově vznikajícími webovými stránkami [12].

Do té doby všechny společnosti svá nashromážděná data, jako například mapové podklady, nikde na internetu zdarma nevystavovaly ani nedávaly zdarma k dispozici vývojářům pro jejich aplikace. Na počátku prvního desetiletí roku 2000 se však situace začala postupně měnit a objevily se první snahy společností Yahoo, Google a Microsoft zpřístupnit svá mapová data co nejširší skupině uživatelů, kteří by je mohli dále využít. Systémy se tak otevřely pro veřejnost a začaly vznikat první aplikace využívající tato data. Každá firma měla své vlastní API, které umožňovalo jednodušší práci s jejich zdroji, což vzbudilo velký zájem u uživatelů, neboť šlo o něco zcela nového a nepoznaného – svobodný přístup k datům.

Začalo se také novým způsobem nahlížet na data přidaná uživatelem. Uživatel nebyl již pouhým návštěvníkem, ale stal se nyní tím, kdo tvoří obsah webu, internetové stránky tak již nejsou natolik závislé na příspěvcích vlastníka stránky, ale naopak přispěvateli se stává celá internetová komunita. Jiné stránky doslova žijí pouze z dat vložených uživateli, příkladem je videoportál YouTube nebo obrovská online encyklopedie Wikipedie. Jednotlivým videím na Youtube lze přiřadit prakticky libovolné značky tzv. tagy, které usnadňují jiným uživatelům vyhledávání a v neposlední řadě také práci vyhledávacím algoritmům. I to však nese jistá úskalí. Vlastník stránek ztrácí plnou kontrolu nad vloženým obsahem a je nutné data uživatelů kontrolovat, zda neporušují zákony dané země. Klasickým příkladem jsou pornografická videa na portálu Youtube, která jsou sice zakázána, ale občas se nějaká objeví, jsou však velmi rychle provozovatelem stažena.

Stále více finančních prostředků se vyčleňuje na reklamu takto vzniklých stránek, protože weby s miliónem přístupů denně se stávají lukrativním prostorem pro inzerenty, z čehož profitují jejich provozovatelé. Uvolnění dříve placených informací se tak nestává nevýhodou, ale naopak se ukazuje, jak prozíravý tah to byl [2].

Dříve web sloužil k získávání informací a pouze „ke čtení“. Dnes se uživatelé aktivně zapojují do jeho tvorby, podílejí se na obsahu, tvoří web a sdružují se do sociálních sítí jako je Facebook nebo Twitter. Musím však také zmínit to, že nový Web 2.0 nám nabídl sice netušené možnosti, ale také nás připravil o soukromí. Lidé v dnešní době neuváženě zveřejňují často a rádi osobní informace v sociálních profilech a vůbec si neuvědomují, že mohou být tato data lehce zneužita. Typickým příkladem je Gmail, který svou reklamu v e-mailu zobrazuje na základě obsahu vašich zpráv. Víme jistě, že je ochráněno naše soukromí a nikdo nemá přístup k našim e-mailům? Nezbývá nám než jen doufat a spoléhat se na to, že tomu tak opravdu je.

Postupem času se ukázalo, jak Tim O'Reilly předběhl svou dobu a jasně definoval něco, co je pro nás v dnešní době samozřejmostí. Svět bez Google Maps, YouTube, Flickr či Amazonu si většina z nás dokáže již těžko představit.

## 4.3 Souvislost mashup aplikací s Web 2.0

Oddělení dat na webových stránkách od jejich vlastní reprezentace přineslo do té doby netušené možnosti. Vznikly webové stránky, které jsou snadno čitelné jak pro uživatele, tak pro stroje a software, což umožnilo ve velké míře rozvoj mashup aplikací.

Vlastníci dat často vývojářům poskytují dobře zdokumentovaná API pro přístup k datům, jejichž prostřednictvím dávají svá data k dispozici. Velmi populární jsou také různé RSS kanály, které zpravidla používají formát XML pro reprezentaci dat. Vývojáři jsou však velmi vynalézaví a jsou schopni webovou stránku, která nedisponuje nějakou z uvedených možností pro snadné získání dat, rozparsovat a získat tak potřebná data.

Všechny tyto aspekty vývoj mashup aplikací velmi zjednodušují. Aplikace, jejichž vývoj dříve znamenal velké množství finančních prostředků a vyžadoval úsilí několika programátorů, lze dnes vytvořit během chvíle za použití výše uvedených prostředků.

## 4.4 Budoucnost mashup aplikací

Mashup aplikace se proto stávají velmi oblíbenými a jejich počet se neustále zvyšuje. Dle mého názoru tomu v budoucnu nebude jinak. Pro uživatele je velmi pohodlné, když může nalézt co nejvíce informací na jednom místě a nemusí přitom opustit jednu webovou stránku. Tento trend také dokazuje oblíbenost různých mapových mashup aplikací, na nichž kromě mapy najdete také informace o firmách ve vašem okolí, otevírací době obchodů a restaurací, ale také informace o památkách a zajímavých místech. Mashup aplikace jsou podle mě cestou, jakou se bude web v budoucnu ubírat společně se sociálními sítěmi.

## 4.5 Mashup editory

Tvorbu mashup aplikací výrazně zjednodušují dostupné nástroje, které lze nalézt na internetu. V této části se zmíním o dvou, z nichž první nese název Yahoo pipes a funguje jako webová stránka, zatímco druhý nástroj zvaný OpenKapow RoboMaker vyžaduje stažení a instalaci na klientský počítač.

Především aplikace Yahoo pipes je v zahraničí mezi uživateli internetu poměrně oblíbená, u nás již takovým hitem není. Přisuzuji to skutečnosti, že tento nástroj není lokalizován do češtiny a také tomu, že uživatelé o něm příliš nevědí. Osobně jsem se s uvedenými nástroji setkal teprve nyní v souvislosti s mou bakalářskou prací.

U obou nástrojů se pokusím pro srovnání o vytvoření mashup aplikace, která bude čerpat data ze dvou RSS zdrojů se sportovními zprávami. Použiji zdroje online deníků *lidovky.cz* a *idnes.cz*. U prvního zdroje provedu zobrazení zpráv týkajících se hokeje, zatímco ze druhého vyfiltruji všechny zprávy mimo hokejový obsah. Vyfiltrované zprávy poté spojím dohromady a provedu jejich seřazení podle data vydání od nejnovějších po nejstarší.

### 4.5.1 Yahoo pipes

Yahoo pipes<sup>1</sup> je bezplatný nástroj od společnosti Yahoo!, který umožňuje vytvářet mashup aplikace kombinací různých datových zdrojů přímo v prostředí prohlížeče. Použití zmíněného online nástroje je podmíněno zřízením účtu na Yahoo!. Vytvořené mashupy lze pak pod tímto účtem zveřejňovat ostatním uživatelům nebo je používat pro soukromé účely.

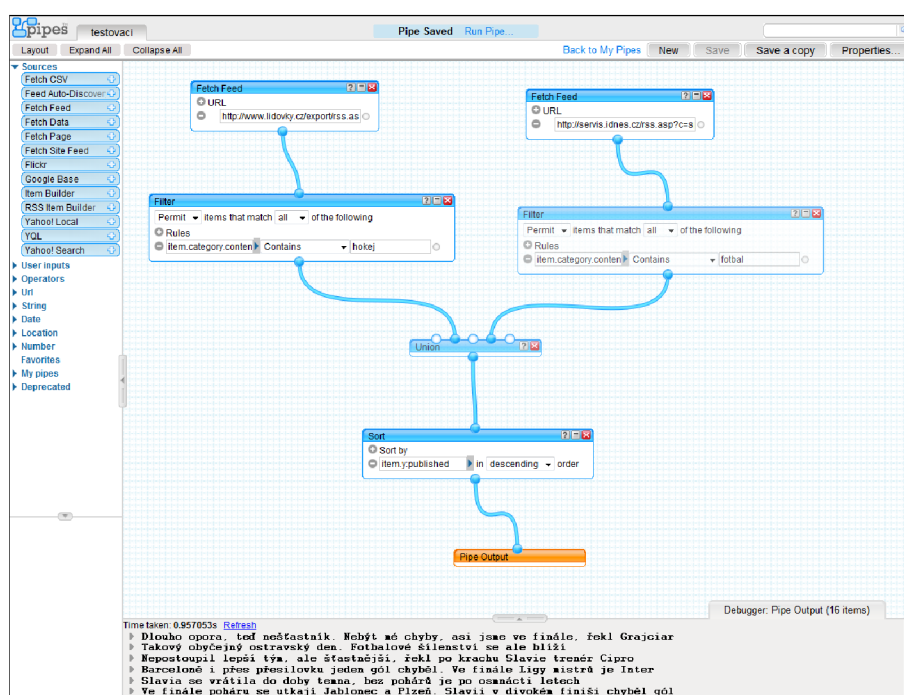
---

<sup>1</sup><http://pipes.yahoo.com/pipes/>



Práce se tímto nástrojem je velmi intuitivní a probíhá metodou *drag-and-drop* v prostředí prohlížeče. Uživatel si z nabídky umístěné v levé části stránky vybere komponentu, kterou jednoduše přetáhne na „plátno“, kde dochází ke spojování jednotlivých komponent pomocí rour tzv. *pipes*. Při tvorbě mashupu je kromě datových zdrojů možné použít také různé filtry, provádět transformaci, řazení a spoustu dalších užitečných operací s načtenými daty. Výstup aplikace je vidět ihned v debuggeru, který je umístěn v dolní části stránky a usnadňuje kontrolu výstupu v průběhu tvorby mashupu. Aplikace je možné jedním kliknutím uložit a během okamžiku zobrazit na webu.

Vytvoření testovací aplikace mi trvalo několik minut, práce s tímto online nástrojem je velmi jednoduchá a umožňuje vytvářet aplikace mnohem větších rozměrů než jsem uvedl jako testovací příklad. Byl jsem příjemně překvapen jeho jednoduchostí a určitě se k tomuto nástroji někdy v budoucnu vrátím.



Obrázek 4.1: Ukázka prostředí aplikace Yahoo pipes.

## 4.5.2 OpenKapow RoboMaker

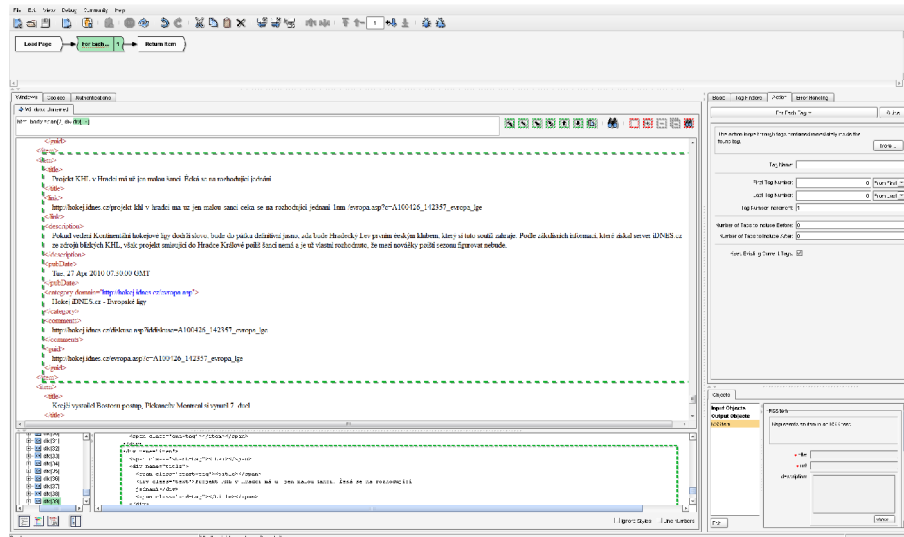
Druhým nástrojem, který se věnuje tvorbě mashupů, je OpenKapow RoboMaker<sup>2</sup> společnosti Kapow Technologies. Jedná se o opensource aplikaci napsanou v jazyce Java, která však vyžaduje stažení většího souboru do počítače a jeho následnou instalaci.

Při vytváření mashupu jsou používány tzv. *roboti*, kterým je specifikován zdroj dat ve formě URL, který má být tímto robotem zpracován. Poté je nutné definovat oblast HTML stránky a určit způsob, jakým má být zpracována. Robotu lze také definovat operace, které má s načtenými daty provést jako například řazení podle různých kritérií, filtrování, transformaci, apod. Uvedený nástroj poskytuje uživateli mnohem větší možností než Yahoo

<sup>2</sup><http://allmyapps.com/app/windows-xp/openkapow-robomaker-robot-maker-for-openkapow>

pipes, ale práce s ním není již natolik intuitivní a podobá se spíše programování. Nástroj obsahuje rovněž debugger, který velmi usnadňuje ladění tvořeného mashupu.

Seznámení s nástrojem RoboMaker mi trvalo poměrně dlouhou dobu a vytvoření testovací mashup aplikace zabralo téměř dvě hodiny. Z mého pohledu je tento nástroj po běžné uživatele nepraktický především z důvodu nutnosti instalace a své složitosti. Pokročilým uživatelům z řad programátorů však umožní po důkladném zaškolení vytváření rozsáhlých mashup aplikací za použití široké škály prostředků.



