

Univerzita Pardubice

Fakulta elektrotechniky a informatiky

Monitorování dešťových srážek

Bc. Ondřej Bezdíček

Diplomová práce

2013

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Ondřej Bezdíček**
Osobní číslo: **I08345**
Studijní program: **N2646 Informační technologie**
Studijní obor: **Informační technologie**
Název tématu: **Monitorování dešťových srážek**
Zadávající katedra: **Katedra softwarových technologií**

Z á s a d y p r o v y p r a c o v á n í :

Cílem diplomové práce je návrh a implementace aplikace pro monitorování dešťových srážek a varovného systému, který bude informovat uživatele o zvýšeném nebezpečí, dle jím nastavených kritérií. A to tak, aby došlo k včasnému varování a možnosti reagovat na blížící se nebezpečí v podobě zvýšeného průtoku řeky.

V úvodní teoretické části bude uveden význam aplikace a proveden teoretický rozbor zadání. Dále budou analyzovány potřebné algoritmy pro monitorovací systém. Budou navrženy vhodné technologie pro vývoj tohoto typu aplikace.

V druhé části práce bude proveden návrh a implementace webové aplikace pro nepřetržité monitorování aktuální srážkové situace. Aplikace bude automaticky analyzovat a vyhodnocovat získaná data, porovnávat je s uživatelem nadefinovanými hodnotami a následně zasílat varovné emaily a sms zprávy. Aplikace bude napsaná v jazyce C na platformě .NET, databázovou vrstvu zajistí MS SQL Server.

Rozsah grafických prací:

Rozsah pracovní zprávy:

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

Seznam odborné literatury:

1. **WALTERS, E. R.**a kolektiv **Mistrovství v SQL Server 2008**. Computer Press Books, 2009.
2. **Christian Nagel, Bill Evjen, Jay Glynn, Karli Watson, Morgan Skinner** **C 2008 - Programujeme profesionálně**. Computer Press Books, 2009.
3. **Bill Evjen, Scott Hanselman, Devon Rader** **ASP.NET 3.5 v jazycích C a Visual Basic - Programujeme profesionálně**. Computer Press Books, 2009.

Vedoucí diplomové práce:

Ing. Soňa Neradová

Katedra softwarových technologií

Datum zadání diplomové práce:

31. října 2012

Termín odevzdání diplomové práce:

17. května 2013



prof. Ing. Simeon Karamazov, Dr.
děkan



L.S.



prof. Ing. Antonín Kavička, Ph.D.
vedoucí katedry

V Pardubicích dne 15. listopadu 2012

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 5. 8. 2013

Ondřej Bezdíček

Anotace

Diplomová práce se zabývá rozбором povodňové problematiky se zaměřením na vytvoření aplikace pro monitoring dešťových srážek a následným včasným varováním. V teoretické části jsou popsány druhy povodní, jejich dopad a možnosti omezení škod. Vytvořen algoritmus pro definování kritické hranice jednotlivých toků.

V praktické části, je naprogramována aplikace pro automatické hlídání hladin toků s možností notifikací uživatele. Pro možnost automatického hlídání je implementován algoritmus analyzovaný v teoretické části.

Klíčová slova

System včasného varování, monitoring dešťových srážek, hlídání hladiny toků

Title

Rainfall Monitoring System

Annotation

The topic of Dissertation work is analysis of flooding problems and creation of application for monitoring of rainfall with consecutive early warning system. In teoretical section, there is description of flooding types, their impacts and means of damage controls. An algorithm for defining critical water levels has been created.

In practical section, there is an application for automated monitoring of water levels with optional notifications. This application uses algorithm from teoretical section.

Key words

Early warning system, rainfall watches, flood watches

Obsah

0	Úvod	10
1	Faktory povodní.....	12
1.1	Povodeň.....	12
1.2	Povodňové mapy	13
1.3	Pojistná událost	15
2	Povodeň 2013	16
2.1	Příčiny	16
2.2	Vzestup řek.....	18
2.2	Shrnutí škod	22
3	Měrné stanice.....	23
3.1	Použité měrné stanice a jejich podrobné informace.....	23
4	Webová aplikace	30
4.1	Zdrojová data	30
4.2	Představení aplikace.....	31
4.3	Možnosti rozšíření aplikace	33
5	System včasného varování	35
5.1	Technické řešení.....	35
5.2	MVC.....	35
5.3	Razor View Engine	38
5.4	Datová struktura modelu	41
5.5	Singleton	44
5.6	Sběr dat.....	46
5.7	Báze dat v MS SQL.....	47
5.8	Relační model webové aplikace.....	50
6	Email a SMS zdarma	52

6.1	Zřízení SMS emailu	52
6.2	Nastavit přeposílání.....	53
6.3	Ověření.....	55
7	Závěr.....	56
	Seznam použité literatury	58

Seznam ilustrací a tabulek

Obrázek 1 Povodňová mapa České asociace pojišťoven (Zdroj: ČAP).	13
Obrázek 2 Povodňová mapa při interaktivním výběru (Zdroj: ČAP).	14
Obrázek 3 Detailní povodňová mapa z vygenerovaného emailu (Zdroj: ČAP).	14
Obrázek 4 Indikátor nasycení půdy k 31. 5. 2013.	18
Obrázek 5 Graf znázorňující souhrn srážek od 1. 6. 2013 do 2. 6. 2013 (Zdroj: ČHMÚ).	20
Obrázek 6 Mapa měrných stanic ze stránek ČHMÚ (Zdroj: ČHMÚ).	30
Obrázek 7 Hlavní stránka webové aplikace.	31
Obrázek 8 Vazby v MVC architektuře.	36
Obrázek 9 Interakce uživatele s MVC architekturou.	37
Obrázek 10 MVC-Service layer architektura.	37
Obrázek 11 Intellisense v podání Razor engine ve Visual Studiu 2010.	38
Obrázek 12 Jednoduchá ilustrace dvojdimenzionálního rozsahového stromu.	41
Obrázek 13 ASP.NET Membership Provider.	48
Obrázek 14 Relační model ASP.NET Providers (Zdroj: MSDN).	50
Obrázek 15 Relační model Systému včasného varování.	51
Obrázek 16 Přihlášení do „Můj T-Mobile“.	52
Obrázek 17 Aktivace SMS E-mail ve Vodafone samoobsluze.	53
Obrázek 18 Nastavení akce pro nový filtr.	54
Obrázek 19 Výsledek nastavení přeposílání.	55
Tabulka 1 Počty srážek za měsíc květen na území České republiky.	17
Tabulka 2 Počty srážek za měsíc červen na území České republiky.	19
Tabulka 3 Přehled nejpostiženějších toků v první vlně povodní.	20
Tabulka 4 Přehled napáchaných škod u největších povodní za posledních 15 let.	22
Tabulka 5 Složitosti jednotlivých operací, pro práci s vrcholy.	43
Tabulka 6 Složitosti jednotlivých operací, pro práci s vrcholy.	43

Seznam použitých zkratk

Zkratka	Význam v anglickém jazyce	Význam v českém jazyce
AJAX	Asynchronous JavaScript and XML	
ASP	Active Server Pages	Aktivní stránky serveru
ČAP		České asociace pojišťoven
ČHMÚ		Český hydrometeorologický ústav
HD	High-Definition	Vysoké rozlišení
HTML	HyperText Markup Language	Základní značkovací jazyk, ve kterém jsou psány internetové stránky a jenž popisuje jejich vzhled.
JSON	JavaScript Object Notation	
MS	MicroSoft	Jméno význačné SW firmy.
MSDN	MicroSoft Developers Network	Síť Microsoft vývojářů.
MVC	Model-View-Controller	
SEO	Search Engine Optimization	optimalizace webových stránek za účelem co nejlepšího indexu v internetových vyhledávacích
SMS	Short message service	System krátkých zpráv.
SPA		Stupně povodňové aktivity
SQL	Structured Query Language	Strukturovaný dotazovací jazyk
SVV		System včasného varování
T-SQL	Transact-SQL	
XML	eXtensible Markup Language	

0 Úvod

Práce se zaměřuje na problematiku spojenou s nebezpečím v podobě náhlého zvýšení hladin vodních toků v měrných místech. Dává si za úkol, přiblížit srozumitelnou formou problematiku s tím spojené. Objasnit, které hodnoty jsou ještě v normách a které hodnoty je zapotřebí sledovat. Díky ucelenému přehledu je možné sledovat vybraný kraj nebo konkrétní řeku a tím i možné záplavové oblasti. Základní myšlenkou této práce je fakt, že pokud znám někoho proti proudu řeky, mohu od něho získat informaci o blízcím se nebezpečí a výrazně snížit množství materiálních škod. Pro názornost uvedu konkrétní příklad.

Dva bratři žijí po proudu řeky, přibližně 30 km od sebe. Pan Pavel, který pracuje v měrné stanici a pan Karel, který žije o zmíněných 30 km níže (bráno z pohledu toku řeky) v záplavové oblasti. Přichází velký déšť a hladina řeky začíná nebezpečně rychle stoupat. Pavel je zrovna v práci a údaje vyhodnotí jako závažné a hned volá bratrovi. Karel ze zkušeností již ví, že pokud mu bratr včas zavolá, tak má přibližně 30 minut, aby přeparkoval auto na vyvýšené místo, případně přichystal pytle s pískem před okna od sklepa, než se řeka vyleje z koryta u jeho domu. Pro tentokrát Karel vše stihl přichystat a ušetřil si tak vysoušení sklepa.

Bohužel ne všichni mají stejné možnosti, jako v našem modelovém příkladu, ale mnoho lidí žije v záplavových oblastech. Zejména důležitý je faktor včasného varování a spolehlivý chod. Tyto dva podstatné faktory nelze dlouhodobě vyžadovat po člověku. Proto jsem se rozhodl alespoň trochu pomoci a snažím se sestrojít aplikaci, která by fungovala jako pan Pavel, ale ne jen pro pana Karla. Všichni, kteří ji budou chtít využít, mohou tak učinit na <http://www.systemvcasnehovarovani.cz/>. Systém je dostupný zdarma a podrobněji se s ním můžete seznámit v části „Systém včasného varování“.

V rámci praktické části, je vytvořena aplikace, která se pokusí snížit škody způsobené přírodními vlivy. Za které může příliš vysoká hladina řek a jejich následné vylití z koryt. Umožňuje to samotný návrh aplikace v podobě systému napsanému tak, aby automaticky zpracoval měřené údaje, které vyhodnocují a online poskytují měrné stanice po celé České republice.

Aby byla alespoň nějaká šance na ochranu majetku, je zvláště důležité včasné varování. Pro tyto účely byl vytvořen algoritmus pro vyhodnocení blížícího se nebezpečí. Algoritmus

je navržen tak, aby nedocházelo ke zbytečnému šíření paniky. Data byla podrobně analyzována a výsledný algoritmus by měl odrážet stav, kdy v případě zaslání varování se opravdu jedná o reálnou hrozbu, nikoli pouze o dočasné zvýšení toku. V případě, kdy hodnota spočtená algoritmem nevyhovuje, tak si uživatel sám může zvolit vlastní kritéria, na jejichž základě se bude vyhodnocovat potenciální riziko. Případně přenechat řízení čistě na systému, dle předem definovaných hodnot. V obou případech dojde při překročení limitů, k vygenerování varovného hlášení a následnému zaslání v podobě emailové zprávy dle parametrů zadaných uživatelem. V části „Email a SMS zdarma“ je popsáno, jak si zajistit zaslání SMS zdarma, pro obdržení varovných zpráv ze systému.

1 Faktory povodní

S pojmem povodeň se zcela určitě setkal každý z nás. Žijeme v zeměpisné šířce, ke které lokální záplavy prostě patří. Byly tu vždy, a pokud se nějakým zásadním způsobem nezmění klima naší planety, tak tu budou i nadále. Na úvod této práce bych chtěl shrnout hlavní faktory povodní a více rozebrat běžně užívanou terminologii. V podkapitole „Povodňové mapy“ se zaměřím především na prevenci a pomoc při výběru lokality pro budoucí bydlení. Pokud vás již povodeň zasáhla a máte pojistnou smlouvu, je pro vás určena kapitola „Pojistná událost“, ve které stručně osvětlím základní kroky pro co možná nejkvalitnější a dle možností rychlé vyřízení pojistné události.

1.1 Povodeň

Technicky vzato se pojmem povodeň rozumí, náhlé výrazné navýšení hladiny vodního toku, při kterém již voda zaplavuje území a způsobuje škody. Tento jev doprovází nezřídka i stav, kdy je podloží přesycené a voda tak nemůže přirozeným způsobem odtékat, či odtéká velmi pomalu. Povodeň lze dále dělit na přírodní nebo umělou.

Přírodní povodní se rozumí stav, kdy se zvýší hladina toku kvůli vlivu přírodních elementů, které člověk nemůže ovlivnit, jako jsou dlouhodobé vydatné či extrémní přivalové deště. Náhlé změny teplot, jež způsobují rychlé tání sněhu. Přírodní povodně zasahují především střední a velké toky. Výjimky tvoří jarní rychlé tání sněhu, které nejvíce postihuje podhorské toky a samozřejmě lokální extrémní srážky, které mohou udeřit prakticky kdekoli.

Druhou kategorií jsou povodně způsobené člověkem. Jedná se o špatně uměle upravená koryta potoků a řek, příliš nízké profily mostů, či jejich pilíře, které neumožní plynulý odtok vody a ta se následně vyleje z koryta. Dalším faktorem bývají špatné odhady přehrad a přespřílišné upouštění v očekávání budoucích srážek. Stav, který sice člověk nemůže přímo ovlivnit, ale přesto patří do této kategorie, je stav havarijní. Technická havárie na přehradě, případně dokonce její protržení je velmi nebezpečná, neboť udeří nenadále a ve velmi vysoké síle.

