

# **Vývoj webového rozhraní systému Floreon+**

## **Development of Web-based Interface for the Floreon+ System**

## Zadání bakalářské práce

Student:

**Jakub Smékal**

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Vývoj webového rozhraní systému Floreon+  
Development of Web-based Interface for the Floreon+ System

Zásady pro vypracování:

Systém Floreon+ je modulárním systémem pro předpověď povodní a podporu krizového řízení. V současné době probíhá vývoj nového webového rozhraní tohoto systému s mapovou komponentou založeného na client-side technologiích a WMS/WFS webových službách.

Cílem této práce je rozšířit připravované webové rozhraní o moduly autentizace a autorizace, uživatelského nastavení a personalizace a vkládání informací o krizových událostech do systému včetně jejich vizualizace.

Konkrétní body zadání bakalářské práce jsou:

1. Seznámit se se současným stavem připravovaného webového rozhraní systému Floreon+ a jeho technologiemi.
2. Navrhnout a implementovat jednotnou autentizaci a autorizaci přes všechny propojené komponenty systému.
3. Navrhnout strukturu a možnosti uživatelského nastavení a personalizace systému a implementovat navrženou funkčnost do systému.
4. Vyvinout moduly pro vkládání informací o krizových událostech a zajisit jejich vizualizaci v mapové komponentě webového rozhraní.

Seznam doporučené odborné literatury:

[1] JQuery:

<http://jquery-navod.cz/serial/serial-o-jquery>

<http://www.w3schools.com/jquery/>

[2] SVG:

<http://www.w3schools.com/svg/>

<http://www.inkscape.org/en/learn/tutorials/>

[3] OpenLayers:

<http://workshops.boundlessgeo.com/openlayers-intro/>

<http://trac.osgeo.org/openlayers/wiki/Documentation>

[4] WMS a WFS:

<http://www.opengeospatial.org/standards/wms>

<http://www.opengeospatial.org/standards/wfs>

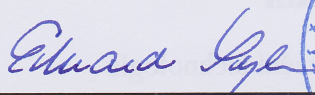


Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

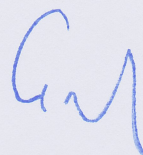
Vedoucí bakalářské práce: **Ing. Štěpán Kuchař**

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015



doc. Dr. Ing. Eduard Sojka  
vedoucí katedry

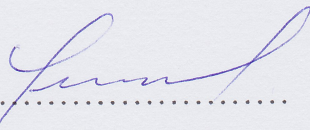


prof. RNDr. Václav Snášel, CSc.  
děkan fakulty



Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

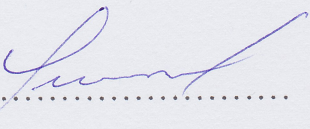
V Ostravě 7. května 2015



.....

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2015



.....



Na tomto místě bych rád poděkoval vývojářskému týmu Floreon+, jmenovitě Ing.Štěpánu Kuchařovi, Ing.Marku Bobrovi, Ing.Davidu Vojtkovi, PhD. a Janu Křenkovi, za možnost realizace bakalářské práce formou podílení se na projektu takové velikosti.



## **Abstrakt**

Systém Floreon+ je modulárním systémem pro předpověď povodní a podporu krizových řízení. V současné době probíhá vývoj nového webového rozhraní tohoto systému s mapovou komponentou založenou na client-side technologiích a WMS/WFS webových službách. Cílem této práce je rozšířit webové rozhraní o moduly autentizace a autorizace, uživatelského nastavení a personalizace a vkládání informací o krizových událostech do systému včetně jejich vizualizace.

**Klíčová slova:** Floreon, personalizace, uživatelské rozhraní, autentizace, autorizace

## **Abstract**

The Floreon+ system is a modular system for flood prediction and disaster management support. Currently, a new web interface is being developed which includes a map component based on client-side technologies and WMS/WFS web services. The aim of this thesis is to extend the web interface by adding modules allowing authentication and authorisation, user settings and personalisation and inserting information about critical events into the system including their visualisation.