Obrázek 4.2: Ukázka prostředí aplikace OpenKapow RoboMaker.

## Kapitola 5

# Předpověď počasí

Následující kapitola se blíže zabývá problematikou předpovědi počasí, objasňuje základní pojmy a zmiňuje důležité historické mylníky týkající se tohoto oboru lidské činnosti. Závěrem dochází k rozdělení předpovědí podle několika kritérií a klasifikaci předpovědí používaných pro potřeby této bakalářské práce.

### 5.1 Základní pojmy

Studiem atmosféry se zabývá věda nesoucí název *meteorologie*. Tato vědecká disciplína se mimo jiné specializuje na děje a jevy probíhající v atmosféře. Jedním z těchto jevů je také počasí. Odhad budoucího stavu počasí na základě aplikace meteorologických poznatků, měření a záznamů z družic se nazývá *předpověď počasí*.

Většina zemí má vlastní organizace, které se studiem počasí a vydáváním předpovědí zabývají. V naší republice je hlavní autoritou Český hydrometeorologický ústav, na Slovensku je jeho obdobou Slovenský hydrometeorologický ústav. Zmíněné ústavy mají jako jediné v republice právo kromě vydávání předpovědi počasí vydávat také výstrahy a varování, například o blížící se vichřici nebo sněhové bouři. Předpovědi počasí se zabývají i soukromé organizace nebo nadšenci, kteří zpravidla své závěry pozorování a předpovědi prezentují na vlastních webových stránkách.

V dnešní době hraje správná a přesná předpověď počasí velmi významnou roli. Například v oblasti letectví na ní závisí lidské životy, v zemědělství ovlivňuje rozhodování zemědělců, ale dokáže také včasnou identifikací blížící se bouře, hurikánu nebo monzunu zachraňovat lidské životy.

### 5.2 Historie předpovídání počasí

Již ve starověku byli lidé odkázáni na rozmary počasí. V období sucha hrozil hladomor, neboť se jim nedostávalo potravin ani lovné zvěře. Naopak v době příznivých povětrnostních podmínek byla úroda bohatá a potraviny dostatek. Starověké národy věřily, že počasí ovlivňuje nálada bohů, a proto konali různé slavnosti, aby se bohům zavděčili. V starověké Číně se snažili naopak o racionální vysvětlení, pozorovali oblohu a na základě tohoto pozorování vytvořili kalendář rozdělený na 24 období, přičemž pro každé období definovali počasí. V Indii dokonce zaznamenávali množství srážek prvními srážkoměry. Na základě pozorování oblohy začaly vznikat v této éře lidstva první předpovědi.

Prvním učencem, který shrnul dosavadní poznatky o počasí byl Aristoteles ve své knize *Meteorologica*. Právě v této době lze nalézt původ slova meteorologie. Na Aristotela navázal o několik let později jeho žák knihou *Znaky počasí*, v níž byly popsány příznaky bouřky, deště a větru. Pozadu nezůstali ani římsí učenci, kteří shrnuli záznamy, pozorování a pověry z Egypta, Řecka a Říma do díla nazvaného *Naturalis historia*. Stále se však jednalo o nepodložené závěry a předpovědi získané pouhým pozorováním.

Obdobím největšího rozkvětu meteorologie lze považovat renesanci. V tomto období bylo zkonstruováno několik meteorologických přístrojů, které používáme dodnes. Galileo Galilei vynalezl teploměr, Leonardo da Vinci stojí za objevem vlhkoměru, nejdůležitějším objevem však byl bezesporu tlakoměr Evangelisty Torricelliho, díky němuž bylo možné na základě změny tlaku předpovídat počasí. Toho si také všiml Blaise Pascal a definoval jeden ze základních zákonů meteorologie.

Po objevení těchto vynálezů zažívala meteorologie v období 17. a 18. století obrovský rozmach. Byla definována Celsiova a Fahrenheitova stupnice, byly budovány první povětrnostní stanice a naměřené výsledky byly mezi vědci sdíleny, což přineslo lidstvu na tu dobu poměrně přesné informace o počasí.

Počátek 20. století je považován za počátek současné meteorologie. Začaly se objevovat předpovědi pro různá místa v různých časových pásmech na základě matematických modelů. Měření počasí se automatizuje, budují se meteorologické stanice, u nichž není nutná bezprostřední obsluha a jsou nově využívány údaje z družic a radarů. První meteorologická družice byla vypuštěna v dubnu roku 1960. Oproti těm dnešním byly první družice velmi primitivní, dokázaly sledovat jen pohyb oblačnosti a poskytovat údaje střediskům na Zemi. Díky tomuto pokroku bylo možné závčas identifikovat bouři mířící z moře na pevninu a lépe se na ni připravit.

V současné době meteorologie využívá k předpovědi počasí a jeho pozorování znalosti nabyté za celou dobu vývoje této disciplíny. Základem kvalitní předpovědi jsou nejnovější technologie, které jsou základním stavebním kamenem moderní meteorologie. K analyzování naměřených hodnot získaných z družic a meteorologických stanic a modelování dalšího průběhu počasí jsou využívány výkonné superpočítače. Velmi důležitou roli hraje také mezinárodní spolupráce meteorologů na celém světě, díky internetu prakticky v reálném čase. V dnešní době tak není problém předpovídat počasí od několika hodin dopředu po několik sezón. Meteorologické předpovědi ovlivňují dnes rozhodování prakticky všech lidí na naší planetě a jsou nezbytným základem fungování moderní společnosti [3].

### 5.3 Druhy vydávaných předpovědí

Každou vydanou předpovědi počasí lze klasifikovat třemi kritérii [4]. Prvním důležitým kritériem je druh předpovědi, který se dělí do následujících tří kategorií:

- **Všeobecné** – jedná se o standardní předpověď se stavem oblačnosti, teplotou, apod., která může být doplněna o upozornění na nebezpečný jev.
- **Speciální** – vydávají se pro konkrétní oblast lidské činnosti a obsahují informace podstatné pro daný obor.
- **Výstražné informace** – upozorňují na nebezpečné meteorologické a hydrologické jevy.

Z hlediska rozsahu platnosti je lze dále rozdělit do čtyř druhů:

- **Velmi krátkodobá předpověď** – s platností maximálně 12 hodin od vydání.
- **Krátkodobá předpověď** – s platností až 36 hodin od vydání.
- **Střednědobá předpověď** – s předpovědí až na 8 dní dopředu ode dne vydání.
- **Dlouhodobá měsíční předpověď** – s dobou platnosti 30 dnů.

Třetím kritériem je rozsah působnosti vydané předpovědi:

- **Bodová předpověď** – s předpovědí počasí pro danou lokalitu, např. letiště.
- **Liniová předpověď** – pro vybrané úseky komunikací, zpravidla dálnice.
- **Regionální předpověď** – pro větší územní celky jako jsou kraje nebo okresy.
- **Předpověď pro celé území** – zahrnující oblast celého státu.

Pro potřeby mé bakalářské práce nás budou zajímat všeobecné regionální předpovědi počasí pro jednotlivé kraje se střednědobou platností. V případě výstrah se bude jednat o výstražné informace, krátkodobého až střednědobého charakteru vydávané pro konkrétní kraj nebo okres, případně celé území České republiky.

# Kapitola 6

## Analýza

Kapitola se věnuje rozboru zadání bakalářské práce a shrnuje důvody, které autora vedly k volbě tohoto zadání. Dále zde lze nalézt sumarizaci dostupných zdrojů s předpovědí počasí, výstražnými informacemi a varováními na internetu, rozsah poskytovaných dat a odkaz na stránky poskytovatelů. V závěru je zodpovězena otázka, zda je možné na základě dostupné analýzy zdrojů vytvořit požadovanou mashup aplikaci splňující zadání.

### 6.1 Zadání práce a motivace

Úkolem mé bakalářské práce je vytvoření webové mashup aplikace, která bude zobrazovat informace s předpovědí počasí pro Českou a Slovenskou republiku na několik dní dopředu. Aplikace má pracovat se zdroji ve formátu XML. Pokud se podaří získat data s výstrahami a varováními pro uvedené státy, budou tyto informace také její součástí. Mashup aplikace bude napsána pomocí technologie Silverlight, která byla blíže popsána v kapitole 2.

K výběru tohoto zadání mě vedlo hned několik důvodů. Tato problematika mě zaujala na první pohled, neboť jako každý člověk tak i já, neustále sleduji aktuální stav počasí a musím bohužel konstatovat, že na stránkách všech poskytovatelů mi vždy nějaká informace chybí nebo je nutné se k ní dlouze proklikat.

Mám-li být konkrétní, často potřebuji vyhledat předpověď pro konkrétní místo, ale domácí zdroje touto možností nedisponují, možnost přímého zadání názvu města zkrátka chybí, proto jsem nucen zvolit zahraniční portály, které možnost vyhledání předpovědi na základě zadaného města umožňují.

Také jsem zatím nenarazil na portál, který by zobrazoval více informací a byl tak skutečnou mashup aplikací v pravém slova smyslu. Většinou se poskytovatelé zaměřují na konkrétní oblast, například Českou republiku a uživatel musí pro informace o počasí na Slovensku navštívit jinou stránku. Pokud jej zajímají navíc vydané výstrahy a varování musí si zobrazit dva další weby, pro každou republiku jeden. Když vezmeme v potaz, že jsme na čísle čtyři, kladu si otázku, zda by nebylo vhodnější mít všechny informace dostupné na jednom místě, což mě přivádí zpět k důvodu, proč jsem si zvolil právě zadání, jehož cílem je mashup aplikace s předpovědí počasí.

Z výše uvedených důvodů se domnívám, že má práce bude mít smysl, nebude se jednat pouze o další stránku s předpovědí počasí, ale přinese uživatelům hledané informace komplexně na jednom místě.

Zaujalo mě, že předmětem této práce je naprogramování aplikace pomocí technologie Silverlight, se kterou nemám žádné zkušenosti a rád bych se s ní naučil pracovat. Vyzkoušení



práce s XML dokumenty je pro mě rovněž velmi lákavě, neboť jsem se s tímto, poslední dobou hojně využívaným, formátem po programovací stránce příliš nesetkal, proto věřím, že tato práce bude pro mě po všech stránkách přínosem.

Před započítím fáze návrhu a následného vývoje aplikace bylo nutné zmapovat zdroje na internetu, poskytující data s předpovědi a výstrahami ve formátu XML. V případě, že by tyto zdroje neexistovaly, bylo by nutné předmět práce přehodnotit nebo zvolit jiné zadání. Dostupnými zdroji na internetu se zabývá podkapitola 6.2.

## 6.2 Zmapování dostupných zdrojů

Při hledání jsem se zaměřil kromě českých a slovenských zdrojů také na zdroje zahraniční. Souhrn obsahuje všechny nalezené zdroje roztríděné do tří kategorií – české, slovenské a zahraniční.

Ke každému zdroji je připojen krátký komentář, odkaz na internetové stránky a nabídku předpovědi, případně také ukázka struktury a obsahu poskytovaného XML souboru.

Výstražné informace a varování pro daný stát mohou poskytovat pouze národní authority, které mají k tomuto oprávnění, ostatní organizace mohou tyto informace pouze přebírat a prezentovat, nemohou je však přímo vydávat. Z tohoto omezení vyplývá, že žádný ze zahraniční zdroj nemůže poskytovat výstražné informace a varování pro Českou nebo Slovenskou republiku.

### 6.2.1 České zdroje

Pro Českou republiku byly nalezeny celkem čtyři vhodné zdroje, z nichž prvně uvedený poskytuje kromě předpovědi počasí také výstražné informace a varování. Data jsou tímto zdrojem poskytována zdarma pouze pro studentské účely. Ostatní uvedené zdroje poskytují data jako placenou službu s výjimkou společnosti Meteocentrum.cz.

#### Český hydrometeorologický ústav (ČHMÚ)

Jedná se o hlavní autoritu v oblasti předpovědi počasí a vydávání výstražných informací a varování (systém nese označení „Systém integrované výstražné služby (SIVS)“). ČHMÚ přímo na svém webu možnost distribuce předpovědi a výstrah ve formátu XML neuvádí, nicméně na základě mého e-mailového dotazu mi bylo sděleno, že tato možnost existuje, ale jedná se především o placená data. Tato data jsou ochotni poskytnout zdarma pro studentské účely oproti čestnému prohlášení studenta.

- Webové stránky:  
<http://www.chmi.cz/>
- Nabídka dat:  
<http://www.chmi.cz/meteo/om/info.html#ion>
- Popis SIVS:  
<http://www.chmi.cz/meteo/om/sivs.html>
- Ukázka XML s výstrahami:  
[http://pocasi.chmi.cz/data/doc/2.0/SIVS-PVI\\_2009\\_66.xml](http://pocasi.chmi.cz/data/doc/2.0/SIVS-PVI_2009_66.xml)

## Meteocentrum.cz

Méně známý portál, který však jako jediný na svých stránkách nabízí bezplatnou možnost poskytnutí dat ve formátu XML. Na jejich webu lze také nalézt ukázkou XML kanálu, který obsahuje čtyřdenní předpověď pro všechna krajská města České republiky.