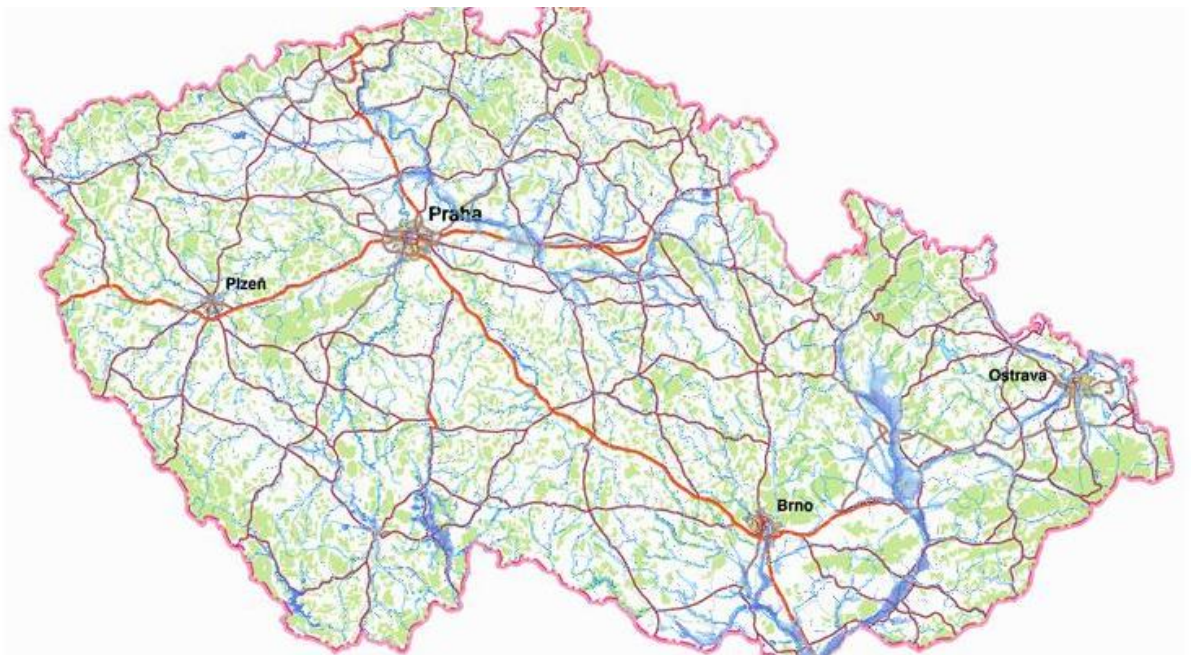
Dopady povodní pomáhají zmírnit pravidelné povodňové prohlídky, jež se soustředí na závady na vodních dílech. Tyto prohlídky provádějí povodňové orgány, občas společně se správci toku a to zpravidla během jarního tání a před obdobím letních povodní.

V České republice je navíc veřejně dostupný systém „Předpovědní a hlásné povodňové“ služby, který informuje povodňové orgány o možnosti vzniku přírodní povodně. Tento systém zabezpečuje Český hydrometeorologický ústav (ČHMÚ) [2] ve spolupráci s vodohospodáři významných vodních toků.

1.2 Povodňové mapy

Povodňové mapy České asociace pojišťoven (ČAP) [8] jsou zaměřeny na vyhodnocování pravděpodobnosti výskytu povodní. Tento systém využívají i členské pojišťovny při stanovování pojistných nebezpečí a kalkulací cen u pojistných smluv. Mapy jsou pravidelně aktualizovány a velmi dobře poslouží pro prvotní vyhodnocení, zda pozemek, o který jevíte zájem, se nachází v ohrožené oblasti či nikoli.

Informace o lokalitě můžete získat dvěma způsoby a to buď přímým vyhledáním podle adresy, anebo pomocí interaktivní mapy.



Obrázek 1 Povodňová mapa České asociace pojišťoven (Zdroj: ČAP).

Při výběru skrze interaktivní mapu, se stačí přiblížit na oblast vašeho zájmu a ihned vidíte, jak se daná oblast jeví. Modře zbarvené zóny představují zvýšené riziko záplav. Pomocí přípinacího objektu v levém horním rohu si přesně označíte konkrétní místo, viz následující obrázek.



Obrázek 2 Povodňová mapa při interaktivním výběru (Zdroj: ČAP).

Po kliknutí na další, stačí již zadat email, opsat bezpečnostní kód a vzápětí obdržíte podrobně zpracovanou zprávu, s konkrétním umístěním a barevným podáním reprezentujícím povodňové nebezpečí.



Obrázek 3 Detailní povodňová mapa z vygenerovaného emailu (Zdroj: ČAP).

V označené levé části obrázku je patrné zbarvení se středním nebezpečím výskytu povodně. Námí vybraný pozemek se tak nachází v relativní blízkosti této oblasti, ale stále ještě v bezpečí. Je třeba místo prozkoumat osobně a vzít v potaz členitost terénu s ohledem na blízký vodní tok.

Když už jsme u systému, který využívají pojišťovny, napíši sem i terminologii, jež užívají při názvosloví povodeň versus záplava. Povodní se rozumí zaplavení území vodou, která se vylila z břehu vodního toku nebo vodních nádrží. Za záplavu, je bráno souvislé pokrytí vodní plochou, které po určitou dobu stojí a nemusí být nutně způsobeno vylitím vodních toků, ale například výraznými dešťovými srážkami.

Je-li váš pozemek zařazen do některé z kategorií označující ohroženou oblast, značně se vám snižuje šance na získání pojistné smlouvy nebo naopak podstatně zvyšuje částka, kterou byste za takovou smlouvu museli platit. Ve své podstatě je to logické, neboť pojišťovny jsou komerční subjekty. Rozhodně se nespolehejte na úřady, neboť ty klidně povolí stavbu i v místě s vysokou pravděpodobností výskytu povodní.

1.3 Pojistná událost

Povodňové události v roce 2013 napáchaly nevyčíslitelné škody nejen na majetku. Některé škody se naštěstí dají zmírnit, pokud máte pojistnou smlouvu. Je jasné, že stres spojený s povodní je obrovský a proto vznikla tato podkapitola, která ve stručnosti připomene základy, podle kterých postupovat pro co možná nejrychlejší vyřízení vaší pojistné události. Jak se chovat a na co si dát pozor tak, abyste nic nepodcenili.

Krom ochrany majetku a vytvoření opatření k zabránění nárůstu škody, zabezpečte i pojistné smlouvy. Bez nich si situaci značně komplikujete. Zkontrolujte statiku a v případě větších problému si nechte vypracovat statický posudek. Prohlédněte také rozvody elektřiny, plyn a kanalizaci. Vypracujte si seznam poničených věcí s přibližnou cenou a stářím. Ideální je situace, kdy k poničené věci můžete doložit přímo účtenku. Pokud je to možné, ponechte poničené věci v původním stavu, pro případnou prohlídku pracovníkem pojišťovny. Pokud to možné není, pořídte řádnou dokumentaci. Fotografie jsou velmi důležité, a proto jejich rozsah nepodceňujte. Pomůžou vám dokazovat škody, které musí být z hygienických či bezpečnostních důvodů eliminovány co nejdříve. Jedná se například o likvidaci potravin, uhynulých zvířat apod. Pro odstranění uhynulých zvířat kontaktujte kafilerní službu skrze obecní úřad, která vám zároveň vydá potvrzení o převzetí uhynulého kusu.

Pojistnou událost nahlaste co nejdříve, podobně postižených jako jste vy je spousta a i když pojišťovny v případě povodní zvyšují počty pracovníků na vyřizování pojistných událostí, přeci jen není příliš dobré být až devítitisící v řadě. Při hlášení pojistné události mějte připravené číslo pojistné smlouvy (někde postačí i rodné číslo), stručně popište pojistnou událost, odhad škod a datum kdy k události došlo.

2 Povodeň 2013

Během tvorby této diplomové práce, došlo na přelomu května a června k rozsáhlým povodním zejména v Čechách. Povodně jsem bedlivě sledoval a v této kapitole bych vás rád seznámil s postřehy, které se mi podařilo zaznamenat a které po následné selekci považuji za důležité. Informace jsem doplnil oficiálními daty z ČHMÚ [2] pro názornost.

2.1 Příčiny

Počátky silných červnových povodní mají svůj základ již v měsíci květnu, který byl srážkově nadprůměrný. Jednalo se o 113 mm, což je o 52% více, v porovnání s dlouhodobým srážkovým normálem z let 1961 – 1990, který je 74 mm. Nejvíce srážek spadlo v posledním květnovém týdnu. V průměru přes 100 mm, v nejpostiženějších oblastech překročily i hranici 180 mm. Tyto srážky zasáhly hlavně západ našeho území, což způsobilo, že hladiny toků začaly stoupat a na některých místech byl dokonce vyhlášen první stupeň povodňové aktivity, tedy bdělost.

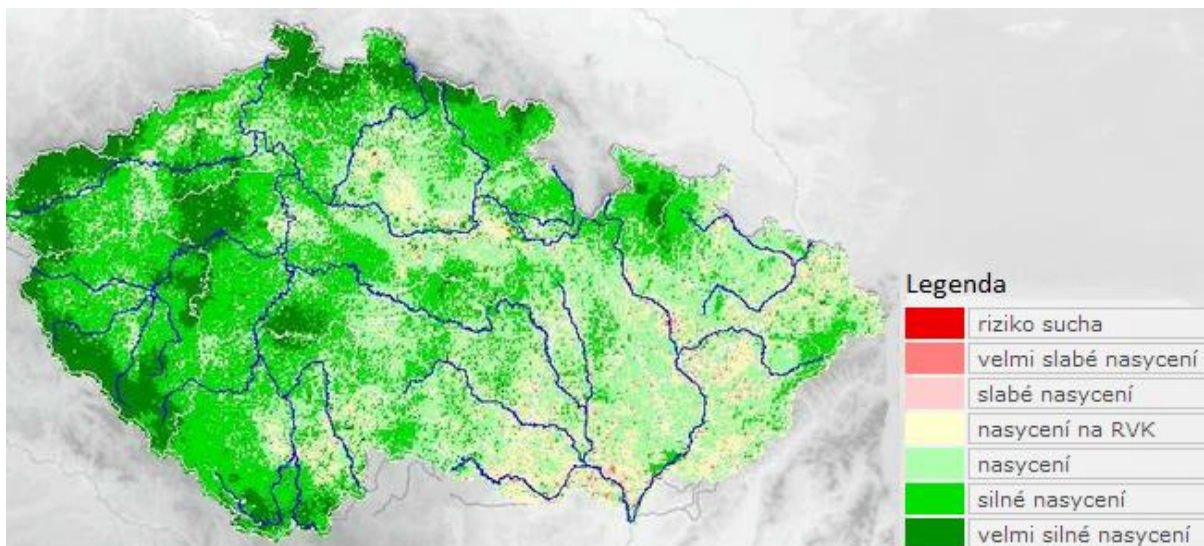
V měsíci květnu byly průměrné srážky nejvyšší v krajích Karlovarský (125 mm), Ústecký (121 mm), Plzeňský (122 mm), Jihočeský (121 mm) a Středočeský včetně Prahy (113 mm).

Podrobné informace ze všech krajů za měsíc květen, si můžete prohlédnout v následující tabulce 1.

Kraj		2013	Kraj		2013
		Květen			Květen
Praha a Středočeský	Srážky	113	Pardubický	Srážky	113
	Normál	70		Normál	77
	Rozdíl v %	161		Rozdíl v %	147
Jihočeský	Srážky	121	Vysočina	Srážky	113
	Normál	75		Normál	76
	Rozdíl v %	161		Rozdíl v %	148
Plzeňský	Srážky	122	Jihomoravský	Srážky	99
	Normál	70		Normál	65
	Rozdíl v %	175		Rozdíl v %	153
Karlovarský	Srážky	125	Olomoucký	Srážky	101
	Normál	61		Normál	80
	Rozdíl v %	205		Rozdíl v %	126
Ústecký	Srážky	121	Zlínský	Srážky	97
	Normál	61		Normál	82
	Rozdíl v %	198		Rozdíl v %	119
Liberecký	Srážky	107	Moravskoslezský	Srážky	112
	Normál	79		Normál	94
	Rozdíl v %	135		Rozdíl v %	119
Královéhradecký	Srážky	113	Česká republika	Srážky	113
	Normál	76		Normál	74
	Rozdíl v %	148		Rozdíl v %	152

Tabulka 1 Počty srážek za měsíc květen na území České republiky.

Dalším důležitým faktorem byla dlouhá zima a s ní spojená vysoká sněhová pokrývka v horských oblastech, jež začala tát až v průběhu dubna. To v kombinaci s vytrvalými dešti vedlo k naplnění hrází. Následné upouštění hladin pro udržení kapacity a hlavně připravenosti retenčních nádrží vedlo k nasycení povodí. Přesycené podloží se vyskytovalo zejména v Karlovarském, Plzeňském, Ústeckém, Libereckém a Středočeském kraji. Výrazného nasycení dosáhl také Jihočeský kraj. Na následujícím obrázku je dobře patrná míra nasycení půdy k poslednímu dni měsíce května.



Obrázek 4 Indikátor nasycení půdy k 31. 5. 2013.

Přesycené území a neutuchající srážky měli za důsledek velmi rychlý nástup červnových povodní. V tuto chvíli se ještě nevědělo, že povodeň udeří ve dvou vlnách.

2.2 Vzestup řek

Vydatné srážky pokračovaly i začátkem června, který zvláště ve svém počátku byl velmi deštivý. I měsíc červen pak ve výsledných statistikách výrazně překonal dlouhodobý průměr a to dokonce o celých 75%. V přímém porovnání s květnem, pak v průměru spadlo ještě o celých 33 mm dešťových srážek více.

Intenzita srážek dosahovala až 46 mm za hodinu, což výrazně překračuje průměrná historická čísla pro tyto měsíce. Nejvíce postiženy byly oblasti jižně od Prahy, jež v kontrastu s mírou nasycení vedlo k rychlému rozvodnění menších toků. Nedlouho na to se začínají objevovat první varování třetího stupně povodňové aktivity, tedy stupeň ohrožení.