**Keywords:** Floreon, personalization, user interface, authentication, authorization



## Seznam použitých zkratk a symbolů

HTML	– Hyper Text Markup Language
CSS	– Cascading Style Sheet
JS	– JavaScript
AJAX	– Asynchronous JavaScript and XML
SVG	– Scalable Vector Graphics
WMS	– Web Map Service
WFS	– Web Feature Service
WCF	– Windows Communication Foundation
HTTP(s)	– HyperText Transfer Protocol (Secure)
WWW	– World Wide Web
XML	– Extensible Markup Language
WCF	– Windows Communication Foundation
LDAP	– Lightweight Directory Access Protocol
UUID	– Universal unique Identifier
OGC	– Open Geospatial Consortium



## Obsah

<b>1</b>	<b>Úvod</b>	<b>4</b>
<b>2</b>	<b>Architektura systému</b>	<b>5</b>
<b>3</b>	<b>Technologie</b>	<b>6</b>
3.1	HTML 5.0 . . . . .	6
3.2	CSS 3.0 . . . . .	6
3.3	JavaScript . . . . .	7
3.4	SVG . . . . .	8
3.5	OpenLayers . . . . .	8
3.6	WMS . . . . .	8
3.7	WFS . . . . .	9
3.8	WCF . . . . .	9
<b>4</b>	<b>Stav systému</b>	<b>10</b>
4.1	Uspořádání . . . . .	10
4.2	Autentizace v systému . . . . .	11
4.3	Možnost personalizace . . . . .	11
4.4	Přijaté body úprav . . . . .	11
<b>5</b>	<b>Autentizace a autorizace</b>	<b>12</b>
5.1	Registrace . . . . .	12
5.2	Přihlášení . . . . .	12
5.3	Server side - WCF . . . . .	13
5.4	Autentizace na Geoserveru . . . . .	16
<b>6</b>	<b>Uživatelské rozhraní a personalizace</b>	<b>17</b>
6.1	Časová osa . . . . .	17
6.2	Souřadnicový systém . . . . .	17
6.3	Mapové výřezy . . . . .	19
6.4	Komponenta ovládaní vrstev . . . . .	20
6.5	Lokalizace jazyka . . . . .	22
<b>7</b>	<b>What-if analýza</b>	<b>24</b>
7.1	Co je What-If analýza . . . . .	24
7.2	Polygonální výběr . . . . .	24
7.3	Vizualizace simulace . . . . .	26
<b>8</b>	<b>Závěr</b>	<b>28</b>
<b>9</b>	<b>Reference</b>	<b>29</b>
<b>10</b>	<b>Přílohy</b>	<b>30</b>

---

## Seznam obrázků

1	Rozvržení systému Floreon+ . . . . .	5
2	Původní vzhled systému . . . . .	10
3	Nová komponenta registrace do systému . . . . .	13
4	Diagram dědičnosti tříd sloužících k autorizaci . . . . .	14
5	Diagram části databáze spojené s autentizací a autorizací . . . . .	14
6	Úprava časové osy . . . . .	17
7	Komponenta k přepočtu souřadnicových systémů . . . . .	19
8	Ukázka komponenty mapových výřezů . . . . .	20
9	Nová komponenta mapových vrstev (vlevo) a stará (vpravo) . . . . .	22
10	Náhled komponenty what-if analýzy . . . . .	25
11	Náhled komponenty polygonálního výběru . . . . .	25
12	Vygenerovaný hydrogram . . . . .	26
13	Vizualizace rozlivu v mapové komponentě . . . . .	27

---

## Seznam výpisů zdrojového kódu

1	Ukázka HTML kódu . . . . .	6
2	Ukázka CSS . . . . .	7
3	Ukázka JavaScriptu . . . . .	7
4	Ukázka SVG v jazyku XML . . . . .	8
5	Ukázka využití vystavené WCF service služby . . . . .	9
6	Přepočítání do souřadnicového systému S-JTSK . . . . .	18