- Webové stránky:  
<http://meteocentrum.cz/>
- Nabídka dat:  
<http://meteocentrum.cz/sluzby/>
- Ukázka XML:  
[http://ragaj.cz/examples/meteocentrum\\_xml.html](http://ragaj.cz/examples/meteocentrum_xml.html)

## Meteopress, s. r. o.

Velmi známý portál s počasím pro Českou republiku. Na jejich webu inzerují možnost poskytnutí dat ve formátu XML, jedná se však o placenou službu, cenu na webových stránkách neuvádějí. Dotázal jsem se na možnost bezplatného poskytnutí dat pro studentské účely, vyžádali si ode mně další informace o mé práci, čímž naše komunikace skončila. Přisuzuji to skutečnosti, že je předmět mé práce nezaujal.

- Webové stránky:  
<http://meteopress.cz/>
- Nabídka dat:  
<http://meteodat.cz/media.php>

## IN-POČASÍ

Tato instituce poskytuje placenou předpověď počasí ve formátu XML. Na jejich stránkách si lze vybrat z několika možných balíčků, jejichž cena se pohybuje v rozmezí od 79 Kč do 1999 Kč v závislosti na rozsahu poskytovaných informací.

- Webové stránky:  
<http://www.in-pocasi.cz/>
- Nabídka dat:  
<http://www.in-pocasi.cz/pocasi-na-web/>
- Ukázka XML:  
<http://www.in-pocasi.cz/pocasi-na-web/xml-ukazky/30mest.xml>

### 6.2.2 Slovenské zdroje

Pro Slovenskou republiku byly nalezeny tři vhodné zdroje, z nichž první poskytuje navíc kromě předpovědi počasí také výstražné informace a varování. U všech uvedených zdrojů se jedná o placené služby a to i v případě použití pro studentské účely. Jediným bezplatným poskytovatelem je společnost IN-POČASIE.



## Slovenský hydrometeorologický ústav (SHMÚ)

Tato instituce je obdobou ČHMÚ pro Slovenskou republiku. Na svých webových stránkách neuvádí žádnou informaci o tom, že by poskytovali dalším subjektům data s předpovědí počasí nebo výstrahami. Během e-mailové komunikace s touto organizací se mi podařilo zjistit, že data poskytují jako placenou službu v libovolném formátu na přání zákazníka, tedy i v XML. Poskytování dat zdarma pro studentské účely neprovozují.

- Webové stránky:  
<http://www.shmu.sk/>

## IN-POČASIE

Obdoba české instituce IN-POČASÍ, poskytují zdarma předpověď pro SR ve formátu XML. V nabídce mají pouze jeden možný formát obsahu s předpovědí pro všechny kraje Slovenské republiky.

- Webové stránky:  
<http://www.in-pocasio.sk/>
- Nabídka dat:  
<http://www.in-pocasio.sk/pocasio-pre-web/>
- Ukázka XML:  
<http://www.in-pocasio.sk/pocasio-pre-web/pocasio.xml>

## Meteo.sk

Jedná se o zdroj, který poskytuje placenou předpověď v rozsahu odpovídajícím typu zakoupeného balíčku. Cena se pohybuje v rozmezí od 0,20 € po 6,64 €. Formát poskytnutí dat je neznámý a ani se mi jej nepodařilo zjistit. Na můj e-mailový dotaz ohledně poskytnutého formátu a možnosti poskytnutí dat zdarma pro studentské účely jsem nedostal žádnou odpověď.

- Webové stránky:  
<http://meteo.azet.sk/>
- Nabídka dat:  
<http://meteo.azet.sk/sluzby/produkty-web>

### 6.2.3 Zahraniční zdroje

Žádný ze zahraničních zdrojů neposkytuje komplexní předpověď pro Českou nebo Slovenskou republiku ani výstražné informace nebo varování pro tyto státy. Zahraniční zdroje disponují pouze možností vyhledat předpověď pro jednotlivá města.

Tyto zdroje zpravidla poskytují pro přístup ke svým datům rozhraní (API). Data poskytují bezplatně prostřednictvím dotazu na webovou adresu pro město, které bylo zadáno jako parametr dotazu společně s kódem státu. Odpovědí na tento dotaz je XML dokument obsahující předpověď pro požadované město.

První dva zdroje obsahují plnohodnotnou předpověď s výhledem na minimálně 4 dny dopředu, poslední uvedený je RSS kanál, který obsahuje data pouze pro aktuální den. Jeho nevýhodou oproti ostatním je také nutnost znát přesný kód požadovaného města.

Podrobnější informace o XML schématu uvedených zdrojů a způsobu získání předpovědi lze nalézt pod odkazem s popisem jejich API.

### Weather Underground

- Popis API:  
[http://wiki.wunderground.com/index.php/API\\_-\\_XML](http://wiki.wunderground.com/index.php/API_-_XML)
- Ukázka XML s předpovědí pro Brno:  
<http://api.wunderground.com/auto/wui/geo/ForecastXML/index.xml?query=Brno,CZ>

### Google

- Popis API:  
<http://blog.emerick.org/2008/05/07/google-weather-api-feed-documentation/>
- Ukázka XML s předpovědí pro Brno:  
<http://www.google.cz/ig/api?weather=Brno,CZ>

### Yahoo

- Popis API:  
<http://developer.yahoo.com/weather/>
- Ukázka XML s předpovědí pro Brno:  
<http://weather.yahooapis.com/forecastrss?w=786869>

## 6.2.4 Zhodnocení zdrojů

Na základě provedené analýzy dostupných zdrojů v podkapitole 6.2 se domnívám, že by nemělo být problémem vytvořit za použití uvedených zdrojů mashup aplikaci splňující zadání. Jediným problémem by mohly být výstražné informace a varování, pro něž je zdroj dostupný pouze v případě České republiky. V ostatních případech je pro každý stát dostupný alespoň jeden bezplatný zdroj s předpovědí počasí, případně se nabízí také možnost použití některého z placených zdrojů.

# Kapitola 7

## Návrh

Následující kapitola se věnuje problematice návrhu funkcionality mashup aplikace a jejímu grafickému uživatelskému rozhraní.

V první části je zdůvodněna volba zdrojů, se kterými bude aplikace pracovat, dále následuje sumarizace zobrazovaných informací a vytvoření vlastního číselníku stavů oblačnosti. Pro jednodušší převod stavů oblačnosti z reprezentace jednotlivých zdrojů do našeho číslování, lze v kapitole nalézt konverzní tabulku.

Design aplikace je popsán pomocí schématu, na kterém jsou vysvětleny hlavní rysy grafického uživatelského rozhraní, včetně podporovaného rozlišení. Nechybí také výčet grafických prvků, jejichž vytvoření bude nutné provést ve fázi implementace.

Závěrem kapitoly se čtenář dozví o iniciativě autora v podobě zřízení vlastní domény a blogu k bakalářské práci.

### 7.1 Funkcionalita aplikace

Při návrhu funkcionality výsledné mashup aplikace jsem se snažil zohlednit všechny nedostatky, kterými podle mého názoru trpí ostatní dostupné aplikace na internetu. Možnosti uživatele při práci s mou aplikací bych shrnul do čtyř hlavních bodů:

1. Zobrazení jednodenní předpovědi souhrnně pro oba státy s možností výběru libovolného dne z nabídky v horizontu čtyř dní dopředu.
2. Zobrazení čtyřdenní předpovědi pro libovolné krajské město ČR nebo SR kliknutím na jeho ikonku společně s výstrahou nebo varováním pro kraj, v němž zvolené krajské město leží, pokud je nějaká výstraha nebo varování pro tuto oblast vydána.
3. Vyhledání a zobrazení čtyřdenní předpovědi pro libovolné město ČR nebo SR zadáním jeho názvu.
4. Zobrazení detailů o výstraze nebo varování po najetí na ikonku oznamující dostupnou výstrahu nebo varování.

Mezi vedlejší funkce bude patřit možnost přepnutí aplikace do režimu celé obrazovky, zobrazení nápovědy a informací o programu. Samozřejmostí bude také možnost zobrazení webových stránek každého z poskytovatelů dat.

### 7.1.1 Výběr zdrojů

Jelikož jsem si stanovil za cíl vytvořit aplikaci, která bude čerpat informace z bezplatných zdrojů, zúžil se výběr vhodných zdrojů na několik poskytovatelů. Rozhodování probíhalo na základě poznatků získaných v podkapitole 6.2. V aplikaci budou použity následující čtyři zdroje.

- **Předpověď počasí pro ČR**

Pro Českou republiku byl zvolen zdroj poskytovatele Meteocentrum.cz, který disponuje veškerými potřebnými daty s předpovědí. Uvažoval jsem také o použití zdroje ČHMÚ, ale tuto variantu jsem nakonec zavrhl. Bylo to z toho důvodu, že chci vytvořit aplikaci, která bude čerpat data z co největšího počtu zdrojů. Při použití druhé varianty by byla data s výstrahami i předpovědi pro ČR stahována od stejného poskytovatele a to od České hydrometeorologického ústavu. Díky volbě první varianty bude docíleno různorodosti zdrojů.

- **Předpověď počasí pro SR**

V případě Slovenské republiky již volba zdroje tak jasná nebyla. Společnost IN-POČASIE poskytuje předpověď s teplotou pouze pro celé Slovensko, nikoliv pro jednotlivé kraje této země, jako je tomu u společnosti Meteocentrum.cz. Stav oblačnosti je již definován pro každý kraj zvlášť. Uvažoval jsem proto, že bych data pro SR získával z některého ze zahraničních zdrojů stažením předpovědi pro každé krajské město Slovenska zvlášť. Tato varianta mi však nepřišla jako příliš vhodná z toho důvodu, že by načtení těchto dat neúměrně zatěžovalo linku a pro uživatele s pomalým připojením by mohlo načtení aplikace trvat příliš dlouhou dobu. Po pečlivém zvážení a konzultaci s vedoucím práce jsem se nakonec rozhodl použít zdroj společnosti IN-POČASIE.

- **Výstražné informace a varování**

Původním představou, která počítala se zobrazením výstražných informací a varování pro Českou i Slovenskou republiku, musela být změněna na základě poznatků zjištěných při analýze dostupných datových zdrojů. Jelikož SHMÚ odmítl data poskytnout, budou v aplikaci k dispozici výstražné informace a varování pouze pro Českou republiku. Data budou stahována od hlavní autority pro Českou republiku, kterou je Český hydrometeorologický ústav.

- **Vyhledání předpovědi pro zadané město**

Pro vyhledání předpovědi pro zadané město bylo původně naplánováno použití dvou zdrojů, a sice Weather Underground a zdroje společnosti Google. Primárním zdrojem pro vyhledávání měl být zdroj Googlu, který vrací přesnější předpovědi, ale nedokáže aproximovat předpověď na nejbližší město, pro něhož daty disponuje, pokud pro zadané město předpověď nezná. Doplnovat jej měl zdroj z Weather Underground, který aproximovat dokáže, ale nevrací natolik přesné předpovědi jako Google. Primárně se tedy měla vyhledávat předpověď pomocí Googlu, v případě negativní odpovědi na Weather Underground. V počáteční fázi implementace nastaly komplikace způsobené absencí souboru `crossdomain.xml`<sup>1</sup> v root adresáři všech domén Googlu, a proto bylo nutné tento plán po konzultaci s vedoucím přehodnotit a použít pro vyhledávání pouze zdroj Weather Underground. Více se tomuto problému věnuje podkapitola 8.4.2 v kapitole věnující se implementaci.

---

<sup>1</sup>[http://www.adobe.com/devnet/articles/crossdomain\\_policy\\_file\\_spec.html](http://www.adobe.com/devnet/articles/crossdomain_policy_file_spec.html)

Na základě uvedené volby zdrojů bylo ČHMÚ zasláno čestné prohlášení o neposkytnutí informací jiné osobě a zároveň jsem je požádal o zpřístupnění XML kanálu s výstrahami a varováními pro Českou republiku. Čestné prohlášení je součástí přílohy této práce. Dále jsem kontaktoval společnosti Meteocentrum.cz a IN-POČASIE s žádostí o zpřístupnění jejich zdroje.

Pokud bych uvažoval někdy v budoucnu o vytvoření komerční aplikace, zřejmě bych použil některý z placených zdrojů, který by především pro Slovenskou republiku disponoval teplotní předpovědí i pro jednotlivé kraje. Rovněž by byl také odstraněn problém s chybějícími výstrahami a varováními pro Slovensko.

### 7.1.2 Zobrazované informace

Jak již bylo řečeno výše, předpověď bude zobrazována na čtyři dny dopředu, přičemž aktuální den se počítá za první den předpovědi. Celkem bude k dispozici předpověď pro 13 krajských měst České republiky a 8 krajských měst Slovenské republiky. Seznam zobrazovaných měst se nachází níže.