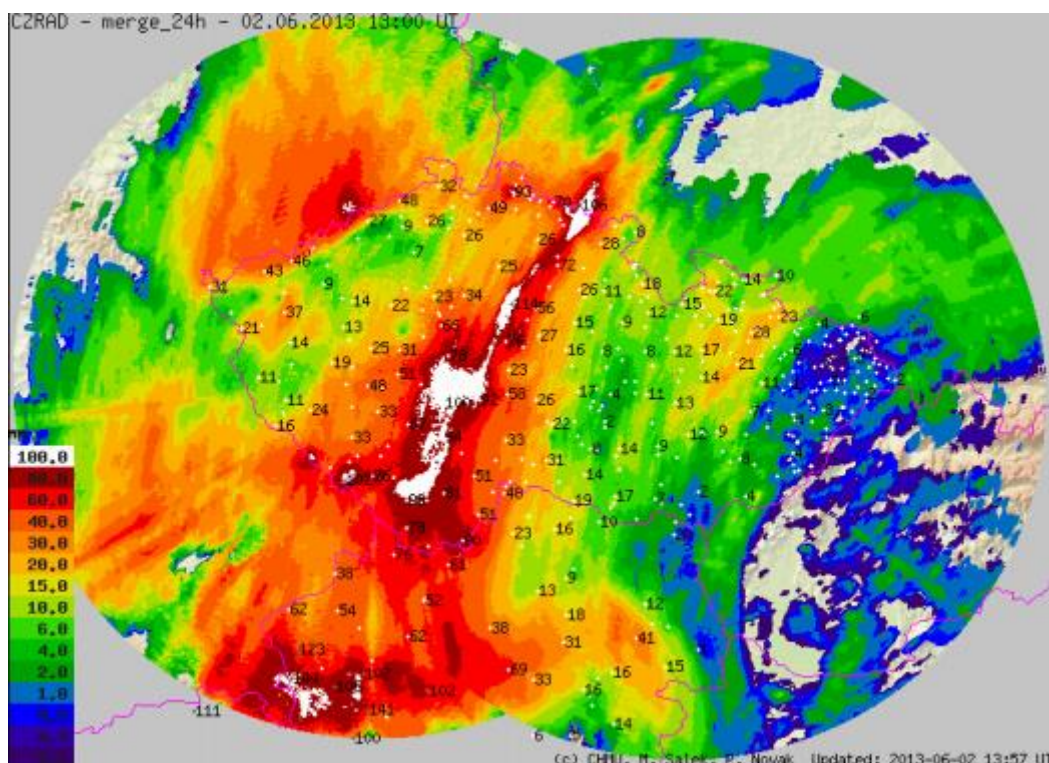
Podrobné informace ze všech krajů za měsíc červen, si můžete prohlédnout v tabulce 2.

Kraj		2013	Kraj		2013
		Červen			Červen
Praha a Středočeský	Srážky	163	Pardubický	Srážky	122
	Normál	75		Normál	87
	Rozdíl v %	217		Rozdíl v %	140
Jihočeský	Srážky	187	Vysočina	Srážky	137
	Normál	94		Normál	82
	Rozdíl v %	199		Rozdíl v %	168
Plzeňský	Srážky	126	Jihomoravský	Srážky	117
	Normál	78		Normál	75
	Rozdíl v %	161		Rozdíl v %	157
Karlovarský	Srážky	139	Olomoucký	Srážky	134
	Normál	75		Normál	94
	Rozdíl v %	185		Rozdíl v %	142
Ústecký	Srážky	141	Zlínský	Srážky	117
	Normál	68		Normál	102
	Rozdíl v %	207		Rozdíl v %	114
Liberecký	Srážky	175	Moravskoslezský	Srážky	156
	Normál	83		Normál	108
	Rozdíl v %	211		Rozdíl v %	144
Královéhradecký	Srážky	151	Česká republika	Srážky	146
	Normál	86		Normál	84
	Rozdíl v %	176		Rozdíl v %	174

Tabulka 2 Počty srážek za měsíc červen na území České republiky.

Ze zmíněných údajů je jasné, co muselo zákonitě následovat. První povodňová vlna na sebe bohužel nenechala dlouho čekat a udeřila se smrtící rychlostí. Pro zajímavost ukáži i graf, který se nevidí příliš často a na kterém, je dobře patrné jakou sílu měli tyto deště. Zejména pak na jihu Čech.

Souhrn srážek je dobře viditelný na následujícím obrázku. Bíle zbarvená oblast protínající Čechy značí velmi silné srážky, přesahující v průměru 100 mm s nejvyšší intenzitou na jihu Čech.



Obrázek 5 Graf znázorňující souhrn srážek od 1. 6. 2013 do 2. 6. 2013 (Zdroj: ČHMÚ).

Nárůstem vodního toku Berounky a v důsledku odpuštění Vltavských kaskád (pro oddálení kulminace vody), byl vyhlášen první stupeň povodňové aktivity i v Praze. V rámci rychlé reakci kaskád, byl dán prostor pro postavení protipovodňových bariér. První povodňová vlna kulminovala na většině území mezi 3. - 6. červnem. Nejpozději však kolem 11. června.

V následující tabulce jsou zobrazeny nejvíce postižené toky, podle n-leté vody.

Tok	Měrná stanice	Datum kulminace	Stav	Limit aplikace	Průtok	Průtok při 10-leté vodě	N-letá voda
Chotýšanka	Slověnice	2.6.2013	270	147	76	29	500
Blanice	Radonice I	2.6.2013	504	N/A	189	75,2	500
Labe	Vestřev	2.6.2013	354	151	310	162	100
Labe	Mělník	5.6.2013	936	535	3800	2520	100
Mrlina	Vestec	3.6.2013	314	218	82,5	45,6	100
Výrovka	Plaňany	2.6.2013	454	230	100	38,2	100
Smutná	Rataje	2.6.2013	349	250	136	62	100
Lužnice	Bechyně	2.6.2013	594	316	561	308	100
Mastník	Radíč	2.6.2013	282	167	103	35,8	100
Brzina	Hrachov	2.6.2013	259	N/A	79,6	32,2	100
Kocába	Štěchovice	2.6.2013	248	N/A	101	46,3	100

Tabulka 3 Přehled nejpostiženějších toků v první vlně povodní.

Ve sloupci stav je zobrazen údaj o výšce hladiny toku v den kulminace v centimetrech. Limit aplikace ukazuje vypočítané hodnoty pro automatický systém včasného varování. V prvotních verzích algoritmu byl výpočet limitu příliš benevolentní a tak nemuselo dojít k včasnému varování. Data jsem podrobně studoval a následně algoritmus vylepšil tak, aby vyhodnocení rizika bylo co nejpřesnější. Z tabulky 3 je patrné, že přednastavený limit aplikace by v aktuální podobě splnil svůj účel a to ve všech případech. Varovná zpráva bude vygenerována s dostatečným předstihem a přesto ne příliš brzy na to, aby budila obavy ve chvílích kdy to ještě není nutné. Pokud je zapsána hodnota N/A, tak uvedená měrná stanice není v systému dostupná. Sloupec průtok zobrazuje sílu průtoku v den kulminace v jednotkách $[\frac{m^3}{s}]$. Průtok při 10 leté vodě, nám přidává porovnání, jaká hodnota je ještě v pořádku. V posledním sloupci je údaj, který nám vyjadřuje sílu povodně, znázorněnou při n-letém opakování.

Když už se zdálo, že je záplavám konec a lidé začali odklízet nánosy bahna a sčítat škody, udeřily lokální přívalové deště, které vzedmuly druhou povodňovou vlnu. Země byla stále ještě silně nasáklá a tak velké vodě nic nekladlo překážky a ta měla znovu rychlý vzestup. V centru se tentokrát ocitlo Chrudimsko, kde během dvou dní spadlo více než 100 mm srážek. V globálu pak byly zasaženy větší plochy a tak došlo k lepšímu odvodnění a tím pádem i mírnějšímu průběhu. Druhá vlna tak byla vlna o poznání slabší a k její kulminaci došlo na většině území kolem 26. června. Svá maxima dosahovaly toky v podobě 50 letých vod.

2.2 Shrnutí škod

Závěrem této kapitoly bych shrnul povodně 2013 v číslech. Na pomoc se stavbou protipovodňových zábran a na následném odstraňování způsobených škod se podílelo více než 10 000 vojáků s 2 000 kusů vojenské techniky. Na záchranných a likvidačních pracích se podílelo více než 28 000 profesionálních i dobrovolných hasičů a přes 21 000 policistů. Zachráněno bylo přes 600 lidí a dalších více než 26 000 muselo být evakuováno. Povodně si i přes veškeré úsilí vyžádaly 15 lidských životů a zasáhli přibližně 970 obcí a měst. Výše škod se odhaduje na desítky miliard korun a jen zhruba čtvrtina z nich byla nějakým způsobem pojištěna. Pojišťovny hlásí více jak 39 000 nahlášených pojistných událostí s celkovou cenou přesahující 7 miliard korun. V konečném důsledku se tato povodeň nejvíce podobá té, z roku 2002. Jak příčinami, tak i konečnými počty obětí. Jen škod na majetku bude letos výrazně méně.

Povodňové škody v Česku		
povodeň	povodňové škody (mil. Kč)	ztráty na životech
1997	62 600	60
1998	1 800	10
2000	3 800	2
2001	1 000	0
2002	75 100	16
2006	6 200	9
2009	8 500	15
2010	15 000	6
2013	cca 30 000	15

Tabulka 4 Přehled napáchaných škod u největších povodní za posledních 15 let.

3 Měrné stanice

V současné době se na území České republiky nachází 362 měrných míst. Měrné stanice pravidelně aktualizují údaje o výšce hladiny toku v centimetrech. Většina z nich podává informace i o rychlosti průtoku řeky v metrech krychlových za sekundu. Pro systém včasného varování jsem vybral 23 měrných stanic, které jsem podrobil analýze a vybral z nich pro nás zajímavé údaje. Toky jsem vybíral dle geografického umístění tak, aby byla pokryta celá Česká republika a samozřejmě se zaměřením na více riziková místa. Systém jsem postavil tak, abych v budoucnu mohl síť poskytovaných informací dále rozšiřovat.

V následujícím přehledu ukáži jednotlivé vybrané toky s jejich limity povodňové aktivity, které vyhodnotil ČHMÚ [2] v porovnání s daty, jež jsem analyzoval já. Mnou vyhodnocená data, jsou použita v aplikaci včasného varování, pro automaticky přednastavený limit, viz praktická část diplomové práce. Vzorec použitý pro výpočet přednastaveného limitu se opírá o průtok síly desetileté vody, jež považuji za zvláště vhodný ukazatel povodňové aktivity. Zmíněný vzorec, podrobně rozepsaný vypadá následovně:

$$\frac{p_q + o_q}{2} = Q_a \quad \frac{Q_{10}}{Q_a} = Q \quad \frac{p + o}{2} * \left(\frac{Q}{100} + 1\right) + C = L$$

L - vypočítaný automaticky přednastavený limit, p - pohotovost, o - ohrožení, p_q - pohotovost při průtoku, o_q - ohrožení při průtoku, Q_{10} - průtok při desetileté vodě, C - konstanta modifikující vzorec na základě zkušeností uživatelů

3.1 Použité měrné stanice a jejich podrobné informace

V této podkapitole podrobně rozeberu zanalyzovaná data v podobě jejich sumarizace do přehledných a ucelených bloků. Dozvíte se z nich název použitého toku, v jaké obci se nachází měrná stanice, limity pro průtoky v intervalech každoroční, až po pojem stoletá voda a její hodnotu průtoku. Dále jsou tu limity jednotlivých stupňů povodňových aktivit dle nastavení ČHMÚ a vypočítaná hodnota limitu aplikace. Jelikož je tato hodnota počítána dynamicky, tak je zde uvedena spíše formou ilustrativní, neboť se v průběhu času a po automatických aktualizacích dat může změnit. Její aktuální hodnotu si můžete vždy zobrazit na webové prezentaci samotné aplikace.

Tok:	Ohře	Stanice:	Karlovy Vary - Drahovice			
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	150	297	369	556	645
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		190		153		
Pohotovost:		230		208		
Ohrožení:		270		271		
Přednastavený limit aplikace:		254		369		
Poznámka:						

Tok:	Ohře	Stanice:	Louny			
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	251	466	568	767	942
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		400		157		
Pohotovost:		440		190		
Ohrožení:		490		235		
Přednastavený limit aplikace:		477		568		
Poznámka:						

Tok:	Labe	Stanice:	Mělník			
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	1080	2060	2520	3640	4150
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		400		814		
Pohotovost:		500		1200		
Ohrožení:		550		1440		
Přednastavený limit aplikace:		535		2520		
Poznámka:						

Tok:	Berounka	Stanice:	Beroun			
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	270	615	799	1310	1560
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		260		210		
Pohotovost:		320		308		
Ohrožení:		400		467		
Přednastavený limit aplikace:		367		799		
Poznámka:						

Tok:	Berounka	Stanice:		Bílá Hora		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	124	295	389	655	790
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		250		69,6		
Pohotovost:		350		133		
Ohrožení:		450		262		
Přednastavený limit aplikace:		408		389		
Poznámka:						

Tok:	Lužická Nisa	Stanice:		Liberec		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	16,3	46	64,2	120	150
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		85		9,1		
Pohotovost:		125		22		
Ohrožení:		160		39,9		
Přednastavený limit aplikace:		145		64,2		
Poznámka:						

Tok:	Labe	Stanice:		Vestřev		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	59	127	162	256	301
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		110		39,7		
Pohotovost:		135		56,6		
Ohrožení:		160		75,2		
Přednastavený limit aplikace:		151		162		
Poznámka:						

Tok:	Mrlina	Stanice:		Vestec		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	15,3	35	45,6	75,1	90
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		150		14,2		
Pohotovost:		190		24,5		
Ohrožení:		240		42,2		
Přednastavený limit aplikace:		218		45,6		
Poznámka:						

Tok:	Výrovka	Stanice:		Plaňany		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	12,8	29,3	38,2	62,7	75,1
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		150		10,2		
Pohotovost:		200		14,9		
Ohrožení:		250		20,2		
Přednastavený limit aplikace:		230		38,2		
Poznámka:						

Tok:	Chrudimka	Stanice:		Nemošice		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	50,2	99,2	123	186	215
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		150		36,7		
Pohotovost:		180		49,6		
Ohrožení:		220		68,6		
Přednastavený limit aplikace:		204		123		
Poznámka:						

Tok:	Chotýšanka	Stanice:		Slověnice		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	11,5	23,2	29	44,2	51,5
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		100		6,6		
Pohotovost:		130		12,3		
Ohrožení:		160		21,8		
Přednastavený limit aplikace:		147		29		
Poznámka:						

Tok:	Mastník	Stanice:		Radič		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	7,8	24,8	35,8	71,3	91,7
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		100		9,4		
Pohotovost:		150		20,9		
Ohrožení:		180		30,3		
Přednastavený limit aplikace:		167		35,8		
Poznámka:						

Tok:	Orava	Stanice:					Katovice
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100	
	$[\frac{m^3}{s}]$	133	227	280	432	510	
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$			
Bdělost:		180		121			
Pohotovost:		220		171			
Ohrožení:		280		257			
Přednastavený limit aplikace:		253		280			
Poznámka:							

Tok:	Smutná	Stanice:					Rataje
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100	
	$[\frac{m^3}{s}]$	21	46	62	108	132	
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$			
Bdělost:		170		12,5			
Pohotovost:		210		21,4			
Ohrožení:		250		37,2			
Přednastavený limit aplikace:		235		62			
Poznámka:							