## 1 Úvod

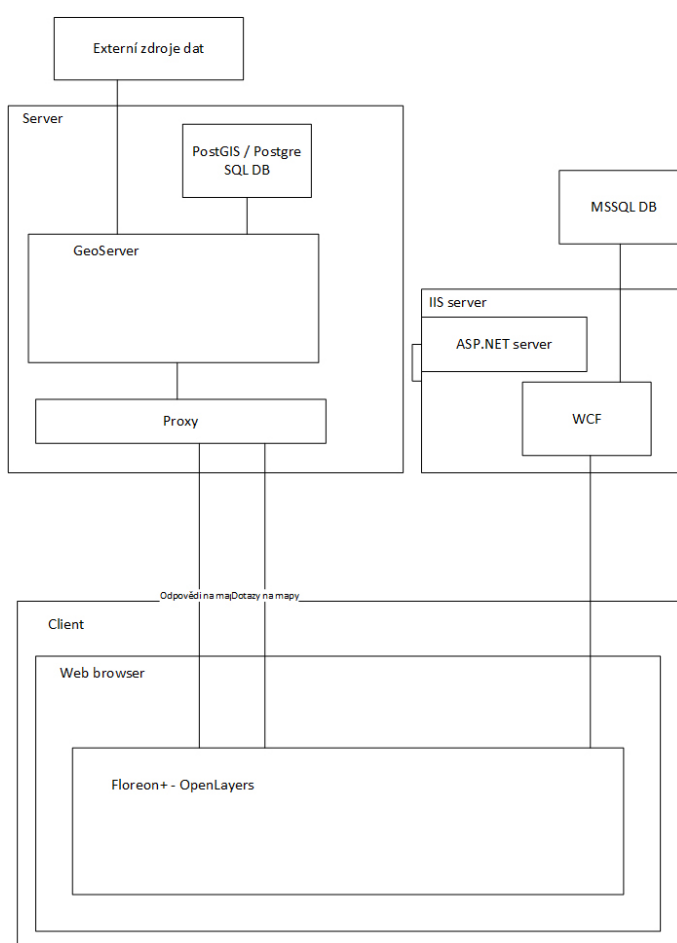
"Přírodě se nedá poručit jinak než tím, že se jí podřídíme", prohlásil již v 16. století Francis Bacon, anglický spisovatel, filosof, přírodovědec a státník. Ve svých vědeckých pracích prosazoval, aby příroda nebyla pokusy předvídána, ale vykládána. Trval na zákonech příčiny a důsledku, zdůrazňoval význam pokusu a praxe.

Extrémní výkyvy počasí existují odjakživa a na všech kontinentech, ačkoliv četnost a rozsah výkyvů je podnebím značně ovlivněno. V zalidněných oblastech však mohou tyto přírodní jevy způsobit při nejmenším finanční škody a v těch horších případech ztráty na lidských životech. Bylo by bláhové předpokládat, že se tyto extrémy objeví jen zřídka, nebo vůbec. Proto je dnes, v době moderních technologií, v zájmu vědy snažit se tyto extrémní výkyvy počasí včas předvídat, lokalizovat a umožnit tak včas jednat aby se následky takovéto události co nejvíce minimalizovaly, nebo aby se vzniku škody úplně předešlo.

Systém Floreon+ je zaměřen na problematiku predikce povodní a dalších krizových situací. Aktuálně probíhá úprava webového rozhraní systému a přidání několika nových funkcionalit. Tento dokument by měl být sumarizací a specifikací úkonů provedených při inovaci systému Floreon+. Mou základní náplní práce bylo aktivní podílení se na úpravách systému Floreon+, tzn. při každotýdenní poradě konzultovat o provedených změnách, připomínkovat k provedeným změnám, konzultovat a debatovat nad dalšími kroky a implementovat zadané úkoly z jednotlivých porad. Práce je rozdělena na čtyři základní části, jimiž jsou „Seznámení se se systémem Floreon+“ (vizte kapitoly 2, 3, 4), zabývající se analýzou architektury, použitými technologiemi a v poslední řadě aktuální formou systému. Druhou částí je „Autorizace a autentizace (vizte kapitolu 5), popisující principy přihlášení uživatele do systému a funkcionality spjaté s autentizací uživatele v systému. Třetí část je nejrozsáhlejší a je věnována uživatelskému rozhraní (vizte kapitolu 6) a popisu rozdílů mezi původní variantou řešení a nově implementovanou variantou. Poslední část je věnována vkládání informací o krizových situacích uživatelem (vizte kapitolu 7). V tuto chvíli toto řešení zahrnuje pouze what-if analýzu, ale je otevřené budoucím úpravám a rozšířením (vizte kapitolu 8). Spolu s touto prací vzniká paralelně práce od mého kolegy Jana Křenka, který se spolu se mnou podílel na vývoji uživatelského rozhraní, implementaci modulů a testování systému. Křenková práce pojednává hlavně o zátěžovém testování systému, výsledcích těchto testů a optimalizaci systému k zefektivnění chodu.