- **Krajská města ČR**

Brno, České Budějovice, Hradec Králové, Jihlava, Karlovy Vary, Liberec, Olomouc, Ostrava, Pardubice, Plzeň, Praha, Ústí nad Labem, Zlín

- **Krajská města SR**

Banská Bystrica, Bratislava, Košice, Nitra, Prešov, Trenčín, Trnava, Žilina

Při zobrazení jednodenní předpovědi bude v případě České republiky zobrazen stav oblačnosti pro příslušný kraj společně s nejvyšší denní a nejnižší noční teplotou.

U Slovenské republiky bude zobrazován pro každý kraj pouze stav oblačnosti a v pravém dolním rohu nejvyšší denní a nejnižší noční teplota souhrnně pro celé Slovensko.

V případě, že bude pro některé krajské město ČR vydána výstraha nebo varování, bude u daného krajského města vyobrazena ikonka výstrahy v barvě označující stupeň výstrahy. Po najetí na ikonku výstrahy se pak zobrazí detaily výstražné informace nebo varování.

U čtyřdenní předpovědi bude pro každý den zobrazena ikonka se stavem oblačnosti doplněná slovním popisem, dále pak nejvyšší denní a nejnižší noční teplota a také datum společně s názvem dne v týdnu. Se čtyřdenním přehledem také úzce souvisí výstražné informace a varování. Pokud bude pro dané krajské město vydána nějaká výstraha nebo varování, bude zobrazena pod touto čtyřdenní předpovědí s informací o jejím stupni, kterou bude vyjadřovat ikonka, názvem příslušné výstrahy, začátkem a koncem platnosti. Po najetí na místo výstrahy se opět zobrazí její podrobnosti.

### 7.1.3 Stavby oblačnosti, výstražné informace a varování

Jelikož každý ze zdrojů, se kterými bude aplikace pracovat, používá jiný způsob reprezentace stavů oblačnosti, bude nutné vytvořit vlastní číselník, který pokryje všechny možné stavy těchto zdrojů. Návrh vhodného číselníku, který by pokryl všechny možnosti stavů oblačnosti jednotlivých zdrojů, obsahuje 19 různých možností. Číslování a název všech stavů uvádí tabulka 7.1.

Při načítání dat bude nutné převést reprezentaci stavů oblačnosti jednotlivých zdrojů na některý z 19 stavů popsaných v tabulce 7.1. Pro snazší implementaci tohoto převodu jsem se rozhodl vytvořit přehlednou konverzní tabulku 7.2, která ukazuje způsob tohoto přemapování.

Kód stavu	Slovní označení stavu
00	Neznámý
01	Jasno
02	Skoro jasno
03	Polojasno
04	Oblačno
05	Zataženo
06	Děšť
07	Dešťové přeháňky
08	Sněžení
09	Sněhové přeháňky
10	Sněžení s deštěm
11	Sněhové přeháňky s deštěm
12	Plískanice
13	Možnost plískanic
14	Bouřky
15	Možnost bouřek
16	Kroupy
17	Opar
18	Mlha

Tabulka 7.1: Číselník stavů oblačnosti.

Kód stavu	Meteocentrum.cz <obrazek>...</obrazek>	IN-POČASIE <stav>...</stav>	Weather Underground <icon>...</icon>
00	00	-	unknown
01	01, 03	01	sunny, clear
02	02	02	mostlysunny
03	04, 05 ,06	03	partlysunny
04	07, 08, 09	04	mostlycloudy, partlycloudy
05	10, 11, 12	05	cloudy
06	15, 16, 17	12	rain
07	18, 19, 20	08, 10, 11	chancerain
08	24, 25, 26	17	snow
09	27, 28, 29	14	chancesnow
10	30, 31, 32	16	sleet
11	-	15	chancesleet
12	-	-	flurries
13	-	-	chanceflurries
14	21, 22, 23	07, 13	tstorms
15	-	09	chancetstorms
16	33	18	-
17	-	-	hazy
18	13, 14	06	fog

Tabulka 7.2: Převodní tabulka pro jednotnou reprezentaci stavů oblačnosti.

Po nastudování specifikace Systému integrované výstražné služby poskytnuté Českým hydrometeorologickým ústavem jsem se rozhodl zobrazovat pouze ikonku v barvě stupně vydané výstrahy. ČHMÚ používá trojbarevné označení – žlutou, oranžovou a červenou barvu, která v uvedeném pořadí symbolizuje první až třetí stupeň nebezpečnosti výstrahy.

Uvažoval jsem také o vytvoření ikonky pro každý typ výstrahy (např. povodňová aktivity, silný vítr, apod.). Když jsem ale vzal v úvahu velikost zobrazované ikonky, měl jsem obavu o to, aby byla ikonka se symbolem dobře čitelná. Testování uvedeného způsobu zobrazení výstrahy mé obavy potvrdilo, a proto jsem se rozhodl pro pouhé barevné rozlišení stupně.

## 7.2 Grafické uživatelské rozhraní

Při návrhu schématického rozložení prvků na stránce jsem kladl především důraz na to, aby byla stránka přehledná a nezatěžovala uživatele zbytečnými informacemi. Stránka bude opticky rozdělena na část s jednodenní předpovědí a místem pro zobrazení detailů čtyřdenní předpovědi, nad kterým se bude vyskytovat vyhledávací panel pro zadání parametru vyhledávání předpovědi pro konkrétní město.

Aplikace bude optimalizována pro rozlišení 1024 na 768 pixelů. Toto rozlišení je podle statistiky dostupné na internetu<sup>2</sup> mezi uživateli stále nejvíce zastoupené. Uvažoval jsem také o druhém nejčastějším rozlišení, kterým je 1200 na 800 pixelů, ale dle mého názoru je pro uživatele lepší nemít aplikaci zobrazenou přes celou šířku obrazovky než používat horizontální posuvník pro zobrazení informace na pravém okraji stránky.

Při přepnutí do režimu celé obrazovky bude obraz zvětšen poměrově na základě rozlišení obrazovky uživatele, díky čemuž nebude docházet k deformaci obrazu. Aplikace tedy nebude automaticky roztazena na celou obrazovku, jako je tomu například u videí na Youtube.

Grafické stránka aplikace bude vytvořena za použití nástroje Microsoft Expression Blend 3, který byl popsán v podkapitole 2.4.2. Návrh grafického uživatelského rozhraní aplikace se nachází na obrázku 7.1.

Ovládání aplikace bude probíhat výhradně pomocí myši, pouze v případě vyhledávání předpovědi pro konkrétní město bude použito klávesnice pro zapsání názvu tohoto města.

Dále bude nutné vytvořit následující částí aplikace, které obohatí grafické uživatelské rozhraní:

- Obrysové mapky České a Slovenské republiky s vyznačením jednotlivých krajů.
- Ikonky pro 19 stavů oblačnosti, které byly popsány v tabulce 7.1.
- Ikonku pro každý ze tří stupňů vydané výstražné informace a varování.
- Ikonky pro přepnutí do celoobrazovkového režimu a zpět, zobrazení nápovědy a informací o programu.

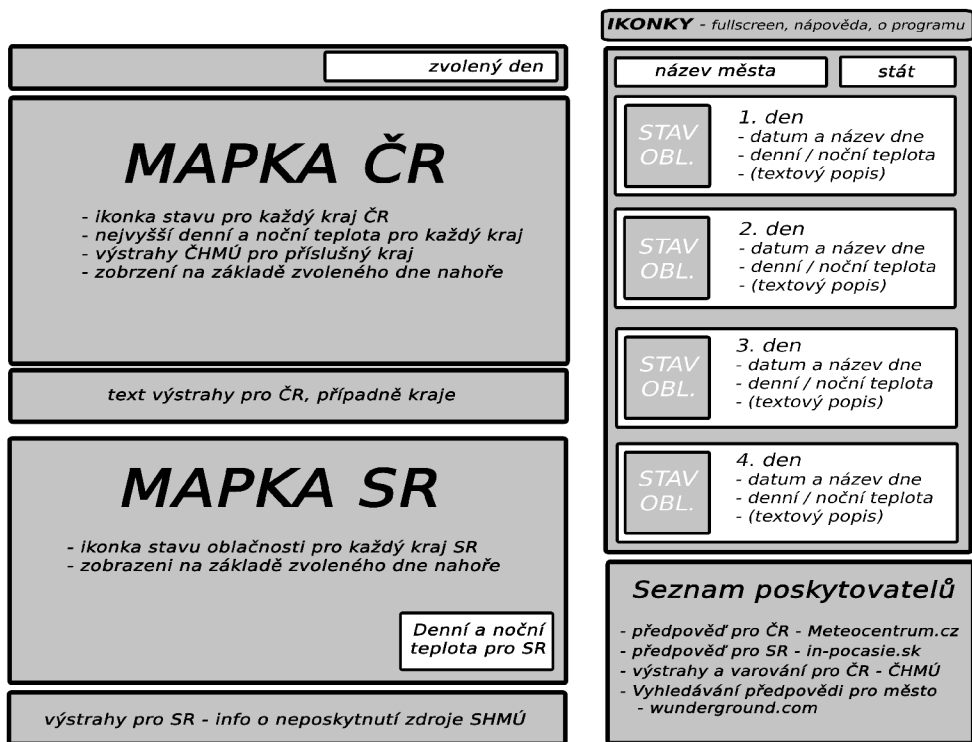
## 7.3 Zřízení domény a blogu

Již delší dobu jsem uvažoval o zřízení vlastní domény, proto jsem se rozhodl v souvislosti s bakalářskou prací pro její založení. Zpočátku na ní budu vést blog k bakalářské práci,

---

<sup>2</sup>Statistika rozlišení monitorů – <http://www.w3counter.com/globalstats.php>





Obrázek 7.1: Návrh grafického uživatelského rozhraní aplikace.

později se na ní objeví aplikace a poté ji budu využívat k vlastní prezentaci a soukromým účelům.

Se zřízením vlastní domény také úzce souvisí rozhodnutí založení blogu týkajícího se mé bakalářské práce. Od tohoto kroku si slibuji neustálou dostupnost všech mých poznámek, pozdější usnadnění psaní písemné části práce a v neposledním řadě také lepší možnost kontroly mé činnosti ze strany vedoucích. Blog bude umístěn na nově zřízené doméně.



# Kapitola 8

## Implementace

Obsahem této kapitoly je z mého pohledu nejzajímavější část vývojového cyklu aplikace, kterou je implementace.

V úvodu je věnován poměrně velký prostor implementaci samotné aplikace, kde jsou vysvětleny podstatné detaily kódu a principy fungování mashup aplikace. Následující část popisuje implementaci grafického uživatelského rozhraní a jeho jednotlivých částí. Poté je v krátkosti zmíněno zřízení vlastní domény a obsah blogu.

V závěru kapitoly se nachází popis dvou problémů, které se objevily ve fázi implementace. U obou problémů jsou také uvedena řešení, která vedla k jejich odstranění.

### 8.1 Aplikace

Aplikace byla naprogramována v technologii Silverlight za použití programovacího jazyka C#. Vývoj proběhl nástrojem Microsoft Visual Studio 2008.

Program používá ke své činnosti dva základní typy objektů. Objekt třídy `City`, který obsahuje předpověď pro konkrétní město na jeden den a objekt třídy `Warning` obsahující data o vydané výstraze. Atributy obou uvedených objektů jsou popsány níže.

```
public class City
{
    public string name { get; set; }           // jmeno mesta
    public string pattern { get; set; }       // vyhledavaci vzorek
    public string date { get; set; }         // datum
    public string day { get; set; }          // den v tydnu
    public string temperatureDay { get; set; } // denni teplota
    public string temperatureNight { get; set; } // nocni teplota
    public int state { get; set; }           // stav oblacnosti
}

public class Warning
{
    public string code { get; set; }         // kod kraje
    public int awarenessLevel { get; set; }  // stupen vystrahy
    public string awarenessType { get; set; } // typ vystrahy
    public string startDate { get; set; }    // datum zacatku
    public string startTime { get; set; }    // cas zacatku
    public string startPattern { get; set; } // vzorek zacatku pro porovnani
    public string endDate { get; set; }      // datum konce
    public string endTime { get; set; }     // cas konce
    public string endPattern { get; set; }   // vzorek konce pro porovnani
    public string information { get; set; }  // informace CHMU
    public string recommendation { get; set; } // doporuceni CHMU
}
```

Uvedené objekty jsou uloženy v globálních objektech typu `List`, což jsou kolekce objektů stejného typu obsahující předpověď pro konkrétní stát nebo výstražné informace pro ČR. Názvy jednotlivých kolekcí uvádí tabulka 8.1.

předpověď pro ČR	<code>List&lt;City&gt;forecastCzech</code>
předpověď pro SR	<code>List&lt;City&gt;forecastSlovak</code>
výstražné informace a varování pro ČR	<code>List&lt;Warning&gt;warningsCzech</code>

Tabulka 8.1: Seznam objektů obsahující načtená data.

### 8.1.1 Inicializace a načtení dat

Během načítání aplikace je nejprve zaregistrována (handler) `Content_Resize`, který bude vyvolán vždy při změně velikosti okna nebo přepnutí do režimu celé obrazovky a zajistí tak správné roztažení aplikace. Při startu aplikace dochází také k inicializaci slovníků pomocí metody `InitializeDictionaries`, které aplikace používá v průběhu své činnosti. Slovník `Dictionary` je objekt, který obsahuje dvojici položek klíč–hodnota libovolných datových typů a umožňuje přístup k hodnotám slovníku na základě klíče.