Tok:	Lužnice	Stanice:					Bechyně
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100	
	$[\frac{m^3}{s}]$	111	241	308	488	577	
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$			
Bdělost:		240		91,3			
Pohotovost:		290		135			
Ohrožení:		330		172			
Přednastavený limit aplikace:		316		308			
Poznámka:							

Tok:	Vltava	Stanice:					České Budějovice
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100	
	$[\frac{m^3}{s}]$	172	350	452	751	908	
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$			
Bdělost:		300		244			
Pohotovost:		370		361			
Ohrožení:		430		489			
Přednastavený limit aplikace:		404		452			
Poznámka:							

Tok:	Jihlava	Stanice:		Dvorce		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	16	30	37	54	62
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		120		12,9		
Pohotovost:		160		21,7		
Ohrožení:		210		34,8		
Přednastavený limit aplikace:		187		37		
Poznámka:						

Tok:	Svratka	Stanice:		Brno - Poříčí		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	51,1	110	142	235	283
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		100		19,7		
Pohotovost:		160		54,9		
Ohrožení:		260		142		
Přednastavený limit aplikace:		213		142		
Poznámka:						

Tok:	Svratka	Stanice:		Židlochovice		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	117	208	250	353	400
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		250		80,5		
Pohotovost:		300		103		
Ohrožení:		350		126		
Přednastavený limit aplikace:		332		250		
Poznámka:						

Tok:	Morava	Stanice:		Olomouc – Nové Sady		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	135	258	319	476	551
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		360		147		
Pohotovost:		390		167		
Ohrožení:		430		197		
Přednastavený limit aplikace:		417		319		
Poznámka:						

Tok:	Dřevnice	Stanice:		Zlín		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	70	137	168	251	290
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		170		64,1		
Pohotovost:		200		82,6		
Ohrožení:		260		123		
Přednastavený limit aplikace:		234		168		
Poznámka:						

Tok:	Opava	Stanice:		Opava		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	45,6	124	171	312	388
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		250		64,5		
Pohotovost:		300		102		
Ohrožení:		350		151		
Přednastavený limit aplikace:		329		171		
Poznámka:						

Tok:	Ostravice	Stanice:		Ostrava		
N-leté průtoky:		Q1	Q5	Q10	Q50	Q100
	$[\frac{m^3}{s}]$	186	431	565	936	1120
Stupně povodňové aktivity:	[cm]			$[\frac{m^3}{s}]$		
Bdělost:		290		182		
Pohotovost:		400		378		
Ohrožení:		530		665		
Přednastavený limit aplikace:		470		565		
Poznámka:						

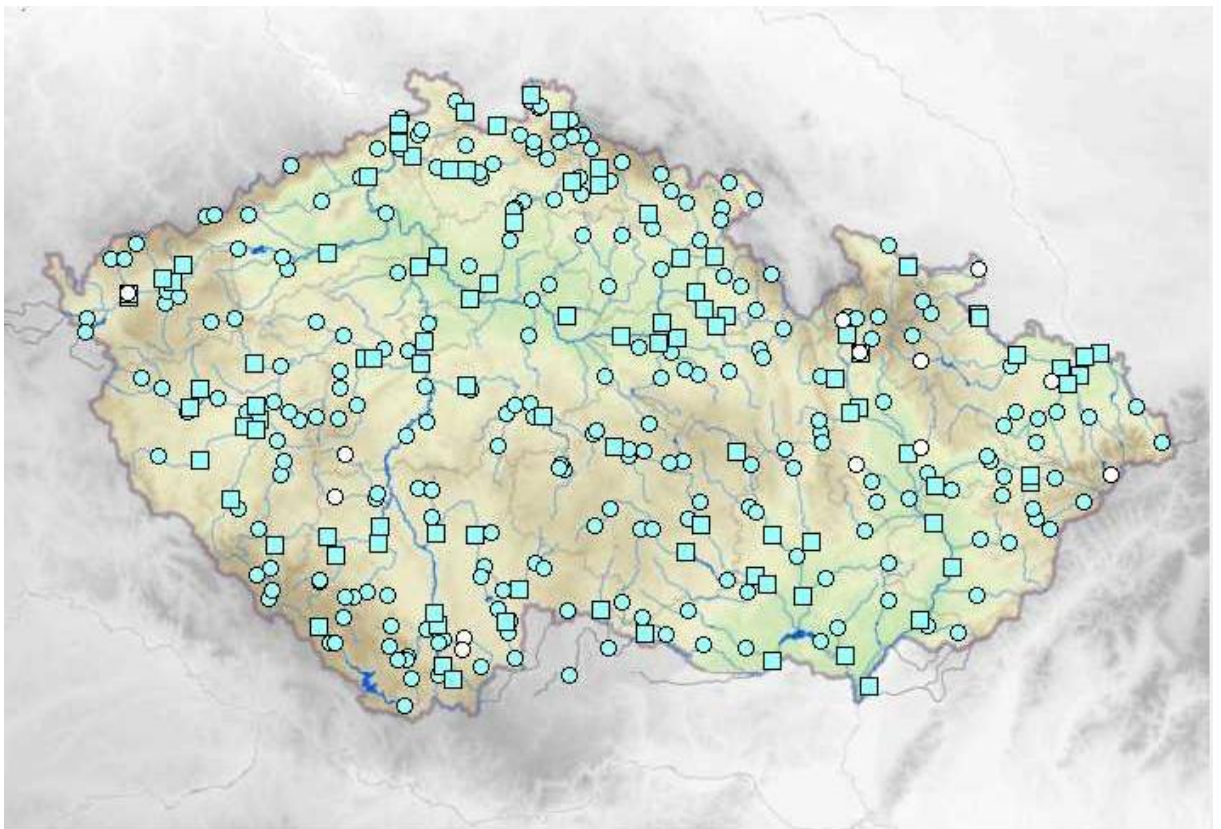
4 Webová aplikace

V rámci praktické části diplomové práce je daný úkol přehlednou a srozumitelnou formou analyzovaná data prezentovat. K tomu byla vytvořena webová aplikace, s níž vás podrobněji seznámím v nadcházejících kapitolách.

4.1 Zdrojová data

V této mini kapitole, bych chtěl napsat něco o původu dat, které využívám pro svůj systém. Zdrojová data jsou automaticky v pravidelném intervalu načítána ze stránek ČHMÚ [2] z veřejně dostupné „Hlásné a předpovědní povodňové služby“. Tato data jsem podrobně studoval a považuji je za velmi kvalitní zdroj informací. Měrné stanice s dostatečnou rychlostí aktualizují potřebné informace a jejich stabilita a kvalita je na velmi vysoké úrovni.

Možná vás nyní napadne, proč vlastně využívat tuto aplikaci a ne přímo zdrojové stránky.



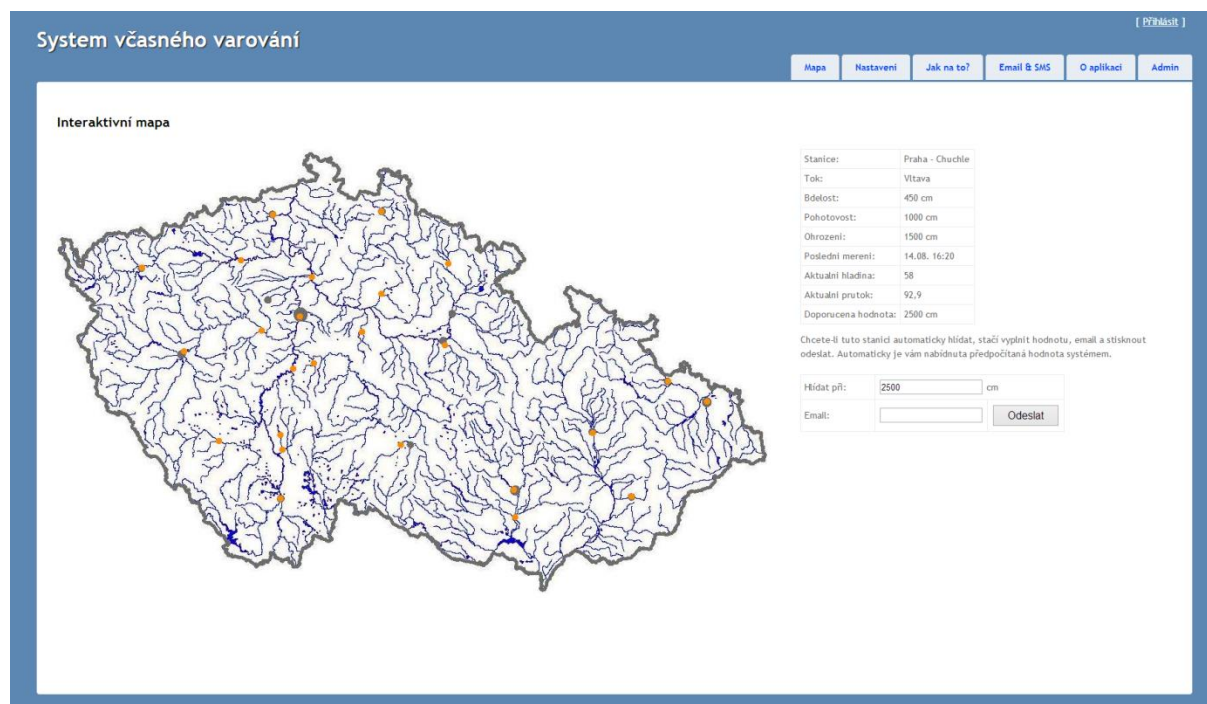
Obrázek 6 Mapa měrných stanic ze stránek ČHMÚ (Zdroj: ČHMÚ).

Mým cílem je aplikaci postavit tak, aby byla co možná nejjednodušší pro ovládání, plně intuitivní a přesto vysoce efektivní v tom, na co je primárně zaměřena. Stránky ČHMÚ mají velmi široký záběr a plně se v nich orientovat, chce spoustu času a zkušeností.

Jelikož ne každý je počítačový expert, postavil jsem tento systém s hlavním zaměřením na koncové uživatele. Základní představu o funkčnosti systému získáte po přečtení následujících kapitol.

4.2 Představení aplikace

Webovou prezentaci si můžete spustit na adrese www.systemvcasnehovarovani.cz a její úvodní stránka vypadá následovně.



Obrázek 7 Hlavní stránka webové aplikace.

Design stránek byl koncipován tak, aby na první pohled prezentoval užití MVC ASP.NET aplikace [5]. Barevné podání i rozmístění jednotlivých prvků je pro toto prostředí typické. Vyvíjen byl na full HD rozlišení 1080p (1920x1080px) s dodržением pravidel přístupnosti a tudíž by měl být plně funkční i na nižších rozlišeních. Jen je možné, že se změnil rozmístění jednotlivých bloků oproti zobrazenému vzoru na obrázku 7.

Systém je koncipovaný na nenucené registraci. Veškerá funkčnost je dostupná bez zaregistrování a samotné zaregistrování (pokud tak neučiníte sami v menu „Přihlásit“) se provede až ve chvíli, kdy na úvodní stránce dáte odeslat svůj email, protože vyžadujete automatické hlídání některé ze stanic. Automatická registrace je nutná z důvodu pozdější editace nastavených limitů v menu „Nastavení“, případně k zrušení hlídání určité stanice, nebo zrušení celého účtu. Na uvedený email je zaslán přihlašovací jméno (má podobu

samotného emailu) a vygenerováno náhodné heslo. Pokud si limity nepřejete měnit, stále můžete aplikaci ovládat bez přihlášení a mít tak skryto nastavovací menu. Poskytovaný email je výhradně použit pouze pro účely zasílání vyžádaných automatických varovných zpráv a přihlašování do aplikace. Není poskytován třetím stranám, ani zneužit k zasílání nevyžádaných zpráv. Jediný další účel, ke kterému může být ve výjimečných situacích použit, je případ důležitých hromadných zpráv týkajících se výhradně aplikace samotné. Jedná se například o zásadní změny v systému (ovlivňující stávající funkčnost, nikoli jen běžné aktualizace), hromadné varování a podobně významné informace.

V pravém horním rohu je možnost přihlášení a hned pod ním, se nachází hlavní menu aplikace, které obsahuje následující položky:

- Mapa – Zobrazuje úvodní stránku aplikace a je také nejdůležitějším uživatelským menu. Obsahuje interaktivní mapu s oranžově zbarvenými body, jež znázorňují jednotlivé měrné stanice. Po kliknutí na některý z bodů, se naplní tabulka s údaji o vybrané měrné stanici. Pod tabulkou se nachází dvě textová pole, pomocí nichž si můžete nastavit, při jaké hladině vás má systém automaticky varovat, na zadaný email. Systém vás bude varovat pouze při prvním výskytu překročení limitu a následně každou hodinu, pokud je hladina překročena. Interval doručování zpráv si můžete změnit v menu nastavení.
- Nastavení – Je přístupné pouze po přihlášení a umožňuje spravovat dříve nastavené hodnoty automatického hlídání. Buď jejich editaci anebo jejich zrušení.
- Jak na to? – Základní manuál k obsluze aplikace. Je zde sepsáno jak s aplikací pracovat, co očekávat za výstup a další pomocné informace. Aplikace byla již od začátku psána se záměrem co nejjednodušší orientace uživatele, tak aby plnila svůj konkrétní úkol jednoduše, ale spolehlivě.
- Email & SMS – Zde naleznete návod, jak si nechat posílat varovné SMS zprávy na váš mobilní telefon.
- O aplikaci – Pár stručných slov o aplikaci a jejím tvůrci. Je zde shrnuta motivace, jež mě k vytvoření takovéto aplikace vedla a ostatní obecné informace.
- Admin – Toto menu slouží k administrativním účelům a je běžně nedostupné. Umožňuje přidávat nové stanice, manuálně spustit job pro znovu načtení dat a další funkce spojené s nutnou administrací webové prezentace.

4.3 Možnosti rozšíření aplikace

V této kapitole jsou rozepsané funkcionality, které jsou nad rámec cílů diplomové práce. Některé jsem již dokončil a implementoval, na některých aktuálně pracuji a ostatní jsou plánovány do budoucna.