## 2 Architektura systému

Systém Floreon+ využívá jak client-side technologie, tak i server-side technologie. Server-side zajišťuje z největší části geoserver a PostGIS/PostgreSQL databáze. Veškeré funkcionality jsou postaveny na client-side technologiích (vizte níže kapitolu 3). Klientská strana přijímá datové struktury od serverové části, nad těmito daty se vykonávají operace k znázornění dat uživateli a naopak od uživatele se berou zadaná vstupní data, formují do správného formátu a odesílají zpět na serverovou část, kde se data ukládají, případně se nad nimi provádějí výpočty, které posléze opět slouží klientské straně ke správné vizualizaci. Architektura systému je naznačena na obrázku 1.



Obrázek 1: Rozvržení systému Floreon+

## 3 Technologie

Táto kapitola se věnuje základním technologiím, na kterých stojí webové rozhraní systému Floreon+.

### 3.1 HTML 5.0

HyperText Markup Language (zkr. HTML) je v informačních technologiích název značkovacího jazyka používaného pro tvorbu webových stránek, které jsou propojeny hypertextovými odkazy. HTML je hlavním z jazyků pro vytváření stránek v systému World Wide Web, který umožňuje publikaci dokumentů na internetu. Jazyk HTML by měl sloužit primárně k vytvoření struktury webových stránek (vizte výpis 1), tzn. k vytvoření příslušných elementů se správnou identifikací. Vzhled elementů zajišťuje jazyk CSS (vizte podkapitulu 3.2) a funkcionalitu JavaScript (vizte podkapitulu 3.3).

Jazyk HTML5 [2] (aktuální a nejnovější verze) umožňuje oproti předchozím verzím širokou škálu nových možností. V sémantickém zápisu přibyly elementy <header>, <footer>, <article> a <section>. Od verze 5 lze přesně definovat vstupní hodnoty ve formulářích - typy „number“, „date“, „time“, „calendar“ a „range“. V systému Floreon je ovšem nejvíce využito nově přidaných elementů <svg> a <canvas>, kde <svg> element umožňuje psaní SVG kódu (vizte podkapitulu 3.4) přímo v těle HTML dokumentu a element <canvas> vložení „kreslicího plátna“ do těla dokumentu.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Title of the document</title>
</head>

<body>
<div id=""mainDiv">
  <<<<div_id=""header">
    <<<<Content_of_header_<...
  <<<</div>
</div>
</body>

</html>
```

Výpis 1: Ukázka HTML kódu

### 3.2 CSS 3.0

Cascading Style Sheet (zkr. CSS) je jazyk definující vzhled, formu a případně primitivní funkčnost HTML (příp. XHTML nebo XML) elementů. Hlavním smyslem je tedy oddělit vzhled od obsahu a struktury dokumentu. Základní pravidlo psaní CSS dokumentu je jednoduché - skládá se z selektoru, složených závorek a slovníkovým výpisem vlastností, které chceme u zvoleného elementu (vizte výpis 2).



---

```
p {
  color: red;
  text-align: center;
  font-family: Arial;
  font-style: normal;
  text-transform: uppercase;
  text-align: center;
}
```

---

Výpis 2: Ukázka CSS

### 3.3 JavaScript

JavaScript (zkr. JS) je multiplatformní, objektově orientovaný skriptovací jazyk, dnes sloužící převážně jako interpretovaný programovací jazyk pro WWW stránky. Syntaxe JS patří mezi jazyky jako jsou C/C++ nebo Java (vizte výpis 3). JS je spouštěn před, v průběhu a nebo až po kompletním načtení HTML dokumentu a jedná se o client-side technologii, tudíž funkcionality jazyka se projeví až přímo u uživatele v jeho prohlížeči, nikoliv na straně serveru jako je tomu například u jiných interpretovaných jazyků jako PHP nebo ASP. JavaScript je dynamicky typovaným jazykem.

JavaScript může být psán rovnou v HTML dokumentu mezi tagy `<script>` a `</script>`, ovšem častější a vhodnější využití je načítání JS z externího souboru pomocí rozšíření tagu o „src“ atribut.

---