V rámci této fáze jsou stažena data s předpovědí pro Českou a Slovenskou republiku a také výstražné informace a varování. Stahování řídí metoda `DownloadData`, která postupně vyvolá metody pro zahájení stahování dat `GetForecastCZ` pro Českou republiku, `GetForecastSK` pro SR a `GetWarningsCZ` pro výstražné informace a varování pro ČR. Všem metodám je jako jediný parametr předán řetězec s adresou zdroje dat.

Stažení dat probíhá asynchronně pomocí komponenty Silverlight zvané `Webclient`. Ve všech třech případech je po stažení dat ze zdroje vyvolán handler, v jehož těle dochází ke zpracování stažených dat ve formátu XML.

V případě České republiky se jedná o handler `clientCZ_DownloadStringCompleted`, pro Slovensko nese název `clientSK_DownloadStringCompleted` a u výstrah a varování se jedná o `clientWA_DownloadStringCompleted`. V jejich těle dojde ke kontrole správnosti stažení dat a jejich rozparsování do XML dokumentu. Nastane-li v některé z těchto fází chyba, je v případě předpovědi zobrazena na mapce státu informace o nedostupnosti předpovědi, u výstrah je pak uživatel upozorněn dialogovým oknem třídy `Error`. XML dokument je pak dále předán referencí metodám, které zajistí jeho načtení.

Pro zpracování stažených dat pro Česko republiku je volána metoda `GetForecastCzech` třídy `ForecastCZ`, ta provede nutné konverze za použití metod ze stejnojmenné třídy a vrátí načtená data v objektu typu `List<City>` nebo `null` při chybě. U Slovenska tuto činnost zajišťuje metoda `GetForecastSlovak` ze třídy `ForecastSK`. Stažená data s výstrahami jsou předána metodě `GetWarningCzech` ze třídy `Warnings`, která je zpracovaná vrací v objektu typu `List<Warning>`.

Načtení rozbalovací nabídky dnů pro volbu jednodenní předpovědi je realizováno metodou `LoadDropDownList`, která načítá data primárně ze zdroje pro Českou republiku, v případě absence používá slovenský zdroj. Při absenci obou zdrojů není nabídka načtena vůbec.

Součástí inicializace je také vytvoření a spuštění časovače, které je provedeno zavoláním metody `CreateTimer`. Zmíněná metoda zobrazí aktuální datum i čas a zaregistruje handler `TimerTick` vyvolávaný při každém přetečení časovače.

### 8.1.2 Zobrazení předpovědi a výstrah

Metody pro zobrazení jednodenní předpovědi počasí pro Českou a Slovenskou republiku jsou volány ihned po stažení dat z příslušných zdrojů. V případě České republiky je volána metoda `FilterDayForecastCZ`, pro Slovensko pak `FilterDayForecastSK`. Oběma metodám je předáno aktuální datum a samy provedou vyfiltrování předpovědi z lokální databáze obsažené v objektech `List<City>forecastCzech` a `List<City>forecastSlovak`. Vyfiltrována data jsou předána k zobrazení uživateli prostřednictvím metod `ShowDayForecastCZ` v případě ČR, pro Slovensko zajišťuje provedení této akce metoda `ShowDayForecastSK`.

Výstražné informace a varování jsou zobrazovány v jednodenní přehledu na mapce ČR u příslušného kraje. Po načtení dat s výstrahami je zavolána metoda `FilterDayWarningCZ` pro aktuální datum, která v první fázi provede vyfiltrování všech výstrah, které jsou aktuální pro tento den (probíhající výstrahy nebo ty, které začínají v tento den). V druhé fázi kontroluje, zda již není po skončení platnosti dané výstrahy. Data jsou poté předána k zobrazení na mapce České republiky metodě `ShowDayWarningCZ`, která na ní zobrazí ikonku podle stupně nebezpečnosti výstrahy.

Podobně je tomu v případě změny zobrazovaného dne ze strany uživatele. Změna položky v nabídce dat způsobí vyvoláním události `selectDay_SelectionChanged`, která zjistí zvolené datum a provede stejnou posloupnost jednotlivých operací jako bylo uvedeno výše jen s tím rozdílem, že místo aktuálního data je metodám pro filtrování předpovědi a výstražných informací předáno datum dne požadovaného uživatelem.

Kliknutím na rámeček s předpovědí počasí u některého z krajských měst České a Slovenské republiky dojde k vyvolání hadleru `IconClick`, který rozpozná na které pole uživatel klikl, přiřadí mu název města a pro toto město je volána metoda `FilterFourDayForecast`. Uvedená metoda vyfiltruje z dat s předpovědí pro ČR nebo SR předpověď pro zadané město a předá tato data k zobrazení metodě `ShowFourDayForecast`. V případě, že se jedná o město České republiky je také zavolána metoda `FilterFourDayWarning`. Tato metoda nejprve převede název města, které obdrží jako parametr při svém volání, na kód kraje a vyfiltruje dostupné předpovědi pro tento kraj. Podmínkou je, aby již nebylo po konci platnosti výstrahy a aby zobrazení výstrahy mělo smysl pro některý ze čtyř zobrazovaných dnů. Data k zobrazení uživateli jsou pak předána metodě `ShowFourDayWarning`.

Při najetí na ikonku příslušné výstrahy na mapce České republiky nebo najetí na oblast popisující výstrahu ve čtyřdenním přehledu, dojde k zachycení této události. Po nezbytné identifikaci kraje, který si uživatel přeje zobrazit dojde k volání metody `ShowWarningDetail` pro vyvolání popup okna s detaily výstrahy, která očekává jako jediný parametr kód kraje. Metoda si přečte data o konkrétní výstraze z objektu `List<Warning>` a zobrazí je uživateli. Opuštěním ikonky nebo oblasti s výstrahou ve čtyřdenní přehledu dojde ke schování popup okna.

### 8.1.3 Vyhledávání pro konkrétní město

Vyhledávání nefunguje v režimu celé obrazovky z důvodů uvedených v podkapitole 8.4.1. Po zadání názvu města, volbě kódu státu a stisku tlačítka „vyhledat“ dojde nejprve ke kontrole, zda uživatel skutečně zadal nějaká data. Pokud tomu tak není, je vyvolán dialog třídy `Error`, který uživatele na tento nedostatek upozorní.

Pokud kontrola proběhne správně, je volána metoda `SearchingCity`. V jejím těle je ze zadaného města odstraněna veškerá diakritika a všechna velká písmena jsou převedena na malá. Poté dochází ke kontrole lokálních databází (pokud jsou dispozici). V případě shody u některého ze států je volána metoda `FilterFourDayForecast`, které je předán

již skutečný název města společně s kódem státu. Tato metoda vyfiltruje předpověď pro požadované město a předá tato data dále ke zobrazení metodě `ShowFourDayForecast`.

Jestliže lokální databáze předpovědí pro požadované město nedisponuje, je vytvořen ze zadaných údajů požadavek, který je předán jako jediný parametr ke zpracování metodě `GetForecastWU`. Ta zahájí stažení XML souboru s předpovědí, po dokončení stahování je soubor rozparsován a předán metodě `GetForecastCity` třídy `Wunder`. Uvedená metoda zajistí správné načtení stažených dat včetně konverze data, stavů oblačnosti apod. a načtená data vrátí zpět v objektu typu `IEnumerable<City>`. Tento objekt je následně předán opět ke zobrazení uživateli metodě `ShowFourDayForecast`.

Pokud se nepodaří soubor z `Weather Underground` získat nebo dojde k chybě při parsování je uživateli oznámena tato skutečnost oznámením, že data nelze zobrazit. O probíhajícím stahování je uživatel po celou dobu hledání informován zprávou o probíhajícím hledání.

### 8.1.4 Ostatní části

Aplikace disponuje mimo hlavních funkcí uvedených výše také několika menšími, které nelze opomenout. Jedná se o pravidelnou aktualizaci času, přepnutí aplikace do režimu celé obrazovky a zpět, zobrazení nápovědy, informací o programu a webových stránek poskytovatelů.

Správné zobrazení aktuálního času a data zajišťuje časovač, který je inicializován při startu aplikace. Každých 100 ms je tímto časovačem vyvolán handler `TimerTick`, jehož obsluha provede nastavení nové hodnoty času, data nebo názvu dne. K této změně dochází pouze tehdy, jsou-li již zobrazené údaje zastaralé, tj. došlo k uplynutí jedné vteřiny nebo změně data od posledního zápisu. Pro získání aktuálního data a času je použit systémový objekt `Datetime`.

Při změně velikost okna nebo přepnutí aplikace z nebo do režimu celé obrazovky dojde k vyvolání handleru `Content_Resize`, který zajišťuje správnou velikost a pozici aplikace v okně prohlížeče. Při přepnutí do režimu celé obrazovky je zjištěna velikost obrazovky monitoru a je vypočítán tzv. *scale*, což je poměr, kterým je vynásoben každý rozměr aplikace. Po přepnutí do normálního režimu je velikost navrácena zpět na původní hodnotu. Při normálním zobrazení tato funkce vypočítává a nastavuje správné okraje na levé i pravé straně, aby byla aplikace vždy umístěna horizontálně uprostřed stránky při horním okraji.

Obsluhu ikonků v pravého horní části zajišťují handlers `border_FullscreenClick`, `border_HelpClick` a `border_AboutClick`, aktivované kliknutí na některou z ikonků. První přepíná aplikaci mezi normálním režimem a režimem celé obrazovky prostou negací atributu aplikace `Application.Current.Host.Content.IsFullScreen`. Druhý handler vyvolá modální dialogové okno s nápovědou definované ve třídě `Help` a poslední zajistí zobrazení dialogu s informacemi o programu ze třídy `About`.

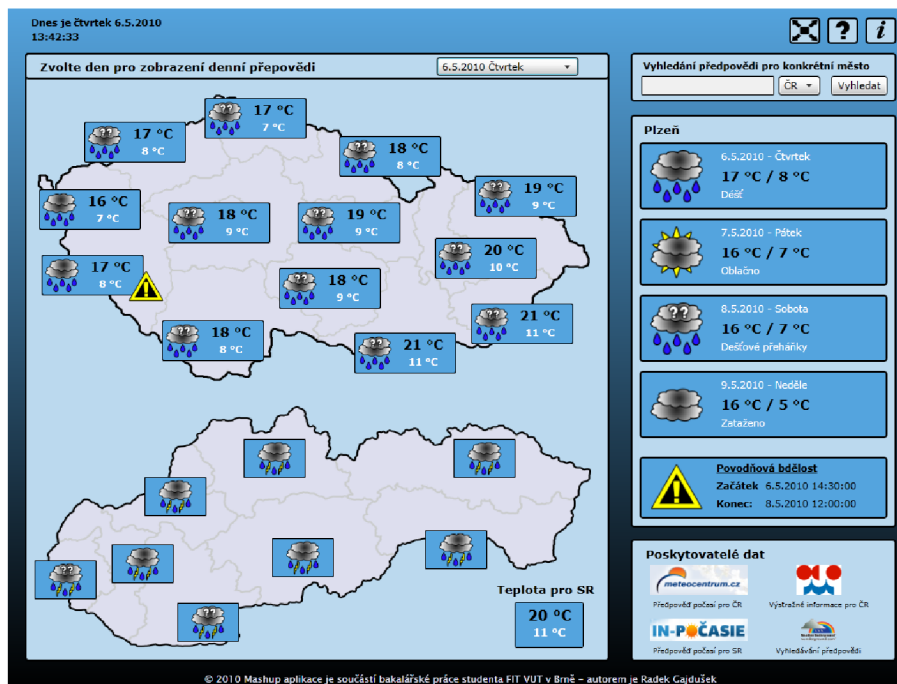
Kliknutím na ikonku některého ze čtyř poskytovatelů v pravé dolní části aplikace je vyvolán handler `LinkIconClick`, který zajišťuje zobrazení webové stránky poskytovatele. Nejprve je zjištěno, na kterou ikonku uživatel klikl a poté dojde k přesměrování na jeho webové stránky.

## 8.2 Grafické uživatelské rozhraní

Design provedení aplikace byl vytvořen na základě návrhu nastíněného v podkapitole 7.2. Při implementaci došlo k několika menším změnám, které obohatily grafické uživatelské rozhraní. Jedná se o přidání aktuálního data, názvu dne v týdnu a aktuálního času do

levého horního rohu aplikace. Dále byl odstraněn rámeček s informací o neposkytnutí dat s výstrahami ze strany SHMÚ. Nově se zobrazuje aktuální výstraha nebo varování také ve čtyřdenním přehledu po kliknutí na příslušné město. Poslední změnou je zobrazování detailu výstrah a varování. Oproti původním plánu se zobrazují po najetí myši místo mapy Slovenska, čímž tuto mapku po dobu zobrazení detailů výstrahy zakrývají.