- Rozšíření aplikace o dalších 100 – 150 měrných stanic. Tento úkol mám prakticky hotový a stanice mohou kdykoli přidat. Jediné co mě aktuálně brzdí, je fakt, že kladu vysoký důraz na bezchybný provoz aplikace. Proto je nyní simulováno jen okolo 23 stanic. Je důležité odladit případné nedostatky dříve, než systém nabobtná do rozměrů, které by byly manuálně neudržitelné. Po přidání takového množství stanic, je nutné, aby byl systém již odladen a fungoval zcela bezchybně a samostatně.
- Umožnit přiblížení mapy na vybranou oblast. Aktuálně se jedná o funkcionality, které věnuji nejvíce času. K tomuto účelu jsem napsal algoritmy pro intervalové vyhledávání v 2D range tree (nejsou součástí diplomové práce). Pomocí intervalového vyhledávání, jsem schopný identifikovat měrné stanice uvnitř intervalu, ale zatím se mi nedaří je virtuálně přepočítat pro správné umístění na zvětšené mapě. To je důležité pro následnou interakci při klikání na konkrétní stanice.
- Změna designu stránek. Kamarádka profesionálního grafika, jsem požádal o vytvoření zajímavého a tematického designu stránek, který poté převedu do kaskádových stylů a přizpůsobím tomu webové prvky. Tato operace není z mého pohledu příliš časově náročná, jen je závislá na jeho pracovním vytížení. Jedná se o návrh zdarma, takže žádný pevný termín nemohu ani požadovat. Každopádně bych se na design stránek chtěl zaměřit. Aktuálně použitý vzor je tvořen záměrně pro prezentaci práce, aby bylo na první pohled jasné, že se jedná o použití ASP.NET MVC Framework.
- Moderování stránek. Při prezentaci aplikace kamarádům, se vyskytla otázka, zda se dá v dlouhodobém horizontu vše udržet v chodu při pouze mé interakci. Tyto poznámky se mi zaryli pod kůži a jak sem tak nad nimi zauvažoval, musel jsem jim dát za pravdu. Vše obsluhovat v horizontu několika let a při tom zvládat běžnou práci, rodinu a život nebude jednoduché. Také určitě není žádoucí, aby takovýto systém závisel pouze na jednom člověku. Pozadí aplikace, jako je stahování dat a rozesílání informačních emailů je prakticky bezúdržbová činnost a byla s tímto důrazem i programována. Ale co případné úpravy koeficientu podle měnící se podoby toků, nebo přidávání nových stanic. K takovým úkolům je třeba spousta času, pro řádné analýzy a správné vyhodnocení údajů. Proto vytvořím moderátorskou část, která bude

mít rozšířený přístup do aplikace a ověření uživatelé tak sami budou moci upravovat globální koeficient u jejich měrné stanice. Myšlenka je založena na tom, že člověk, který v ohroženém místě bydlí a projeví zvýšený zájem o pomoc s tímto projektem, sám nejlépe může rozhodnout, kdy je třeba ostatní na stejné stanici varovat. Bude tak moci upravit automaticky počítaný vzorec o koeficient dle svého svědomí. Toto ovlivní jen automaticky navrhované hodnoty té konkrétní stanice. Nijak se nedotkne uživatelů, kteří si zaregistrovali hlídání na své konkrétní hodnotě.

- Search Engine Optimization (SEO). Ve zkratce se jedná o optimalizaci a zpracování obsahu webové prezentace takovým způsobem, aby se v rámci rozhodovacích algoritmů, jednotlivých vyhledávačů, stránky umístili na předních pozicích. S touto problematikou mám již bohaté zkušenosti z bakalářského studia, kdy se mi podařilo optimalizovat svou práci tak, aby na vybrané klíčové slovo (Jak na Photoshop) se stránka umístila na prvním místě ve vyhledávači Google. To vedlo k takřka 36 000 unikátním přístupům za bezmála 3 letý provoz. Když se vezme v úvahu, že se jedná o specificky zaměřený web, tak to vůbec není špatné číslo.

Takto bych shrnul aktuální plánované změny, které jsem v průběhu psaní této práce navrhl s ohledem na zlepšení uživatelského komfortu při práci se systémem. V době, kdy čtete tyto stránky, je pravděpodobné, že alespoň některé body jsou již implementovány na serveru aplikace a běžně je využíváte.

5 System včasného varování

Vytvořená aplikace, si klade za svůj hlavní cíl orientaci na uživatele. V zásadě je třeba docílit stavu, kdy ovládání bude jednoduché a intuitivní. Přesto však musí aplikace uživateli poskytnout vysokou míru přidané hodnoty. V této kapitole, bych rád rozebral jednotlivé etapy vývoje, jež umožňují splňovat vytyčený cíl.

5.1 Technické řešení

Aplikace je napsána v jazyce C# pod platformou .NET Framework 4 [7] s využitím ASP.NET MVC 3 Razor [5]. Jako vývojové prostředí jsem využil MS Visual Studio 2010, jež považuji za téměř dokonalý nástroj pro vývojáře v jazyce C#. Jako bázi dat jsem zvolil redakční databázový systém MS SQL. V následujících kapitolách ve stručnosti osvětlím použité technologie při vývoji aplikace. Architekturu MVC a view engine Razor.

5.2 MVC

V této podkapitole, vám ve stručnosti představím co to vlastně je a v jaké podobě jsem jí použil pro tvorbu aplikace. Architektura MVC (Model-View-Controller) jsou v podstatě 3 logické části, které se dají programovat odděleně, a jakákoli část se dá nahradit jinou za dodržení určitých pravidel. Pro lepší pochopení uvedu následující příklad.

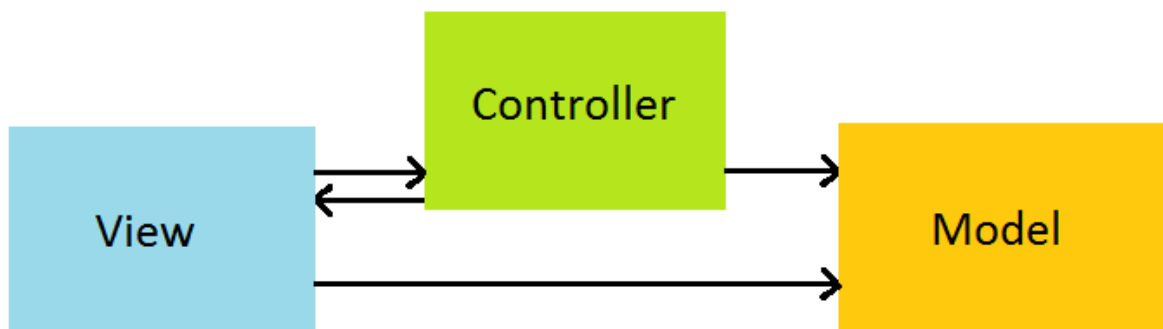
V části Model, mám naprogramovanou logiku pro připojení se k MySQL databázi, ale po nějakém čase se rozhodnu přejít na MS SQL databázi. Při dobře naprogramované MVC aplikaci, stačí pouze vytvořit nový Model bez nutnosti měnit zbylé dvě části. Samozřejmě je třeba při přechodu dodržet datový model, případně naprogramovat vhodné konverzní algoritmy. Ale to trochu předbím. Pojdme si nejdříve vysvětlit, co jsou to ty jednotlivé logické části.

Model – reprezentuje data a obecně i business logiku samotné aplikace. Získává informace z databázových tabulek, nebo třeba i ze souborů na úložišti. Obsahuje však také například validační pravidla. Ty se velmi často dávají do View, což je ale z pohledu MVC není správné, neboť taková validace na počet znaku, je nutná nejen při registraci uživatele, ale i při jeho editaci což by ve View vedlo ke zbytečným duplikacím.

View – uchovává informace, které má zobrazit. Ve své podstatě zprostředkovává rozhraní mezi uživatelem a aplikací. Nejčastěji je interpretované pomocí HTML, ale není to nezbytné. Mezi zajímavé interpretace bych zařadil ještě formáty jako jsou XML nebo JSON.

Controller – je jakýmsi prostředníkem mezi View a Modelem. Stará se o tok událostí a obecně o aplikační logiku. Pracuje s daty získanými od uživatele a dle napsaných algoritmů s nimi nakládá.

Základní a v zásadě i nejdůležitější vlastností MVC architektury je fakt, že Model nesmí mít přímou vazbu na View nebo Controller. View už ze samotné podstaty zobrazování dat, má přímou vazbu na Model. Velmi často mívá také vazbu na Controller, kdy tato vazba bývá obousměrná. Samotný controller má přímou vazbu vždy na Model, aby mohl modifikovat data. Pro lepší pochopení jsem vytvořil obrázek s prezentací MVC architektury.

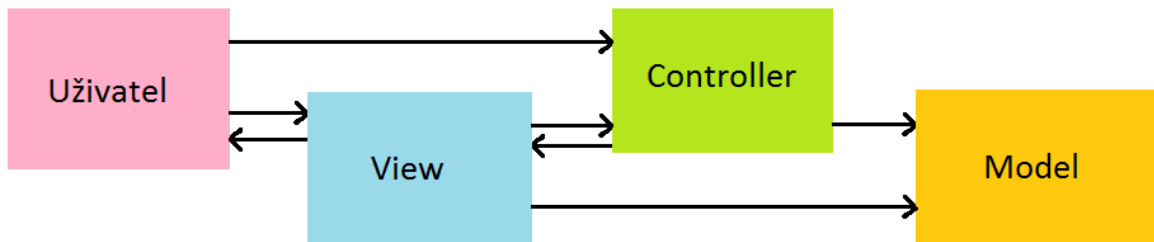


Obrázek 8 Vazby v MVC architektuře.

Pokud tuto architekturu navážeme na uživatele, dojdeme k dvěma možným interakcím.

- 1) Uživatel zadá data do formuláře a ta budou validována přímo ve View. Jedná se typicky o různé registrační formuláře, kde je třeba validovat emailovou adresu, poštovní směrovací číslo, nebo ošetřit počet vkládaných znaků.
- 2) Uživatel odešle formulář a ten se bude validovat až v Controlleru. Například takové přihlášení, pokud obsahuje vícero možností (lokální a např. skrze Facebook nebo Google účet), je třeba ověřit až v Controlleru. Uživatel však nemůže získat zpětnou vazbu přímo od Controlleru a tak je výsledek opět vrácen View, který ho zobrazí.

Následná ilustrace zobrazuje možnosti uživatele při komunikaci s MVC aplikací.



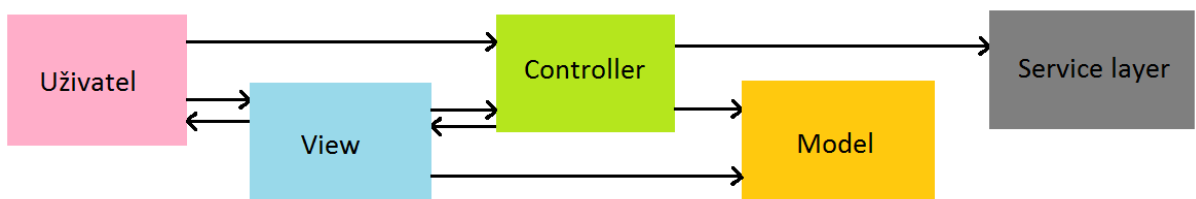
Obrázek 9 Interakce uživatele s MVC architekturou.

Uživateli se v žádném případě nesmí umožnit komunikovat s Modelem přímo. Pokud uživatel odesílá nějaký formulář, při němž dochází k validaci, tak ta je provedena skrze View, které se dotáže Modelu na správnost informací ještě před samotným odesláním. Pokud tato validace neprojde, žádná akce se směrem k Controlleru neprovede a místo toho jsou uživateli zobrazeny nevalidní položky.

Životní cyklus jedné iterace, pro jednoduchost např. přepnutí na jinou stránku:

- 1) Uživatel klikne na položku v menu na zobrazené stránce. Akce je odeslána na server.
- 2) Tato akce je zachycena Controllerem a je vykonána patřičná reakce. V našem případě se nastaví, které View se má zobrazit.
- 3) Uživateli se zobrazí View s požadovaným obsahem.

V současné době se MVC architektura začíná pozvolna rozšiřovat na tzv. MVC-Service layer, kde Service layer přebírá řízení přístupu k bázi dat, nebo nějakému file systému, kde jsou data uložena. Tyto vrstvy, jsou většinou reprezentovány pomocí interface a v zásadě jsou na systému prakticky nezávislé. Fungují na principu, kdy Controller nastavuje interface požadované služby, viz následující obrázek.



Obrázek 10 MVC-Service layer architektura.

Věřím, že nyní vám je již jasnější co to vlastně MVC je a v čem je aplikace včasného varovného systému napsána. Pokud se sami rozhodnete psát nějaké aplikace

v MVC architektuře, mějte prosím na paměti, že není nutné MVC striktně dodržovat v úplně každé stránce. Je mnoho případů, kdy se dá problém vyřešit pomocí lehkého AJAX na 3 řádky.

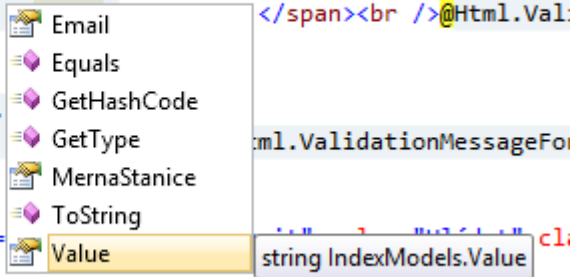
Aplikace pro systém včasného varování je napsána v MVC 3 architektuře, jejíž hlavní novinkou je Razor view engine. Mezi další novinky třetí verze MVC patří například Globální filtry, dynamic View Model anebo podpora Dependency Injection. Jelikož předmětem práce není podstata MVC, což by mimochodem samo stačilo na diplomovou práci, neboť v této oblasti je tolik nejasností a nepřesností, že je velmi obtížné vytvářet správné MVC aplikace. Podíváme se však blíže na zmíněnou novinku v podobě Razor engine, která byla při vývoji použita a snad se mi i podařilo z velké části využít její potenciál.

5.3 Razor View Engine

Jak už název napovídá, jedná se o zobrazovací nástroj, který používám pro psaní View v rámci MVC aplikace. Jeho hlavní výhody bych shrnul do několika bodů.

- Jedná se o engine, který umožňuje efektivní psaní funkcionalit za použití minima znaků, což vede k dobré přehlednosti samotného kódu.
- Je v celku jednoduché pochopit základní syntaxe a nic tak nebrání jeho efektivnímu využívání.
- Samotný kód je možno psát v dobře známých jazycích C# nebo VB. Přípony jednotlivých souborů jsou pak .cshtml nebo .vbhtml.
- Má velmi dobrou podporu ve vývojovém prostředí Visual Studio. Při nainstalování ASP.NET MVC 3 doplňku je k dispozici plně funkční Intellisense.

```
<table>
  <tr>
    <td>@Html.LabelFor(m => m.Value)</td>
    <td>@Html.TextBoxFor(m =>
  </tr>
</tr>
  <tr>
    <td>@Html.LabelFor(m => m.
    <td>@Html.TextBoxFor(m =>
  </tr>
</tr>
  <tr>
    <td colspan="2"><input id=
  </tr>
</tr>
</table>
```



Obrázek 11 Intellisense v podání Razor engine ve Visual Studiu 2010.