```
<script>
var fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits;

function myFunction() {
  fruits.sort();
  document.getElementById("demo").innerHTML = fruits;
}
</script>
```

---

Výpis 3: Ukázka JavaScriptu

#### 3.3.1 jQuery

jQuery [3] [4] je JavaScriptová knihovna s širokou podporou prohlížečů a klade důraz na interakci mezi JavaScriptem a HTML. Tato knihovna je vydávána pod volně šiřitelnou licenci MIT. jQuery zkracuje JS zápis kódu, ulehčuje jeho psaní, poskytuje snadnou a rychlou pomoc při hlídání událostí nebo například vytváření animací. Taktéž umožňuje značné ulehčení využití Ajaxu.

#### 3.3.2 Ajax

Asynchronous JavaScript and XML (zkr. Ajax) [13] je pojem označující technologii pro vývoj interaktivních webových stránek, které mění svůj obsah bez nutnosti kompletního

znovunačtení celé webové stránky. Tuto možnost má právě díky tomu, že probíhá asynchroně - nezávisle.

### 3.3.3 JSON

JavaScript Object Notation (zkr. JSON) [12] je datový formát nezávislý na platformě, sloužící k přenosu dat. Vstupem je jakákoli datová struktura (řetězec, celé číslo, reálné číslo, boolean hodnota, objekt nebo pole) a výstupem je vždy řetězec. Data jsou uchovávána ve slovníkovém systému - ke klíči přísluší jeho hodnota.

## 3.4 SVG

Scalable Vector Graphics (zkr. SVG) [5] je značkový jazyk a zároveň formát souboru popisující dvourozměrnou vektorovou grafiku pomocí XML. Jedná se o jeden z mála vektorových formátů na poli WWW stránek, neboť dodnes dominují rastrové formáty jako JPG, BMP nebo GIF. Předpokládá se, že formát SVG by se mohl stát základním formátem pro vektorovou grafiku na internetu. Mezi nejzákladnější výhody tohoto formátu patří velikost výsledného souboru, nezávislost na platformě, přenositelnost a například i možnost „vyhledávání“ v něm jako v klasickém XML souboru. Textová interpretace obrázku pak je tedy soubor základních geometrických tvarů popsaných základními parametry, text a nebo rastrová, vložená, složka (vizte výpis 4).

```
<rect x="80" y="60" width="250" height="250" rx="20"
  style=" fill :#ff0000;_stroke:#000000;stroke-width:2px;" />

<rect x="140" y="120" width="250" height="250" rx="40"
  style=" fill :#0000ff;_stroke:#000000;_stroke-width:2px;
  _ fill -opacity:0.7;" />
```

Výpis 4: Ukázka SVG v jazyku XML

## 3.5 OpenLayers

OpenLayers [7] [8] je otevřená, volně šiřitelná (BSD licence) JavaScriptová knihovna pro zobrazování mapových dat v internetovém prohlížeči. Tato knihovna poskytuje API pro vytvoření bohaté webové aplikace na zobrazování geografických dat, podobně jako například Google Maps. Knihovna podporuje GeoRSS, KML, GML, GeoJSON a jakýkoliv zdroj využívající OGC standardů jako služby WMS nebo WFS.

## 3.6 WMS

Web Map Service (zkr. WMS) [9] je standard vyvinutý a dodnes vyvíjený organizací Open Geospatial Consortium (zkr. OGC). Služba funguje na principu client-side a umožňuje sdílení geografických informací skrze rastrovou grafiku. Výsledkem požadavku na WMS službu jsou tedy obrázková data v nejrůznějších formátech (JPEG, TIFF, PNG, ..), které

reprezentují danou geografickou informaci, nebo například výsledek překrytu více mapových vrstev.

### 3.7 WFS

Web Feature Service (zkr. WFS) [10] je taktéž standardem dodnes vyvíjeným organizací OGC. Služba taktéž jako WMS umožňuje sdílení obrázkových geografických informací v prostředí internetu, s tím rozdílem, že u WFS služby se jedná o vektorová data, nikoli rastrová. Standardním výsledkem požadavku na WFS službu jsou geodata ve formátu GML.

### 3.8 WCF

Windows Communication Foundation (zkr. WCF) [14] je sada knihoven tvořící framework v rámci frameworku .NET, která zajišťuje komunikaci a umožňuje vytváření servisně orientovaných aplikací. Mezi základní vlastnosti patří podpora Ajax (použití vizte v výpis 5).

Právě přes Ajax a s využitím formátu JSON je propojená JavaScriptová část systému Floreon+ s vystavenými službami.

---