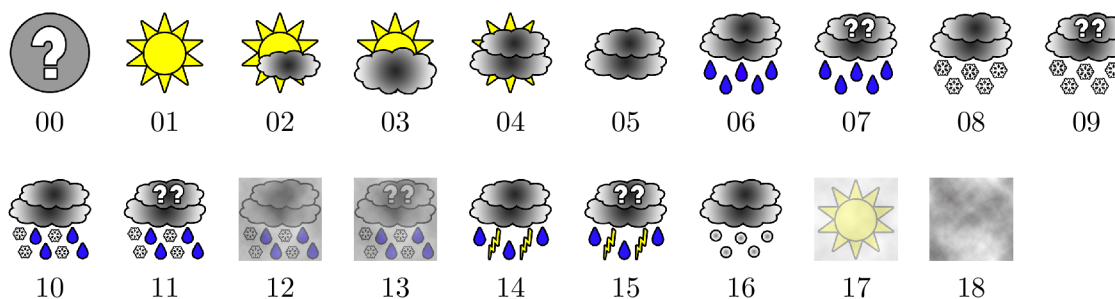
Ukázka finálního designu aplikace se nachází na obrázku 8.1.



Obrázek 8.1: Grafické uživatelské rozhraní mashup aplikace s předpovědí počasí.

### 8.2.1 Ikonky oblačnosti a výstrah

Ikonky pro jednotlivé stavy oblačnosti byly vytvořeny na základě tabulky 7.1. Každá z ikon symbolizuje jeden stav oblačnosti, vzhled všech ikon společně s číslem stavu uvádí tabulka 8.2. V případě vystražných informací a varování označuje barva výstražného trojúhelníku stupeň nebezpečnosti výstrahy podle klasifikace ČHMÚ. Design ikon pro jednotlivé stupně výstrahy je vyobrazen společně se stupněm nebezpečí v tabulce 8.3.



Tabulka 8.2: Vzhled ikonků označujících stav oblačnosti.





Tabulka 8.3: Vzhled ikonек jednotlivých stupňů výstrahy.

Pro tvorbu obou typů ikonек bylo využito programu Adobe Photoshop 7.0. Obrázky jsou uloženy v rastrovém formátu PNG, který nepodléhá deformaci při transformaci obrazu. Tato vlastnost je velmi užitečná při přepnutí aplikace do režimu celé obrazovky, kdy je obraz bez problému čitelný bez jakékoliv známky deformace.

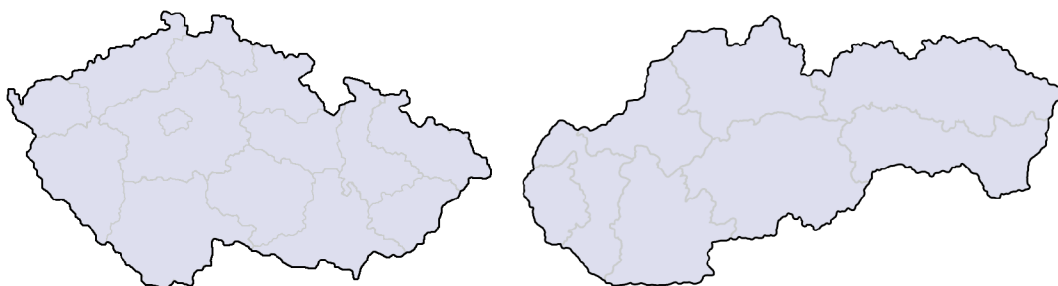
### 8.2.2 Mapky států s vyobrazením krajů

Mapky obou států byly vytvořeny na základě podkladů dostupných na internetu šiřitelných pod volnou licenci.

Mapka České republiky byla vytvořena za použití zdroje, který lze nalézt na této adrese [6] a který je šířen pod licenci GPL<sup>1</sup>.

Při tvorbě mapy Slovenska jsem použil podklad dostupný na tomto místě [15], jehož autorem je člověk s přezdívkou Caroig. Použití tohoto zdroje je podmíněno dodržením licence GFDL<sup>2</sup>.

Obě mapy, které se nachází na obrázku 8.2, byly vytvořeny v programu Adobe Photoshop 7.0 a každý kraj je reprezentován jednou vrstvou. Výhodou tohoto přístupu je možnost libovolné barevné úpravy kteréhokoliv kraje nebo celého státu.



Obrázek 8.2: Design mapek České a Slovenské republiky s vyznačením krajů.

## 8.3 Ostatní detaily

Autorem byla zřízena pro účely bakalářské práce doména 2.řádu s názvem *ragaj.cz*. Na stejné adrese se také nachází blog k bakalářské práci, který byl pravidelně aktualizován v průběhu implementace o poznatky a informace, získané v této fázi. Rovněž na něj byly umístěny všechny informace týkající se analýzy a návrhu aplikace.

Po dokončení aplikace a jejím řádném otestování byla umístěna ostrá verze na adresu <http://weather.ragaj.cz>. Na tomto místě se bude aplikace nacházet minimálně do chvíle

<sup>1</sup>GNU GPL – [http://cs.wikipedia.org/wiki/GNU\\_General\\_Public\\_License](http://cs.wikipedia.org/wiki/GNU_General_Public_License)

<sup>2</sup>GNU FDL – [http://cs.wikipedia.org/wiki/GNU\\_Free\\_Documentation\\_License](http://cs.wikipedia.org/wiki/GNU_Free_Documentation_License)

obhajoby bakalářské práce, poté bude záležet na domluvě s poskytovateli dat, kteří přislíbili poskytnutí zdrojů do konce měsíce června.

## 8.4 Problémy při implementaci

V této podkapitole jsou podrobně rozebrány dva největší problémy, které se objevily při implementaci aplikace. Součástí každého problému je podrobný popis problematiky a aplikované řešení.

### 8.4.1 Vyhledávání v celoobrazkovém režimu

Při implementaci zobrazení aplikace v režimu celé obrazovky došlo ke zjištění, že v tomto režimu nelze zadávat název města pro vyhledání předpovědi.

Technologie Silverlight neumožňuje z důvodu bezpečnosti zadávat znaky na klávesnici při zobrazení aplikace na celou obrazovku. Toto omezení má za úkol zabránit počítačovým hackerům ovládnutí počítače bez vědomí uživatele, který se nachází v režimu celé obrazovky a nemá ponětí o tom, co se v tu chvíli děje na pozadí.

Proto jsem se rozhodl při zobrazení aplikace na celou obrazovku vyhledávací panel schovat a umístit na jeho místo text, který upozorní uživatele na fakt, že vyhledávání není v tomto režimu dostupné. Po najetí myši na tento popisek se zobrazí krátké vysvětlení.

### 8.4.2 Bezpečnostní politika

Při zkoušce stažení dat v rané fázi implementace jsem narazil na problém, projevující se stažením dat pouze z jednoho zdroje, kterým byl Weather Underground.

Podářilo se mi zjistit, že tato situace je způsobena absencí souboru `crossdomain.xml` v root adresářích domén všech poskytovatelů. Uvedeným souborem disponují jen zdroje Weather Underground a Googlu, přičemž druhý jmenovaný neumožňuje Silverlight aplikacím nastavenou politikou přistupovat k jejich datům.

Tento soubor definuje přístupová práva aplikací při přístupu k datům z jiných domén. Před samotným stažením dat hledá Silverlight aplikace v root adresáři dané domény soubor `crossdomain.xml`, pokud jej nalezne podívá se na jeho obsah, který definuje přístupová práva. Jestliže aplikace práva splňuje, jsou data stažena, v opačném případě je přenos zamítnut. Pokud soubor `crossdomain.xml` v root adresáři úplně chybí, chová se aplikace jakoby neměla práva pro přístup k datům.

Jelikož tato komplikace výrazně snížila možnosti mé aplikace, byl jsem nucen tuto situaci řešit. Nejprve jsem požádal ČHMÚ, společnost Meteocentrum.cz a IN-POČASIE o nahrání tohoto souboru do root adresáře jejich domény. Vyšli mi vstříc pouze v případě Českého hydrometeorologického ústavu. Od společnosti Meteocentrum jsem dostal negativní odpověď a od IN-POČASIE jsem pro změnu nedostal odpověď vůbec.

Po konzultaci s vedoucím, Ing. Rudolfem Kajanem, jsme se rozhodli pro řešení spočívající ve vytvoření skriptu, který by každý den stáhl data z nedostupných zdrojů a umístil je na vlastní doménu, kde by přístupová práva již byla.

Skript je napsán pro `bash shell` a jeho úkolem je stáhnout vždy 5 minut po půlnoci data pomocí unixového programu `wget` ze zdroje společnosti Meteocentrum.cz a IN-POČASIE. Stažená data jsou poté vy publikována na web. Skript byl umístěn na server a do root adresáře domény `eliska.jarcec.net` byl umístěn soubor `crossdomain.xml`.

V případě Googlu bylo nutné z výše uvedených důvodů tento zdroj zahrnout a použít pro vyhledávání pouze zdroj Weather Underground.



## Kapitola 9

# Testování

Následující kapitola se věnuje testování finální mashup aplikace. V této fázi vývojového cyklu bylo provedeno 16 testů, které měly za úkol ověřit správnou funkčnost aplikace a odhalit chyby, kterých se autor dopustil při implementaci.

Testování je rozděleno na čtyři okruhy, které postupně testují hlavní části aplikace. Nejprve je otestována předpověď počasí, poté výstražné informace a varování, následuje vyhledávací modul a v závěru jsou testovány menší části aplikace. U každého testu je podrobně popsán jeho význam, očekávané výsledky a jeho přínos.

Pro testování bylo použito všech pět nejpoužívanějších prohlížečů v prostředí operačního systému Windows. Verze prohlížečů, na nichž testování proběhlo, uvádí tabulka 9.1.

Název prohlížeče	Verze
Google Chrome	4.1
Internet Explorer	8.0
Mozilla Firefox	3.6
Opera	10.10
Safari	4.0

Tabulka 9.1: Verze prohlížečů použitých při testování.

Pod operačním systémem Linux se mi technologii Silverlight i přes instalaci plug-inu Moonlight firmy Novell nepodařilo v žádném prohlížeči zprovoznit. Z uvedeného důvodu nebyla aplikace na této platformě testována.

### 9.1 Předpověď počasí

V této části budou popsány a provedeny čtyři testy, které ověří správnost zobrazovaných informací z hlediska předpovědi počasí. Zároveň bude otestována schopnost aplikace detekovat chybové stavy a reagovat na požadavky uživatele na změnu zobrazovaného dne s jednodenní předpovědí. Dále bude také otestováno správné zobrazení detailu čtyřdenní předpovědi pro konkrétní město.

#### Test č. 1 – Zobrazení předpovědí při startu aplikace

První test aplikace měl za úkol ověřit správné načtení a zobrazení předpovědi počasí pro aktuální den na mapce České i Slovenské republiky při startu aplikace.

- **Očekávaný výsledek:**

Na mapce obou států se zobrazí předpověď pro aktuální den, v případě České republiky společně s ikonkou oblačnosti také nejvyšší denní a nejnižší noční teplota, u Slovenska se na mapce zobrazí jen stav oblačnosti pro každý kraj, v pravém dolním rohu pak nejvyšší denní a nejnižší noční teplota souhrnně pro celou SR.

- **Závěr testu:**

Test potvrdil, že tato stěžejní část aplikace funguje bez problému.

## Test č. 2 – Změna předpovědi výběrem dne z nabídky

Smyslem tohoto testu bylo ověřit správnou reakci aplikace na změnu dne pro zobrazení jednodenní předpovědi. Postupně byly vybrány všechny možnosti z nabídky a bylo sledováno, zda došlo ke správné reakci ze strany aplikace. Také byla sledována správnost zobrazovaných předpovědí na základě porovnání se zdrojovým XML dokumentem u obou států.

- **Očekávaný výsledek:**

Po výběru jiného dne z nabídky by mělo dojít ke změně zobrazovaných informací u obou států. Měla by být načtena nová předpověď pro zvolený den a zobrazena na mapce České i Slovenské republiky.

- **Závěr testu:**

Výsledky tohoto testu potvrzují očekávané chování a zobrazovaná data odpovídají zdrojům.

## Test č. 3 – Zobrazení čtyřdenní předpovědi

Cílem následujícího testu bylo ověřit správnou funkci zobrazení čtyřdenní předpovědi pro krajská města. Tento test byl proveden kliknutím na všechna krajská města České a Slovenské republiky a byla sledována korespondence čtyřdenní předpovědi s jednodenní, zobrazené na mapce státu.

- **Očekávaný výsledek:**

Kliknutím na rámeček příslušného kraje musí dojít k zobrazení čtyřdenní předpovědi v pravé části stránky a musí být správně uveden název města. Souhlasit také musí údaje týkající se data, názvu dne, ikonka a slovní popis stavu oblačnosti u všech dnů.

- **Závěr testu:**

Aplikace se v tomto ohledu chová přesně podle uvedeného popisu.

## Test č. 4 – Chybějící data s předpovědí

Poslední test byl zaměřen na namodelování chybového stavu, do něhož se aplikace dostane v případě, kdy se jí nepodaří načíst data z některého ze zdrojů, ať už z důvodu nedostupnosti XML souboru nebo chyby při jeho parsování.