Pro psaní dynamického obsahu se využívá znak zavináče (@), místo původní otevírací a zavírací ostré závorky doplněné o znak procenta (<% %>). Rozdíl v zápisu syntaxe je dobře patrný na příkladu, jež uvádí sám Scott Guthrie [5], který Razor engine jako první oznámil na svém blogu.

Původní zápis s použitím defaultního .aspx souboru:

```
<ul id="produkty">
  <% foreach(var p in produkty) { %>
    <li><%=p.Nazev%> - <%=p.Cena%></li>
  <% } %>
</ul>
```

Stejný kus kódu pomocí Razor:

```
<ul id="produkty">
  @foreach(var p in produkty) {
    <li>@p.Nazev - @p.Cena</li>
  }
</ul>
```

Dle mého názoru je přehlednost a i samotná jednoduchost zápisu, jasně patrná již na takto malé části kódu. Jistě si dovedete představit, jak byly původní .aspx stránky nepřehledné, pokud bylo třeba použít složitější logiky.

Razor sám rozezná, zda je zavináč použit jako deklarace funkčního bloku anebo jestli se jedná o znak používaný v textu, který je třeba zobrazit v nezměněné podobě. Například u emailové adresy. Je to díky chytře napsanému parseru, který engine využívá.

Stejně jednoduché, jako psaní cyklů, je i psaní podmínek.

```
@if (1 == 1)
{
  <p>Je to pravda!</p>
}
else
{
  <p>Ne, je to lez!</p>
}
```

Funkční blok se pak samozřejmě dá použít i pro generování celého obsahu dynamických stránek. Zde je výpis dynamicky renderovaného obsahu použitého v aplikaci.

```
<body>
  <div class="page">
    <div id="header">
      <div id="title">
        <h1>System včasného varování</h1>
      </div>
      <div id="logindisplay">
        @Html.Partial("_LogOnPartial")
      </div>
      <div id="menucontainer">
        <ul id="menu">
          <li>@Html.ActionLink("Mapa", "Index",
"Home")</li>
          @if (User.Identity.IsAuthenticated)
          {
            <li>@Html.ActionLink("Nastavení", "Nastaveni",
"Home")</li>
          }

          <li>@Html.ActionLink("Jak na to?", "How",
"Home")</li>
          <li>@Html.ActionLink("Email & SMS", "Email",
"Home")</li>
          <li>@Html.ActionLink("O aplikaci", "About",
"Home")</li>
          @if (User.IsInRole("Admin"))
          {
            <li>@Html.ActionLink("Admin", "Admin",
"Home")</li>
          }
        </ul>
      </div>
    </div>
    <div id="main">
      @RenderBody()
    </div>
    <div id="footer">
    </div>
  </div>
</body>
```

Kdysi jsem slyšel velmi ambiciózní názor, že Razor je poslední hřebíček do rakve staříčkému PHP. Osobně se domnívám, že nějaký hromadný přechod z PHP k vyšším programovacím jazykům samotný Razor engine nerozpoutá, ale rozhodně je to dobrý začátek.

5.4 Datová struktura modelu

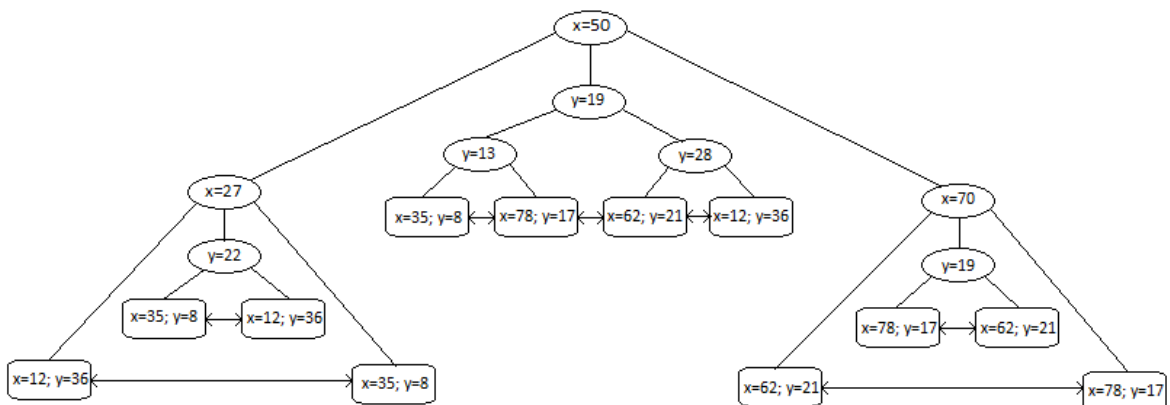
Pro uchovávání dat za běhu aplikace jsem vytvořil datovou strukturu 2D range tree [1], jež mi umožňuje uchovávat vrcholy a hrany v paměti serveru a není tak nutné je pokaždé načítat z databáze.

Dvojdímní rozsahový strom je datová struktura, jejímž zobecněním dostaneme kD strom. Je založen na principu základního stromu s několika podstromy. Každý strom vychází z principů binárního vyhledávacího stromu. Pro vybudování stromu, jsou použity mediány první dimenze stromu a pro každý medián je následně budován stejným principem i podstrom druhé dimenze.

Nejprve je tedy nutné zdrojová data seřadit a následně určit medián, který se stane kořenem stromu. Hodnoty nižší než kořen stromu, budují levý podstrom. Hodnoty vyšší pak budují pravý podstrom. Tento postup se rekurzivně opakuje pro oba podstromy a následně i pro druhou dimenzi. Cílem je vybudovat vyváženou strukturu, k čemuž je zapotřebí znát data předem.

V rámci implementace rozsahového stromu na problematiku sítě měrných stanic, jsem za vrcholy zvolil koordináty $[x, y]$, jež reprezentují umístění v mapě měrných stanic. Jako hrany jsem zvolil samotné toky, jež stanicemi protékají. Hrany nenesou žádné informace a mají tak v aplikaci pouze ilustrativní charakter.

Na následujícím obrázku, je dobře patrná ilustrace dvojdímní rozsahového stromu použitého v této aplikaci.



Obrázek 12 Jednoduchá ilustrace dvojdímní rozsahového stromu.

V oválných buňkách jsou znázorněny mediány listů stromu. Hodnoty z obrázku mají pouze ilustrativní charakter, tak je prosím berte s rezervou. V hranatých buňkách jsou znázorněny samotné listy stromů. Z obrázku je dobře patrná struktura, kdy každý x-ový medián má svůj y-nový podstrom.

Jelikož pro vyvážené vybudování 2D range tree, je třeba znát data předem, vytvořil jsem pomocnou strukturu pole-tabulka, jež uchovává informace o všech vrcholech. Abstrakci datové struktury zajišťuje použití generické kolekce. Při přihlédnutí k požadavkům na řídké vkládání a odebírání vrcholů, jsem zvolil generické kolekce `List<T>` a `Dictionary<TKey, TValue>`. Složitosti jejich operací jsou uvedeny v tabulkách 5 a 6.

Vrcholy:

- jako struktura pro ukládání vrcholů je použito pole
- reprezentuje se jako generická kolekce `List<T>`
`private List<Vrchol<TGraf, TTyp>> vrcholy;`

Hrany:

- v aplikaci se objevují zřídka a mají pouze ilustrační charakter
- pro ukládání hran, jsem využil Tabulku
- reprezentuje se jako generická kolekce `Dictionary<TKey, TValue>`
- kde `TValue` reprezentuje generická kolekce `List<T>`
`private Dictionary<Vrchol<TGraf, TTyp>, List<Hrana<TGraf, TTyp, TValue>>> hrany;`

Operation (`List<T>`):

Add	$O(1)^*$
Contains	$O(n)$
Count	$O(1)$
IndexOf	$O(n)$
Insert	$O(n)$
Remove	$O(n)$

Tabulka 5 Složitosti jednotlivých operací, pro práci s vrcholy.

Operation (`Dictionary<TKey, TValue>`):

Add	$O(1)^*$
ContainsKey	$O(1)$
ContainsValue	$O(n)$
Remove	$O(1)$

Tabulka 6 Složitosti jednotlivých operací, pro práci s vrcholy.

* Pokud kapacita musí být zvýšena, aby pojala nový element. Operace této metody se stává $O(n)$ kde n je počet.

Každý z vrcholů na sobě nese následující informace:

```
public Vrchol(TVrchol vrchol, TTyp typVrcholu,
             MernaStanice mernaStanice)
{
    this.Vrch = vrchol;
    this.Typ = typVrcholu;
    this.MernaStanice = mernaStanice;
}
```

Kde `TVrchol vrchol` je parametr generického typu a v konkrétní implementaci této aplikace představuje koordináty x, y reprezentované jako `new Point(x, y)`;

`TTyp typVrcholu` je také parametr generického typu a v konkrétní implementaci popisuje, zda se jedná o měrnou stanici nebo město, což umožňuje jiný přístup k informacím, které je třeba zobrazit.

Třída `MernaStanice mernaStanice` obsahuje konkrétní informace ke konkrétní stanici. Celý výčet si můžete prohlédnout v přílohách kompletního zdrojového kódu aplikace.

Ve chvíli, kdy je vybudována pomocná struktura, mohu vytvořit 2D range tree. Tento strom je tvořen zřetěžením prvků typu `Value2DRangeTree`. Prvek obsahuje pole dvou prvků typu `Value2DRangeTree`, které reprezentují syny daného prvku. Range tree je implementován ve třídě `Sít'`.

```

public void Build2DRangeTree()
{
    var vrcholy = new List<Vrchol<Point, TTyp>>();
    foreach (var item in this.graf.Vrcholy)
    {
        vrcholy.Add(item);
    }

    vrcholy.Sort(delegate(Vrchol<Point, TTyp> a,
        Vrchol<Point, TTyp> b) { return
        a.Vrch.X.CompareTo(b.Vrch.X); });
    rangeTree = BuildTree(vrcholy, 0, vrcholy.Count - 1);
}

```

Pro uchování datové struktury v paměti jsem vytvořil Singleton, který podrobněji rozepíši v následující kapitole.

5.5 Singleton

Jedná se o návrhový vzor, který umožňuje běh jedné instance třídy po celý životní cyklus aplikace [6]. Třída napsaná pomocí tohoto návrhového vzoru poskytuje globální přístup k ní samotné. Pro příklad z praxe můžeme využít jako ukázkou schránku z prostředí Windows, která existuje v jedné instanci a data se tak skrze ní, mohou předávat napříč různými aplikacemi.

Jedinečnost instance je zajištěna již v soukromém konstruktoru singletonu tak, aby nemohlo dojít k vytvoření další instance. Pozor si musíme dát při vytváření více vláknových aplikací, kdy se musí ošetřit, aby při vytváření singletonu jedním vláknem, nedošlo zároveň k tomu samému procesu v druhém vlákně. V aplikaci jsem využil thread-safe lazy singleton, díky čemuž se singleton vytváří až ve chvíli jeho statického volání.

Na následujících třech blocích kódu uvedu postupně příklady základních singletonů ve verzi non thread-safe, thread-safe a lazy thread-safe singleton využitý přímo v aplikaci.

```

public sealed class Singleton
{
    static Singleton instance = null;

    Singleton()
    {
    }
    public static Singleton Instance
    {
        get
        {
            if (instance == null)
            {
                instance = new Singleton();
            }
            return instance;
        }
    }
}

```

Základní a jednoduchý non thread-safe singleton. Z problematiky více vláken popsané výše, ho nedoporučuji ve více vláknových aplikacích využívat. Místo toho je programátorsky mnohem čistší použít implementaci tohoto návrhového vzoru, viz následující blok kódu.

```

public sealed class Singleton
{
    static Singleton instance = null;
    static readonly object padlock = new object();

    Singleton()
    {
    }
    public static Singleton Instance
    {
        get
        {
            lock (padlock)
            {
                if (instance == null)
                {
                    instance = new Singleton();
                }
                return instance;
            }
        }
    }
}

```

Tento návrhový vzor využívá soukromé proměnné padlock k užití funkce lock, díky čemuž znemožňuje dvojí inicializaci a singleton je tak plně připraven pro více vláknové aplikace.

Vytváříte-li aplikaci ve verzi .NET 4, pak můžete využít instanci třídy Lazy pro vytvoření vaší konkrétní implementace singletonu, viz mnou použitý kód:

```
private static readonly Lazy<Singleton> lazy =
    new Lazy<Singleton>(() => new Singleton());

public static Singleton Instance
{
    get { return lazy.Value; }
}

private Singleton()
{
}
```

Jednoduché a efektivní. Ať se rozhodnete pro jakoukoli implementaci, všechny dobře splní svůj účel a vy tak budete moci mít přístup k datům po celý běh aplikace, tak jako mam já přístup k jednou vybudovanému 2D range tree, obsahujícímu měrné stanice a informace s nimi spojené.

5.6 Sběr dat

Samotná data, získávám automatickým dolováním dat z předpovědního a měrného modelu ČHMÚ [2]. Data jsou získávána v html formátu a následně skrze parser ukládána do databáze.

Pro snadnější práci s dolováním dat z html formátu jsem využil veřejně dostupnou knihovnu HtmlAgilityPack. Práce s daty je díky předpřipraveným metodám velmi snadná a rychlá. Stručný náhled do problematiky osvětlí následující blok kódu.

```
HtmlDocument doc = new HtmlDocument();
doc.Load("file.htm");

foreach(HtmlNode link in doc.DocumentNode.
    SelectNodes("//div[@class='tbody']/table/tr"))
{
    HtmlAttribute att = data["href"];
    att.Value = Parse(att);
}
```

Job je napsaný jako služba a běží nepřetržitě bez ohledu na stav samotné aplikace. Aktuálně je nastavený interval 15ti minutového opakování, jež považuji za dostatečný. V případě potřeby, je možné job spustit manuálně z administračního menu aplikace.

5.7 Báze dat v MS SQL

Pro uchování trvalých informací jsem zvolil relační databázový systém MS SQL [3]. Jeho hlavní výhodou pro mě byla plná integrace v Microsoft prostředích a nástrojích. Mnoho práce mně ušetřila inteligentní integrace přímo ve Visual Studiu. Velmi kvalitně je také zpracovaný nástroj Management Studio, jehož prostředím se dá s daty efektivně pracovat. Jedná se o nástroj pro přímé vykonávání SQL příkazů. MS SQL plně podporuje programování pomocí T-SQL a například v rámci uložených procedur či funkcí se dají vytvářet i značně složité dotazy nebo celé logické bloky.