```
var WCF_URL = "http://develop.floreon.vsb.cz/Services/FloreonClientService.svc/";
$.ajax({
    type: "POST",
    url: WCF_URL + "GetUserSettingsList",
    data: JSON.stringify({
        token: $.cookie("Floreon.User.Token"),
        names: settings
    }),
    error: function (e) {
        alert (e.responseText);
    },
    success: callbackGetUserSettings,
    datatype: "json",
    contentType: "application/json"
});
```

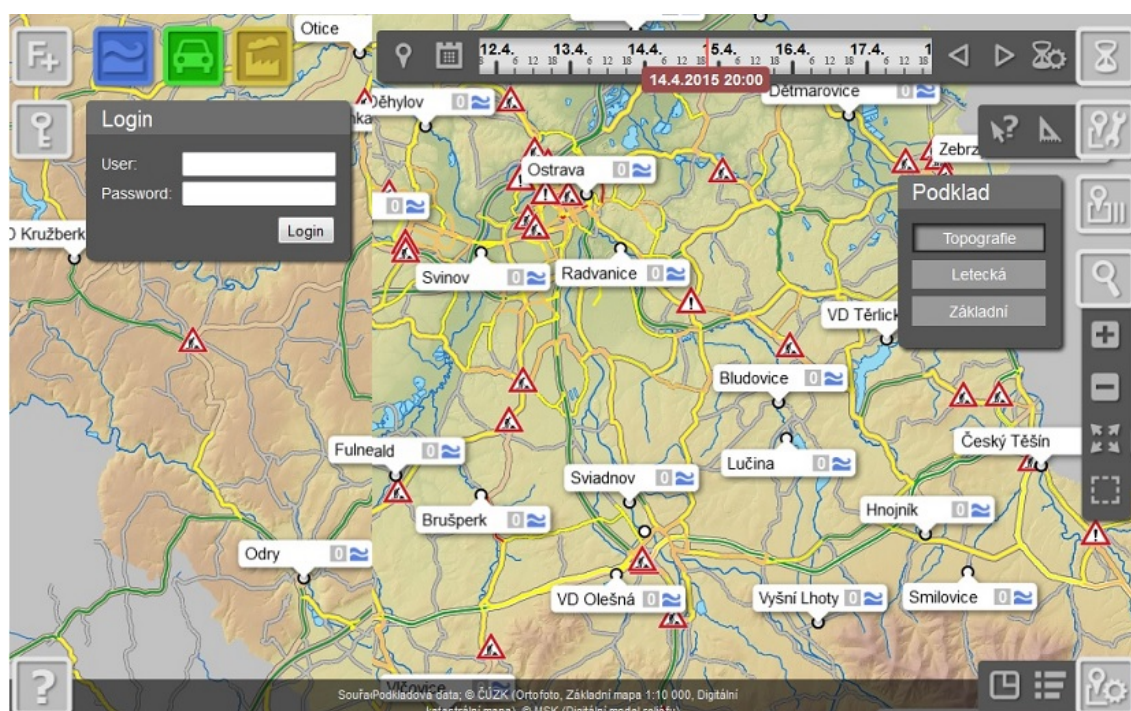
---

Výpis 5: Ukázka využití vystavené WCF service služby



## 4 Stav systému

V této kapitole bych rád rekapituloval stav systému Floreon+ před započítím úprav uživatelského rozhraní a základních funkcí webového rozhraní. Pro upřesnění představy vizte obrázek číslo 2.



Obrázek 2: Původní vzhled systému

### 4.1 Uspořádání

Uspořádání ovládacích prvků je na první pohled velmi nelogické. Ovládací prvky k zobrazování a skrývání jednotlivých mapových vrstev nejsou umístěny na jednotném místě. Komponenta sloužící k zapínání a vypínání vrstev povodní, dopravy a znečištění je v levé horní části, zatím co komponenta sloužící k přepínání typu podkladové vrstvy je v pravém bočním panelu.

Komponenta mapových nástrojů je umístěna mimo kontext ostatních nástrojů. Komponenta skrývající tlačítka pro legendu a náhledovou mapku je matoucí, lehce zaměnitelná s nastavením.

Existence tlačítka s písmenem „F+“ v levém horním rohu, bez jakékoli funkcionality je zbytečná. I za předpokladu, že by funkcionalitu mělo, těžko si domýšlet co by

takové tlačítko symbolizovalo, a zda by nebylo pro uživatele systému spíše klamné než přínosné.

Při výběru konkrétní JSDI kamery v Ostravě se velmi neprakticky objeví dolní panel s náhledem dané kamery. Prostor náhledu kamery zabírá malou část panelu, ale panel samotný zakrývá větší polovinu obrazovky a tak způsobuje prakticky nepoužitelnost mapy.

## 4.2 Autentizace v systému

Komponenta pro přihlášení nemá reálnou funkčnost a chybí funkcionality spojené s možností přihlašování se k vlastnímu účtu, jako např. registrace nového uživatele nebo možnost obnovení hesla. Je tedy zřejmé, že doposud může systém využívat pouze anonymní uživatel, bez jakékoli autentizace v systému. Není tedy jak rozdělit uživatelům role, a dále dle těchto rolí jejich pravomoce a omezení v systému.