- **Očekávaný výsledek:**

Pokud se aplikaci nepodaří načíst data z některého ze zdrojů, musí být tato skutečnost oznámena uživateli zobrazením textu o vzniklé situaci na mapce státu, pro něhož je předpověď nedostupná. Zároveň musí dojít ke správnému načtení nabídky s daty pro volbu dne jednodenní předpovědi. Načtení nabídky neproběhne pouze pokud není dostupná ani jedna z předpovědí.

- **Závěr testu:**

Tento test pomohl odhalit chybu v aplikaci, která způsobovala nenačtení nabídky dnů pro volbu zobrazované jednodenní předpovědi a projevovala se při nedostupnosti zdroje pro ČR. Po odstranění této chyby byl uvedený test proveden znovu a aplikace již fungovala podle očekávání.

## 9.2 Výstražné informace a varování

Výstražné informace nejsou vydávány denně jako například předpovědi počasí, proto bylo nutné pro otestování správné funkčnosti zobrazování výstrah pro jednotlivé kraje vytvořit vlastní testovací XML soubor, který by pokryl všechny možné situace.

Vytvořený XML soubor počítá se situací, kdy je pro několik krajů ČR vydána výstraha 1. stupně (výstraha žluté barvy) typu „Povodňová bdělost“. Tabulka 9.2 shrnuje parametry výstrah vytvořených za účelem tohoto testování.

Aplikace byla také v průběhu práce testována se skutečnými daty vydávanými ČHMÚ, pokud byly k dispozici. XML kanál byl pravidelně kontrolován a pokud došlo k vydání nějaké výstrahy, byla prováděna kontrola správnosti jejího zobrazení v aplikaci až do okamžiku jejího skončení.

Kraj	Začátek platnosti	Konec platnosti
Zlínský	21.4.2010 08:00	23.4.2010 22:00
Moravskoslezský	21.4.2010 08:00	24.4.2010 17:00
Jihočeský	24.4.2010 20:00	26.4.2010 21:00
Středočeský	26.4.2010 12:00	29.4.2010 12:00
Jihomoravský	28.4.2010 04:30	29.4.2010 18:00

Tabulka 9.2: Časové vymezení jednotlivých testovacích výstrah.

### Test č. 5 – Zobrazení výstrahy na mapce ČR

Test je na zaměřen na kontrolu správného zobrazení výstrah uvedených v tabulce 9.2 a jejich detailů. Test byl proveden dne 24.4.2010 v 17:00.

Postupně budeme z nabídky vybírat jednotlivé dny od prvního po poslední a sledovat počet zobrazených výstrah na mapě ČR a jejich správnost u jednotlivých krajů. Zároveň budeme zobrazovat detaily jednotlivých výstrah a provádět kontrolu jejich správnosti.

- **Očekávaný výsledek:**

Na mapce České republiky bude zobrazena žlutá ikonka s výstrahou u těchto krajů v následujících dnech:

Datum (čas)	Zobrazovaný kraj(e)
24.4.2010 17:10	Jihočeský
25.4.2010	Jihočeský
26.4.2010	Jihočeský, Středočeský
27.4.2010	Středočeský kraj

Výstraha pro Zlínský kraj se nezobrazí vůbec, neboť je již dávno po konci její platnosti, jelikož je aktuální čas 17:10, nezobrazí se ani v jednom případě výstraha pro Moravskoslezský kraj. Výstraha pro Jihočeský kraj se rovněž nezobrazí z toho důvodu, že začíná až den po našem zobrazovaném rozmezí.

- **Závěr testu:**

Tento test důkladně prověřil zobrazování výstrah na mapce České republiky a jejich filtrování pro jednotlivé dny, které funguje spolehlivě.

## Test č. 6 – Zobrazení výstrahy ve čtyřdenním přehledu

Cílem tohoto testu bylo ověřit správné zobrazení výstrahy ve čtyřdenním přehledu a ověřit funkčnost zobrazení detailů výstrahy. Pro testování byla stejně jako v prvním testu použita testovací data popsaná na začátku této podkapitoly. Zmíněná komponenta byla otestována dne 24.4.2010 v 18:05.

- **Očekávaný výsledek:**

Ve čtyřdenním přehledu se nesmí zobrazit data o výstrahách, jejichž platnost již skončila. Také se nesmí být zobrazeny výstrahy, které teprve vstoupí v platnost, ale jsou mimo zobrazované rozmezí čtyřdenního přehledu. Pokud aplikace pracuje správně, měly by být zobrazeny výstrahy pouze u Prahy (Středočeský kraj) a Českých Budějovic (Jihočeský kraj). Po najetí na příslušnou výstrahu ve čtyřdenním přehledu se zobrazí detaily výstrahy.

- **Závěr testu:**

Filtrování výstražných informací a varování funguje pro čtyřdenní přehled bezchybně, stejně jako zobrazení detailů po najetí myši na výstrahu nebo varování.

## Test č. 7 – Chybějící data s výstrahami

Následující test měl ověřit schopnost aplikace detekovat chybový stav, který je způsoben nedostupností datového zdroje s výstrahami. Záměrně byla při tomto testu změněna adresa zdroje na vlastní adresu, která nejprve neobsahovala žádná data a poté XML soubor, u něhož došlo k chybě během parsování.

- **Očekávaný výsledek:**

Aplikaci by se nemělo podařit ani v jednom případě načíst správně data. O této chybě musí informovat uživatele vyvoláním dialogového okna s informací o nastalé chybě.

- **Závěr testu:**

Detekce chyby při načítání dat s výstrahami proběhla úspěšně, aplikace v tomto směru pracuje spolehlivě.

## 9.3 Vyhledávání předpovědi

Významnou částí aplikace je modul pro vyhledání předpovědi na základě zadání názvu města. Následující podkapitola se věnuje testům, které ověřují správné chování aplikace při vyhledávání krajských měst obsažených v lokální databázi, dále vyhledávání předpovědi na Weather Underground a schopnosti aplikace pracovat správně bez lokálních zdrojů.

## Test č. 8 – Vyhledání předpovědi pro město v lokální databázi

Tento test je zaměřen na vyhledání předpovědi pro krajské město, které máme již načteno v lokální databázi. Tato funkcionality byla otestována zadáním názvu všech krajských měst České a Slovenské republiky do vyhledávacího řádku.

- **Očekávaný výsledek:**

Aplikace správně rozpozná, že se předpověď pro zadané město nachází v lokální databázi a nebude vyhledávat předpověď na Weather Underground. Po době nezbytné pro nalezení dat dojde k zobrazení čtyřdenní předpovědi.

- **Závěr testu:**

Výsledek tohoto testu potvrdil 100% schopnost vyhledávacího modulu detekovat zadání města z lokální databáze.

## Test č. 9 – Vyhledání předpovědi pro město mimo lokální databázi

Následující test se zaměřuje na správné vyhledání předpovědi pro město, které není obsaženo v lokální databázi.

Pro otestování funkčnosti vyhledávání byla pro Českou republiku zadána města Jindřichův Hradec, Kopřivnice, Nové Město na Moravě, Polička a Strančice. V případě Slovenska se jednalo o tato města: Humenné, Kysucké Nové Město, Liptovský Mikuláš, Martin a Rožňava.

U všech měst byly vyzkoušeny také varianty s diakritikou i bez ní, rovněž varianty s kombinací náhodně zapsaných velkých i malých písmen.

- **Očekávaný výsledek:**

Po zadání názvu je zjištěno, že zadané město není obsaženo v databázi a aplikací je odeslán dotaz na Weather Underground obsahující název zadaného města. Získaný XML dokument je po zpracování zobrazen ve čtyřdenním přehledu. V průběhu získávání dat je uživateli tato skutečnost oznámena textem o probíhající hledání. Pro všechna zadaná města by měla být zobrazena předpověď.

- **Závěr:**

Díky tomuto testu došlo k odhalení chyby, která znemožňovala vyhledání předpovědi pro město obsahující v názvu mezeru. Tato chyba byla opravena a aplikace poté opětovně testována. Nové testování již vrátilo očekávané výsledky.

## Test č. 10 – Vyhledání předpovědi pro město mimo lokální databázi

Předmětem tohoto testu byla správná detekce chybového stavu, kdy nejsou data s předpovědí pro zadané město k dispozici. Pro test vyhledávání neexistujícího města byly zadány následující řetězce: Humené, Kopřivnic, Martine a Hraadec Králové.

- **Očekávaný výsledek:**

Aplikace sice odešle požadavek obsahující zadané město na webovou adresu, Weather Underground však vrátí informaci o neznámé předpovědi a aplikace uživateli zobrazí informaci o tom, že data nejsou k dispozici.

- **Závěr:**

Testování detekce neexistujícího města proběhla úspěšně.

## Test č. 11 – Vyhledání předpovědi pro město při absenci lokální databáze

Tento test měl za cíl otestovat, zda aplikace dokáže správně zjistit, že některý z lokálních zdrojů není dostupný, tudíž nejsou data o krajských měst daného státu dostupné a dokáže pro toto město vyhledat předpověď na Weather Underground.

Nejprve bylo otestováno hledání města Brna a Plzně při absenci lokálního zdroje pro ČR. Totéž bylo provedeno při absenci dat pro SR zadáním měst Bratislava a Prešov. Na závěr byla otestována funkce vyhledávání při absenci obou zdrojů pro města Liberec a Trnava.

- **Očekávaný výsledek:**

Ve všech případech musí aplikace správně rozpoznat, že lokální data nejsou k dispozici a získat odpověď na uživatelský dotaz prostřednictvím Weather Underground a zobrazit jej ve čtyřdenním přehledu.

- **Závěr:**

Tento test odhalil závažnou chybu v aplikaci. Při absenci některého ze zdrojů aplikace nerozpoznala, že nemá lokální data k dispozici a nesprávně hledala předpověď v lokální databázi namísto použití zdroje Weather Underground. Ihned po detekci této chyby došlo k její opravě a byl znovu proveden tento test, který již vracel správné výsledky.

## 9.4 Ostatní funkcionalita

Poslední část testuje především prvky grafického uživatelského rozhraní jako například režim celé obrazovky. Nechybí ani test na správné zobrazení formátu data a času nebo test funkčnost odkazů na stránky poskytovatelů.

## Test č. 12 – Přepínání mezi celoobrazovkovým a normálním režimem

Účelem tohoto testu bylo vyzkoušet správnou funkci přepnutí aplikace do režimu celé obrazovky a zpět. Test proběhl na dvou monitorech. První disponoval rozlišením 1280 na 1024 pixelů, druhý měl rozlišení 1200 na 800 pixelů.

- **Očekávaný výsledek:**

U první obrazovky musí být obraz roztažen na celou šířku stránky a vertikálně umístěn do středu se stejně velkými okraji nahoře i dole. Při druhém rozlišení musí být aplikace roztažena na celou výšku monitoru a horizontálně umístěna doprostřed stránky se stejnými okraji na obou stranách. Při přepínání se musí měnit ikonka v pravém horním rohu informující uživatele o tom, ve kterém režimu se aktuálně nachází.

- **Závěr:**

Přepínání aplikace mezi normálním režimem a režimem celé obrazovky funguje včetně změn ikonky bezchybně.

## Test č. 13 – Změna velikosti okna prohlížeče

Cílem dalšího testu bylo ověřit správnou reakci aplikace na změnu velikosti okna prohlížeče.

- **Očekávaný výsledek:**

Při vertikálním i horizontálním zvětšení či zmenšení okna prohlížeče musí aplikace

zůstat v původním rozlišení a musí být vždy umístěná při horním okraji stránky. Při větší šířce stránky než je podporované rozlišení musí být umístěna vždy ve středu stránky se stejnou velikostí okrajů na levé i pravé straně.

- **Závěr:**

Aplikace se chová přesně podle očekávání, proto lze uvedený test prohlásit za úspěšný.

### **Test č. 14 – Zobrazení webové stránky poskytovatele**

Smyslem dalšího testu bylo vyzkoušet správnou reakci aplikace po kliknutí na ikonku libovolného poskytovatele dat. Test byl proveden u všech čtyř poskytovatelů.

- **Očekávaný výsledek:**

Po kliknutí na libovolnou ikonku poskytovatele aplikace správně rozpozná poskytovatele, jehož stránku si přeje uživatel zobrazit a zobrazí jeho domovskou stránku.

- **Závěr:**

Testovaná funkce pracuje bezchybně pro všechny čtyři poskytovatele dat.

### **Test č. 15 – Správná funkce ikonek nápovědy a o programu**

Následující test se zabývá správnou detekcí kliknutí na ikonky pro zobrazení nápovědy a informací o programu.

- **Očekávaný výsledek:**

Kliknutím na ikonku označující nápovědu nebo informace o programu se uživateli zobrazí modální okno s nápovědou, resp. s informacemi o programu. Do chvíle zavření tohoto okna je běh aplikace pozastaven a uživatel se může pohybovat pouze v oblasti vyvolaného okna.

- **Závěr:**

Vyvolání modálního okna a zobrazení informací funguje bez problému.