Pro autentizaci je použit ASP.NET Membership Provider, který umožňuje přihlašování a registrace pomocí integrovaných funkcí. Tyto funkce vyžadují ke svému bezproblémovému provozu nakonfigurovanou SQL databázi, do které se ukládají informace o uživateli. V uvedené knihovně je hromada předpřipravených funkcí, pro snadnou správu uživatelů. Namátkou vyberu například:

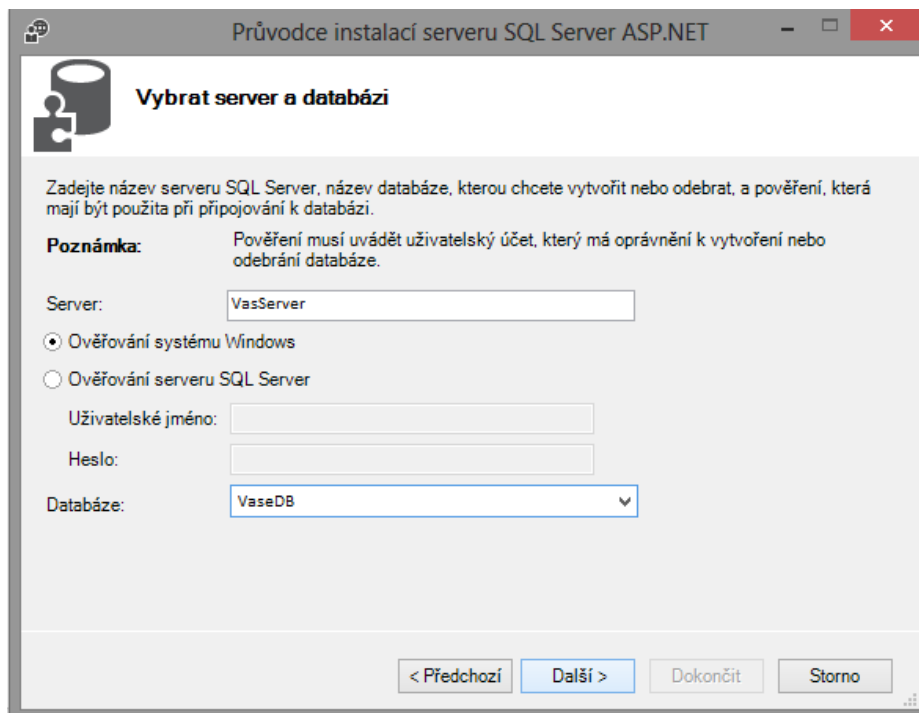
- Správa a údržba uživatelů a jejich hesel.
- Ukládání podpůrných informací o uživateli.
- Automatické registrace nových uživatelů za pomoci pár řádků kódu.
- Změny a automatické resety hesel.
- Vytváření unikátních identifikací.
- Kompletní role management.

Kompletní seznam naleznete na stránkách Microsoft Developer Network (MSDN).

K efektivnímu využívání je nejprve nutné připravit MS SQL Server a poté i samotnou aplikaci. Zmíněné potřebné úpravy, jsou nastíněny v následujících několika řádcích.

Na straně databáze je nejjednodušší provést následující postup.

- 1) Ve složce C:\Windows\Microsoft.NET\Framework\v4.0.30319 spustíte soubor aspnet_regsql.
- 2) Klikněte na Další.
- 3) Zvolte konfigurovat server SQL Server pro služby aplikací.
- 4) Zadejte název serveru, zvolte způsob ověřování a vyberte databázi, viz screen.



Obrázek 13 ASP.NET Membership Provider.

- 5) Překontrolujte si zadané údaje.
- 6) Dokončit.

Použití samotné knihovny je velmi snadné, stačí ve web.config nastavit connectionStrings a membership, viz příklad z aplikace.

```
<connectionStrings>
  <add name="ApplicationServices"
        connectionString="Data Source=VasZdroj; Initial
        Catalog=VaseDB; Integrated Security=True;"
        providerName="System.Data.SqlClient" />
</connectionStrings>
```

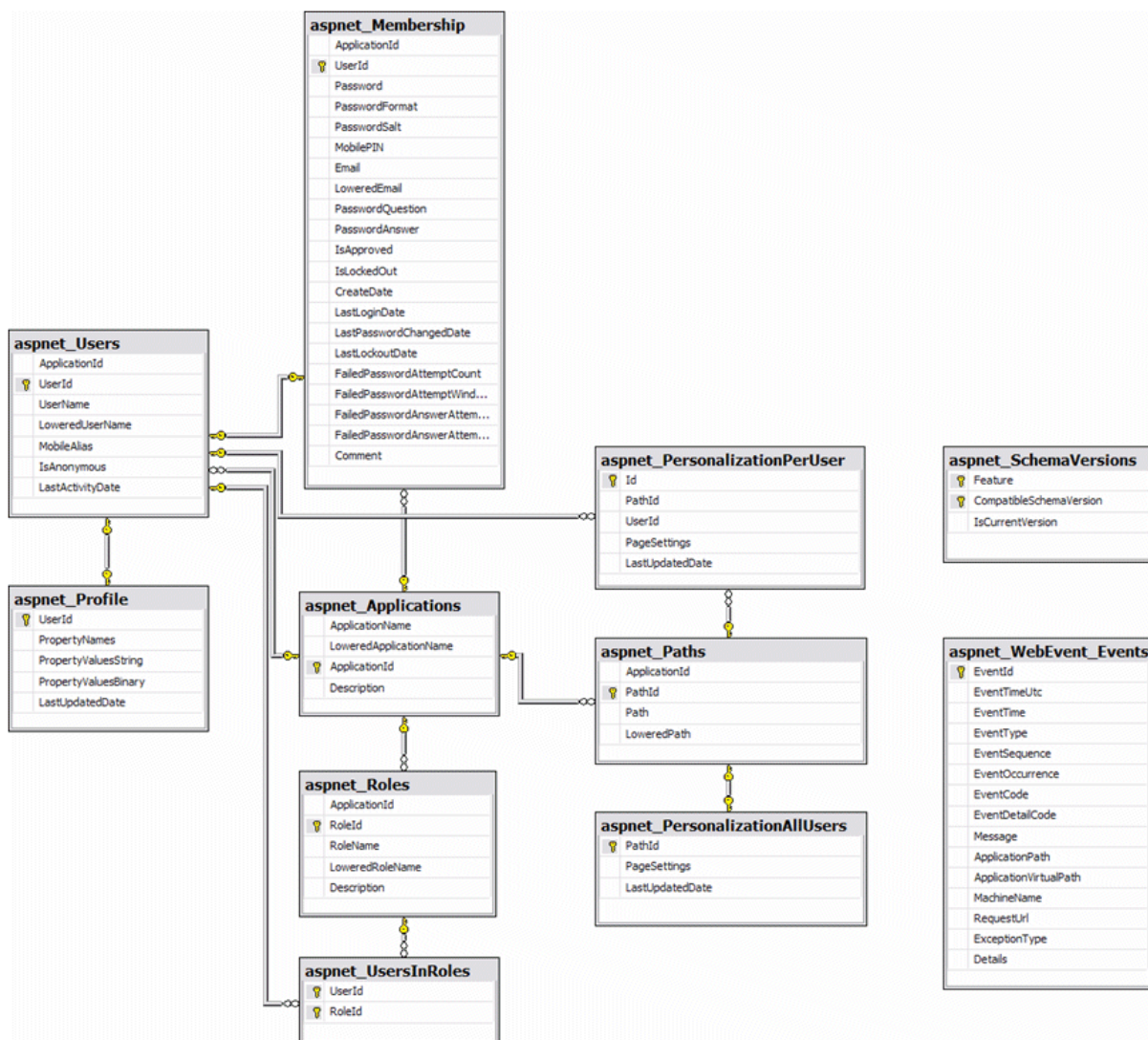
Pro své aplikace můžete využít stejný connectionStrings, jen stačí změnit „VasZdroj“ a „VaseDB“.

V membership části se dají mimo jiné definovat minimální délka hesla, nebo třeba počet pokusů, po kterých se má účet zablokovat (dle dalších parametru).

```
<membership>
  <providers>
    <clear />
    <add name="AspNetSqlMembershipProvider"
        type="System.Web.Security.SqlMembershipProvider"
        connectionStringName="ApplicationServices"
        enablePasswordRetrieval="false"
        enablePasswordReset="true"
        requiresQuestionAndAnswer="false"
        requiresUniqueEmail="false"
        maxInvalidPasswordAttempts="5"
        minRequiredPasswordLength="6"
        minRequiredNonalphanumericCharacters="0"
        passwordAttemptWindow="10"
        applicationName="/" />
  </providers>
</membership>
```

Je to opravdu velmi jednoduché a přesto značně efektivní. Pomocí této knihovny se dá řešit kompletně celá problematika ohledně přihlašování a registrací nových uživatelů. Další obrovskou výhodou je bezpečnost takového návrhu. Knihovna je chráněna proti mnoha způsobům potencionálních útoků, včetně populárního SQL Injection. Je to zajištěno přístupem k jednotlivým datům výhradně skrze procedury.

Relační model ASP.NET Providers je znázorněn na následující stránce. Jedná se o kompletní seznam tabulek pojmenovaných vždy s předponou „asponet_“. Základem je tabulka Applications, která odlišuje jednotlivé aplikace v rámci jedné databáze. Uživatelé jsou spravováni v tabulce Users na kterou je přímo navázána nejrozsáhlejší tabulka Membership, s doplňujícími informacemi o konkrétním uživateli. V rámci aplikace jsou ještě využívány tabulky Roles, pro definici rolí a spojovací tabulka UsersInRoles (která jak již sám název napovídá), říká, jaký uživatel má kterou roli. Pro více informací ohledně této zajímavé technologii, vás odkáži na stránky MSDN, uvedené pod zdrojem číslo 4 na konci této práce.

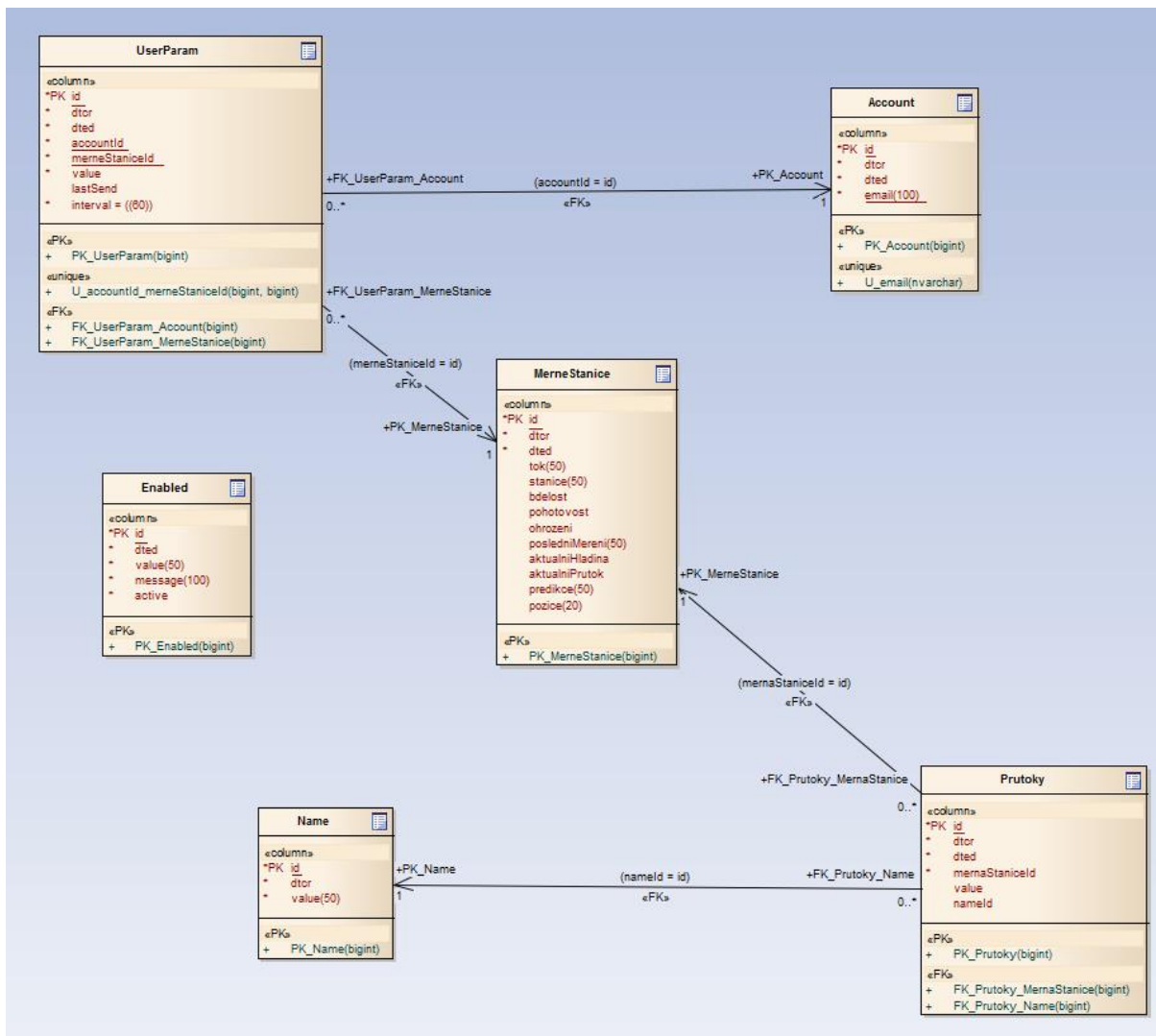


Obrázek 14 Relační model ASP.NET Providers (Zdroj: MSDN).

5.8 Relační model webové aplikace

Relační model databáze je v zásadě jednoduchý, neboť většina logiky je v samotné aplikaci, kdy se informace generují dynamicky a ukládají pouze v předpřipravené datové struktuře.

Základem je tabulka „MerneStanice“, kde jsou uloženy informace spojené s vybranými měrnými stanicemi, které systém aktuálně obsluhuje. Navazují na ní „Prutoky“, kde jsou uloženy doplňující informace a tabulka „UserParam“, v které jsou definovány hodnoty, při kterých má systém uživatele varovat. Speciální tabulkou je „Enabled“, která slouží k nastavení, zda se mají vybrané funkce potlačit, aby se data neměnila pod rukou (například při prezentaci, nebo přidávání nových stanic). Zablokovat se dají jednotlivé funkcionality nezávisle na sobě (přidávání nových stanic, běh jobu, zřizování skupin, apod.).