## 4.3 Možnost personalizace

Systém neumožňuje žádnou formu personalizace. Jelikož uživatel se nemá jak vůči systému autentizovat, nelze ani takové možnosti v této fázi systému očekávat. Uživatel tedy nemá možnost žádných permanentních nastavení při své návštěvě systému.

## 4.4 Přijaté body úprav

Obecně přijaté body týkající se připravovaných úprav na systému, jsou následující.

- Informace o zdroji mapových podkladů ve stavovém řádku
- Vytvoření grafického měřítka
- Zobrazení GPS souřadnic - konverze souřadnic do více souřadnicových systémů
- Zobrazení územního členění (města, obce, kraje, ...)
- Zobrazení popisů řek na mapě a u hydrogramů
- Identifikace všech načtených informací - „progress bar“
- Ukládání historických událostí
- Vrstvy událostí se budou rozšiřovat - FLOREON, RODOS, ovzduší
- Legenda – zobrazení informací k prvkům pro právě zobrazené vrstvy

## 5 Autentizace a autorizace

Pojmem autentizace se rozumí proces ověření identity subjektu. Na základě autentizace navazuje proces autorizace, který na základě právě ověřené identity umožňuje danému subjektu vykonávat vybrané operace dle přidělených práv. Je-li subjekt v pořádku autentizován získá si vlastnost tzv. „autenticitu“.

### 5.1 Registrace

K tomu, aby mohl být uživatel v systému autentizován, bylo zapotřebí vytvořit komponentu umožňující registraci uživatele a jeho následné přihlášení k vlastnímu účtu v systému.

Komponenta pro přihlášení a možnosti spojené s přihlášením (registrace, zapomenuté heslo) byla přesunuta do vysouvacího pravého panelu (vizte obrázek 3), kde v horní části má své permanentní místo. V případě nového anonymního návštěvníka je tedy v horní části formulář pro zadání uživatelského jména (v systému Floreon+ je jako uživatelské jméno využit e-mail zadaný při registraci, z důvodů omezení mnohonásobných registrací jedním uživatelem) a hesla. Dále zde přibýly tlačítka „Registrace“ a „Zapomenuté heslo“.