### **Test č. 16 – Stejný formát času a data při změně lokalizace**

Předmětem posledního testu byla kontrola správného zobrazení času, data a dne v týdnu při změně jazykové lokalizace systému. Při testu byla použita česká, německá a anglická lokalizace. Dále pak přibližně 10 dalších náhodně zvolených.

- **Očekávaný výsledek:**

Aktuální čas musí být za použití jakékoliv lokalizace systému zobrazen ve formátu **hh:mm:ss** a datum ve formátu **dd.mm.rrrr**. Název dne v týdnu musí být v češtině a s malým počátečním písmenem na začátku.

- **Závěr:**

Při tomto testu byl odhaleno špatné formátování data nikoliv podle českého formátu, ale na základě zvolené lokalizace. Stejným problémem se vyskytl i u času. Tento nedostatek byl opraven a tento test poté proveden znovu. Nové testování již potvrdilo bezchybnou funkčnost.



## Kapitola 10

# Závěr

Tato práce se věnovala návrhu a implementaci mashup aplikace s předpovědí počasí pro Českou a Slovenskou republiku. Aplikace je naprogramována v technologii Silverlight a používá ke své činnosti zdroje ve formátu XML.

Mashup aplikaci se podařilo úspěšně naimplementovat s menšími odchylkami podle zpracovaného návrhu.

S výsledky mé práce jsem spokojen, podařilo se vytvořit aplikaci, která zobrazuje předpověď počasí i výstražné informace systematicky na jediném místě. Aplikace reaguje na změnu dne pro zobrazení jednodenní předpovědi prakticky okamžitě a ani vyhledávání předpovědi pro zadané město netrvá příliš dlouhou dobu. Možnost zadání libovolného města a vyhledání předpovědi pro toto město považuji za nejužitečnější funkci celé aplikace. S technologií Silverlight od firmy Microsoft se mi pracovalo velmi dobře, a proto nevyklučuji její použití do budoucna při tvorbě některého ze školních nebo soukromých projektů.

Jelikož bylo použito bezplatných datových zdrojů považuji za menší nedostatek zobrazování předpovědi v případě Slovenska. Původní myšlenku se zobrazením stavu oblačnosti společně s nejvyšší denní a nejnižší teplotou pro každý kraj se nepodařilo uskutečnit stejně jako zobrazení výstražných informací a varování pro tento stát, které bylo způsobenou neochotou poskytnout tato data bezplatně Slovenský hydrometeorologickým ústavem. Uvedené nedostatky by odstranilo použití placených zdrojů.

Aplikaci by bylo možné do budoucna rozšířit o další typ zobrazovaných informací, jakými by mohly být například informace o množství sněhu v lyžařských střediscích apod. Toto rozšíření by si zřejmě vyžádalo změny grafického uživatelského rozhraní. Jednou z dalších možností by mohlo být vytvoření widgetu pro operační systém Windows, který by uživateli umožňoval vybrat město, pro něhož by měl neustále k dispozici předpověď na pracovní ploše systému.

# Literatura

- [1] ADOBE: *Using Adobe Flash CS4 Professional* [online]. 5th March 2009, [cit. 2010-02-11].  
URL [http://help.adobe.com/en\\_US/Flash/10.0\\_UsingFlash/index.html](http://help.adobe.com/en_US/Flash/10.0_UsingFlash/index.html)
- [2] BEDNÁŘ, V.: *Po webu je tu Web 2* [online]. 25. října 2005, [cit. 2010-02-16].  
URL <http://www.lupa.cz/clanky/po-webu-je-tu-web-2/>
- [3] BUCKLEY, B., et al.: *Počasí: Velký obrazový průvodce*. Rebo Productions CZ, spol. s r. o., Dobřejovice, první vydání, 2006, ISBN 80-7234-552-4.
- [4] ČESKÝ HYDROMETEOROLOGICKÝ ÚSTAV: *Druhy vydávaných předpovědí (informace) a meteorologická terminologie využívaná pro předpovědi počasí v OSM* [online]. 9. června 2006, [cit. 2010-03-08].  
URL <http://www.chmi.cz/meteo/om/info.html#drprp>
- [5] CHOW, S. W.: *Programujeme Mashup aplikace pro Web 2.0 v PHP*. Computer Press, a. s., Brno, první vydání, 2008, ISBN 978-80-251-2057-6.
- [6] FREE TOP LINKS: *Slepá mapa ČR* [online]. [cit. 2010-03-12].  
URL <http://free-top.tym.cz/download.php?ds=slepa-mapa-cr>
- [7] HAROLD, E. R., MEANS, W. S.: *XML v kostce*. Computer Press, a. s., Brno, první vydání, 2002, ISBN 80-7226-712-4.
- [8] LACKO, L.: *Silverlight 2.0: Úvod do technologie a tvorby aplikací pomocí nástrojov Microsoft Expression* [online]. Microsoft, 2007, [cit. 2010-02-09].  
URL [http://download.microsoft.com/download/d/0/b/d0b4a128-f397-4f74-ade0-12ed50d8ff0b/Silverlight\\_2.pdf](http://download.microsoft.com/download/d/0/b/d0b4a128-f397-4f74-ade0-12ed50d8ff0b/Silverlight_2.pdf)
- [9] LACKO, L.: *Silverlight 3.0: Úvod do vývoje aplikací na platforme Silverlight 3* [online]. Microsoft, 2009, [cit. 2010-02-10].  
URL [http://download.microsoft.com/download/C/0/6/C063ED26-AD71-4731-A6BA-791C6888BCD0/Silverligh\\_3.pdf](http://download.microsoft.com/download/C/0/6/C063ED26-AD71-4731-A6BA-791C6888BCD0/Silverligh_3.pdf)
- [10] MICROSOFT: *Microsoft Expression Blend 3* [online]. [cit. 2010-03-07].  
URL [http://www.microsoft.com/expression/products/Blend\\_Overview.aspx](http://www.microsoft.com/expression/products/Blend_Overview.aspx)
- [11] MICROSOFT MSDN LIBRARY: *Silverlight* [online]. 20th November 2009, [cit. 2010-02-11].  
URL <http://msdn.microsoft.com/en-us/library/cc838158%28VS.95%29.aspx>
- [12] O'REILLY, T.: *What is Web 2.0* [online]. 30th September 2005, [cit. 2010-02-16].  
URL <http://oreilly.com/web2/archive/what-is-web-20.html>

- [13] PROGRAMMABLE WEB: *Web 2.0 Mashups Directory* [online]. 27th April 2010, [cit. 2010-04-27].  
URL <http://www.programmableweb.com/popular>
- [14] ŠTURALA, A.: *WPF: Úvod* [online]. 17. ledna 2007, [cit. 2010-04-20].  
URL <http://www.vyvojar.cz/Articles/445-0-wpf-uvod.aspx>
- [15] WIKIMEDIA COMMONS: *Slovakia outline map*. [online], 3rd September 2007, [cit. 2010-03-12].  
URL [http://commons.wikimedia.org/wiki/File:Slovakia\\_-\\_outline\\_map.svg](http://commons.wikimedia.org/wiki/File:Slovakia_-_outline_map.svg)
- [16] YOUNG, M. J.: *XML krok za krokem*. Computer Press, a. s., Brno, druhé vydání, 2006, ISBN 80-251-1070-2.

## Příloha A

# Čestné prohlášení

Uvedené čestné prohlášení bylo dne 18. února 2010 zasláno Českému hydrometeorologickému ústavu společně s žádostí o zpřístupnění XML kanálu s výstražnými informacemi a varováními pro potřeby této bakalářské práce.


### ČESTNÉ PROHLÁŠENÍ

Já Radek Gajdušek, datum narození 16.2.1988 tímto

**čestně prohlašuji,**

že veškeré údaje s předpovědí počasí a výstrahami pro ČR, které mi poskytne Český hydrometeorologický ústav, budu využívat pouze k předem smluveným účelům, tj. výhradně pro potřeby mé bakalářské práce. Dále prohlašuji, že tato data nebudu poskytovat dalším osobám a nebudu je používat ke komerční účelům.

V Brně dne 17. února 2010.

  
**Radek Gajdušek**  
řešitel bakalářské práce

  
**Ing. Rudolf Kajan**  
vedoucí bakalářské práce

## Příloha B

# Zdrojový kód skriptu

Skript je napsán pro `bash shell` a odstraňuje problém s absencí souboru `crossdomain.xml` v root adresáři domény poskytovatelů `Meteocentrum.cz` a `IN-POČASIE`. Tento problém byl blíže popsán v podkapitole [8.4.2](#).

```
#!/bin/bash

#####
#   nabez: forecastXML.sh                               #
#   autor: Radek Gajdusek                               #
#   email: xgajdu07@stud.fit.vutbr.cz                  #
#   datum: 05.04.2010                                   #
#   popis: skript stahne XML soubory ze zadanych URL   #
#            a ulozi je do adresare s cestou dest pod   #
#            nazvem pocasiCR.xml a pocasiSR.xml         #
#   #####
# misto ulozeni
dest="xml"

# nabez stazenych xml souboru
xmlCZ="pocasiCR.xml"
xmlSK="pocasiSR.xml"

# nabez logovaciho souboru
logfile="download.log"

# url zdroju
forecastCZ="url_zdroje_pro_CR"
forecastSK="url_zdroje_pro_SR"

# vytvoreni adresare dest pokud neexistuje
mkdir -p "$dest"

# samotne stazeni souboru ze zadanych adres a ulozeni logu
wget $forecastCZ -O $dest/'/$xmlCZ -o $dest/'/$logfile
wget $forecastSK -O $dest/'/$xmlSK -a $dest/'/$logfile
```

## Příloha C

# Ukázka XAML

Tento úsek kódu, zapsaný v jazyce XAML, popisuje rámeček se stavem oblačnosti pro město Bratislava. Na začátku lze nalézt mimo definice akcí také transformace, které se s rámečkem provádějí. Tento prvek dále obsahuje textové pole s názvem města a obrázek, který definuje stav oblačnosti.

```
<Border x:Name="border_Bratislava" HorizontalAlignment="Left" Margin="29,0,0,99"
        VerticalAlignment="Bottom" Width="74" Grid.Column="6" Background="Black"
        BorderThickness="2" CornerRadius="2" MouseLeftButtonDown="IconClick"
        Visibility="Collapsed" RenderTransformOrigin="0.5,0.5">
  <i:Interaction.Triggers>
    <i:EventTrigger EventName="MouseEnter">
      <im:ControlStoryboardAction Storyboard="{StaticResource ShowBratislava}"/>
    </i:EventTrigger>
    <i:EventTrigger EventName="MouseLeave">
      <im:ControlStoryboardAction Storyboard="{StaticResource HideBratislava}"/>
    </i:EventTrigger>
  </i:Interaction.Triggers>
  <Border.RenderTransform>
    <TransformGroup>
      <ScaleTransform/>
      <SkewTransform/>
      <RotateTransform/>
      <TranslateTransform/>
    </TransformGroup>
  </Border.RenderTransform>
  <StackPanel x:Name="stack_Bratislava" Width="69" Orientation="Vertical"
    HorizontalAlignment="Left" VerticalAlignment="Top">
    <Border x:Name="border_TitleBratislava" Height="20" Background="Black"
      BorderThickness="0" CornerRadius="2" d:LayoutOverrides="HorizontalAlignment"
      Margin="0,0,0,-6" Visibility="Collapsed">
      <TextBlock Margin="4,1,0,0" FontSize="9.333" FontWeight="Bold" Foreground="White"
        Text="Bratislava " TextWrapping="Wrap" HorizontalAlignment="Left"
        Width="63" VerticalAlignment="Top" Height="13"/>
    </Border>
    <Border x:Name="border7" BorderBrush="Black" BorderThickness="1" CornerRadius="1"
      Background="#FF54A4DE" Padding="2" Margin="0" Width="70"
      HorizontalAlignment="Left" VerticalAlignment="Top">
      <Image x:Name="img_StateBratislava" Margin="0" HorizontalAlignment="Center"
        VerticalAlignment="Center" RenderTransformOrigin="1.387,-0.433"
        Source="img/icon/sky/11.png" Stretch="Fill" Width="38" Height="38"/>
    </Border>
  </StackPanel>
</Border>
```

# Příloha D

## Obsah CD

Elektronická verze textu bakalářské práce ve formátu PDF se nachází v root adresáři přiloženého CD. Na tomto médiu se dále nachází tři adresáře, jejichž obsah je následující:

### Adresář doc

- `doc/html` – Programová dokumentace k aplikaci ve formátu HTML.
- `doc/pdf` – Programová dokumentace k aplikaci ve formátu PDF.

### Adresář src

- `src/app` – Zdrojové soubory aplikace psané v jazyce C#.
- `src/doc` – Zdrojové soubory programové dokumentace psané v jazyce  $\text{\LaTeX}$ .
- `src/script` – Kód skriptu pro stažení XML souborů psaný pro `bash shell`.
- `src/thesis` – Zdrojové soubory textu bakalářské práce psané v jazyce  $\text{\LaTeX}$ .

### Adresář web

V tomto adresáři se nachází instalační soubory potřebné pro zprovoznění aplikace. Postup instalace je popsán v souboru `README.txt`.