Obrázek 15 Relační model Systému včasného varování.

Oba zmíněné modely jsou vzájemně propojeny skrze tabulky Account – aspnet_Membership vazbou 1:1.

Kompletní datový model naleznete na příloženém datovém médiu.

6 Email a SMS zdarma

Naše nejrozšířenější trojice mobilních operátorů umožňuje zasílání SMS zpráv na váš telefon, po obdržení emailu a ke všemu ještě zcela zdarma. V této kapitole se dozvíte, jak si tuto službu zpřístupnit a nastavit. Názorné ilustrace jsou tvořeny v rámci Gmail emailu, s přeposíláním na mé Vodafone číslo. Ale u ostatních poskytovatelů emailových služeb i ostatních mobilních operátorů je situace velmi podobná. Důležité informace se pokusím sumarizovat i u zbyvajících poskytovatelů mobilních služeb.

6.1 Zřízení SMS emailu

Nejprve je nutné si zřídit SMS email u vašeho mobilního operátora. Nejjednodušší to mají zákazníci Telefonica O2, která tyto adresy zřizuje ke každému číslu automaticky. Ale vezmeme to pěkně pořadě.

T-Mobile – musíte se přihlásit ke správě svého účtu, prostřednictvím stránek „Můj T-Mobile“ a aktivovat si službu „E-mail do SMS“.

..T..Mobile

Poslat SMS Hledat

Osobní Podnikatelé Firmy

Volání Telefony Internet Služby Podpora eShop Můj T-Mobile

Přihlaste se do Můj T-Mobile

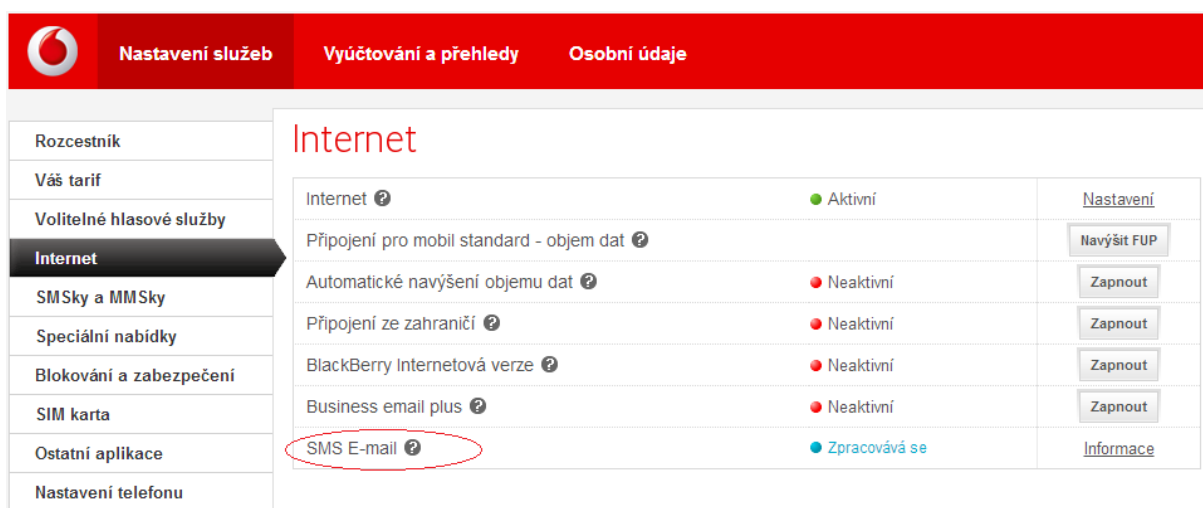
<p>Vstupte jednorázově</p> <p>Vaše mobilní telefonní číslo</p> <input type="text"/> <p>Poslat SMS s kódem</p>	<p>Přihlaste se</p> <p>Uživatelské jméno</p> <input type="text"/> <p>Heslo</p> <input type="password"/> <p>Přihlásit</p> <p><input type="checkbox"/> Zapamatovat přihlášení</p> <p>Zapomenuté heslo</p>	<p>Nevíte si rady?</p> <ul style="list-style-type: none">▪ Obnovte si zapomenuté heslo nebo odemkněte zamčený účet▪ Zaregistrujte se▪ Přihlaste se automaticky přes síť T-Mobile
--------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Obrázek 16 Přihlášení do „Můj T-Mobile“.

Při aktivaci je nutné nastavit adresu, ze které bude možné upozornění dostávat. Poté vám bude přiřazena vaše mobilní emailová adresa ve tvaru xxxxxxxxx@sms.t-mobile.cz, kde za „x“ doplňte vaše telefonní číslo.

Telefonica O2 – každému zákazníkovi je automaticky zřízen SMS email ve tvaru +420xxxxxxxx@sms.cz.o2.com, kde za „x“ doplňte vaše číslo. Výsledný email pak bude vypadat například takto: +420720123456@sms.cz.o2.com.

Vodafone – je třeba se přihlásit do Vodafone samoobsluhy, vstoupit do menu „Nastavení služeb“ sekce „Internet“ a zde si aktivovat službu „SMS E-mail“. Budete vyzváni, abyste si zvolili jméno, které bude doplněno před emailovou doménou. Výsledný email vypadá následovně: vaseadresa@vodafoneemail.cz.



Obrázek 17 Aktivace SMS E-mail ve Vodafone samoobsluze.

U všech uvedených operátorů je aktivace této služby zcela zdarma, stejně jako všechny příchozí sms na váš telefon. Pokud si přejete informace ze systému uchovávat ve vaší emailové schránce, bude nutné nastavit přeposílání doručených varování od systému. Jak to udělat, napoví následující kapitola. Jestli vám však stačí pouze SMS informace, pak prosím pokračujte v čtení kapitolou „6.3 Ověření“.

6.2 Nastavit přeposílání

Následující ukázka je vytvořena v rámci Gmail účtu, ale podobně se bude nastavovat filtrované přeposílání zpráv i u ostatních poskytovatelů emailových služeb. Ve své podstatě nejde o nic složitějšího, než je přeposílání emailu na jiný email. Jen je to třeba dělat automaticky a podle zvoleného filtru.

Přihlaste se do svého Gmail účtu a klikněte na ozubené kolo v pravém horním rohu. Zvolte možnost „Nastavení“ a následně vyberte podkategorii „Přeposílání a protokol POP/IMAP“. Zde klikněte na „Přidat adresu pro přeposílání“ a vyplňte adresu, kterou máte zřízenou z předešlé kapitoly. Po potvrzení a ověření se vám adresa přidá do seznamu „Přeposílání“.

Nyní je třeba vytvořit filtr. Přejděte do záložky „Filtry“ a klikněte na „Vytvořit nový filtr“. Do pole „Od“ vyplňte email `info@systemvcasnehovarovani.cz` a klikněte na „Vytvořit filtr z tohoto vyhledávání“.

« zpět na možnosti vyhledávání ×

Při obdržení zprávy, která odpovídá tomuto vyhledávání:

- Přeskočit Doručenou poštu (Archivovat)
- Označit jako přečtené
- Označit hvězdičkou
- Použít štítek:
- Přeposlat komu:
[přidat adresu k přeposílání](#)
- Smazat
- Nikdy neposílat do složky Spam
- Vždy označit jako důležité
- Nikdy neoznačovat jako důležité
- Zařadit do kategorie:

Použít filtr také na konverzace (0) odpovídající kritériu konverzací.

[Další informace](#)

Poznámka: stará pošta nebude přeposlána.

Obrázek 18 Nastavení akce pro nový filtr.

Výsledek by měl vypadat následovně. Všimněte si, že za vaším SMS emailem je v závorce uvedeno „používána ve filtru“.

Obecné Šítky Doručená pošta Účty Filtry **Přeposílání a protokol POP/IMAP** Chat Webov
Laboratoř Offline Motivy

Přeposílání:
Další informace

Deaktivovat
 Přeposlat kopii příchozí pošty na adresu

vaseAdresa@vodafoneemail.cz (používána ve filtru) a

Uchovat kopii v Doručené poště Gmailu

Tip: Můžete také přeposílat jen některé své zprávy, pokud si k tomu vytvoříte filtr.

Obrázek 19 Výsledek nastavení přeposílání.

6.3 Ověření

Nyní by vám měli automaticky chodit emaily, které vás budou informovat o případném blížícím se nebezpečí, podle vámi definovaných kritérií. Ověřit funkčnost si můžete jednoduchým způsobem a to když si na stránkách webové aplikace zadáte pro nějaký tok příliš nízkou hodnotu. Při následné aktualizaci dat (interval 15 minut), se vyhodnotí překročení vámi stanoveného limitu a systém odešle email na uvedenou emailovou adresu. Buď přímo na SMS email, anebo na jinou adresu, kterou jste uvedli a z které se to automaticky přepoše na SMS email. Z něho vám vzápětí přijde samotná SMS zpráva.

7 Závěr

Hlavním cílem diplomové práce bylo vytvořit aplikaci pro monitorování dešťových srážek podepisujících se na zvyšování hladin toků s ohledem na umístění měrných stanic. Bylo zapotřebí vyhodnotit tyto údaje a vytvořit metody pro včasné varování o blížícím se nebezpečí v podobě zvýšených hladin měřených toků.

Cílem teoretické práce bylo seznámení s problematikou povodní a analyzování měrných stanic. Objasnit terminologii s tím spojenou a poskytnout užitečné rady pro omezení budoucích škod. V rámci psaní práce došlo na přelomu května a června 2013 k rozsáhlým povodním zejména na území Čech a tak byla vytvořena kapitola, jež podrobně analyzovala příčiny a důsledky těchto povodní. Důležitým faktorem analýzy bylo vybrat vzorek měrných stanic, na kterých by se systém mohl postavit a odladit. Bylo zapotřebí navrhnout algoritmy pro včasné varování s ohledem, aby nedocházelo k zbytečnému šíření paniky. Byl proto vytvořen vzorec, založený na síle průtoku desetileté vody, který byl v rámci testování aplikace doplněn o koeficient změny. Jedná se o možnost upravit vypočítanou hodnotu, dle dlouhodobých zkušeností obyvatel, jež v dané lokalitě žijí. Účelem je, poskytnout co nejpřesnější odhad hranice, která již představuje vysoké riziko rozvodnění vybraného toku.

Praktická část se zaměřila na moderní technologie dnešní doby a jejich implementace v rámci vytvořené aplikace. Byly popsány implementace použitých datových struktur, systémových funkcionalit a návrhových vzorů. Vytvořeno webové rozhraní, pro široký přístup veřejnosti. Detailně byly rozepsány použité technologie s jejich konkrétními ukázkami implementací. Nastíněn budoucí směr, kterým se má aplikace ubírat. Naprogramovány metody pro vyhodnocování rizika a následné varování prostřednictvím SMS emailu s rozepsáním jak tuto službu využít zdarma.

Aplikace je nasazena v produkčním režimu a probíhají takzvané praktické testy. Před nasazením aplikace proběhly podrobné testy v lokálním prostředí a odladěno chování tak, aby se minimalizovaly problémy spojené se širokým využitím. Tím považuji zadání diplomové práce za splněné a nad rámec jsem přidal i pár funkcí navíc. Samotný kód je na přílohu značně rozsáhlý a proto si jej můžete prohlédnout na přiloženém datovém médiu, společně s ostatní dokumentací.

Na závěr bych chtěl shrnout samotný vývoj aplikace, k jejímuž zdárnému vyhotovení mi velmi pomohla kvalitně zpracovaná analýza. V rámci teoretické části se vyhodnotily

všechny důležité faktory, které je třeba v aplikaci zohlednit a jejich samotná implementace již byla pouze otázkou umění programování. V tomto ohledu jsem mohl zužít své tříleté zkušenosti s vývojem webových aplikací, v rámci práce na pozici programátora pro významného lídra na poli informačních technologií. Tuto pozici jsem vykonával na plný pracovní úvazek, prakticky po celou dobu magisterského studia. Datový model aplikace byl vytvořen s ohledem na budoucí možnosti rozšiřitelnosti aplikace a dimenzován tak, aby bylo dostatek prostředků i pro několikaletý nepřetržitý provoz. Z vlastních prostředků jsem zakoupil hosting, včetně CZ domény na jeden rok. Náklady na to nejsou zcela zanedbatelné, ale pokud se aplikace osvědčí a bude aktivně pomáhat minimalizovat škody, tak tuto cenu rád zaplatím každý další rok.

Seznam použité literatury

1. Kavička, Antonín: *Datové struktury a algoritmy: 2D range tree*, Učební text
Fakulty elektrotechniky a informatiky Univerzity Pardubice, Univerzita Pardubice,
Pardubice 2013
2. ČHMÚ : *Hlásná a předpovědní povodňová služba* [online]. 2013-08-13
[cit. 2013-08-13]. Měrné stanice. Dostupné z WWW:
<<http://hydro.chmi.cz/hpps/index.php?lng=CZE>>.
3. WALTERS, E. R. a kolektiv. *Mistrovství v SQL Server 2008*.
Computer Press Books, 2009, ISBN 978-80-251-2329-4.
4. MSDN : *Microsoft ASP.NET 2.0 Providers: Introduction* [online]. 2013-08-13
[cit. 2013-08-13]. Membership Providers. Dostupné z WWW:
<<http://msdn.microsoft.com/en-us/library/aa478948.aspx>>
5. *ScottGu's Blog : ASP.NET MVC 3* [online]. 2013-08-13 [cit. 2013-08-13].
Razor View Engine. Dostupné z WWW: <<http://weblogs.asp.net/scottgu/>>
6. Christian Nagel, Bill Evjen, Jay Glynn, Karli Watson, Morgan Skinner
C# 2008 - Programujeme profesionálně. Computer Press Books, 2009,
ISBN 978-80-251-2401-7.
7. Bill Evjen, Scott Hanselman, Devon Rader *ASP.NET 3.5 v jazycích
C# a Visual Basic - Programujeme profesionálně*. Computer Press Books, 2009,
ISBN 978-80-251-2069-9.
8. ČAP : *České asociace pojišťoven* [online]. 2013-08-13 [cit. 2013-08-13]. Mapy.
Dostupné z WWW:
<<http://www.cap.cz/Item.aspx?item=Povod%C5%88ov%C3%A9+mapy>>.