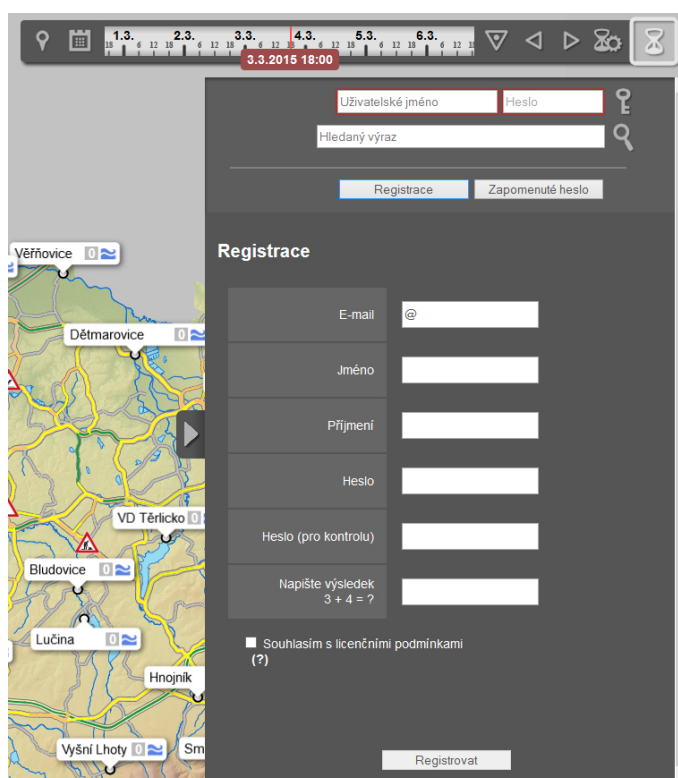
Při události registrace nového uživatele, se v celém obsahu pravého panelu objeví registrační formulář. Krom zadání relevantních dat k registraci, musí uživatel také prokázat, že není „robot“ (označení programu vykonávajícího rutinní činnost - např. sběr dat, odesílání požadavků, apod.) vyplněním pole captcha [11]. Captcha je zkratkou pro „completely automated public Turing test to tell computers and humans apart“ (kompletně automatický veřejný Turingův test k rozlišení počítačů a lidí), což doslovně vyjadřuje svou reálnou funkčnost.

Po úspěšném vyplnění a odeslání formuláře uživatelem se inicializuje funkce *registerNewUser()*, která přes Ajax pošle uživatelem zadaná data ve formátu JSON na WCF službu *RegisterToSystem*, která má za úkol na straně serveru uložit nově vytvořeného uživatele do databáze.

### 5.2 Přihlášení

Po úspěšné registraci je uživateli umožněno se pomocí svých přihlašovacích údajů přihlásit do systému. Po zadání správné kombinace uživatelského jména a hesla je inicializovaná funkce *loginUser()*, která se dotazuje přes službu WCF *LoginToSystem*. Je-li přihlášení úspěšné, volá se funkce *callbackLogin()*. Tato funkce krom faktu, že volá další funkce spojené s přihlášením uživatele má taky za úkol uložit do cookies (malé množství pomocných dat uložených v PC klienta) tzv. token (vizte podkapitulu 5.4). Token je vygenerovaný specifický řetězec znaků, který je jedinečný a jeho hlavní vlastností je autenticita. Tento token je po přihlášení tedy uložen do cookies v počítači uživatele, spolu s časovou stopou, která určuje životnost tohoto tokenu. V systému Floreonu je životnost tokenu nastavena na 1 hodinu. Tento časový údaj znamená, že pokud uživatel opustí

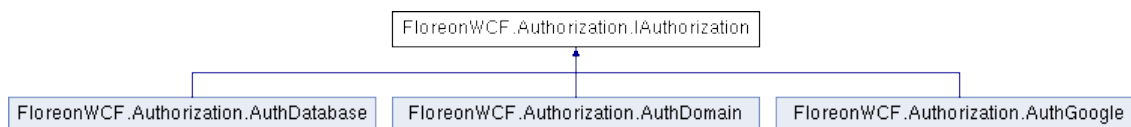
web Floreonu, bude ještě hodinu po odchodu přihlášen v systému. Při návratu na web do 1 hodiny po odchodu, bude tedy stále přihlášen a budou načtena jeho osobní nastavení. Dále funkce *callbackLogin()* volá funkce *setLoggedUser()* - sloužící k zobrazení/skrytí elementů souvisejících s přihlášením uživatele (např. již není vidět přihlašovací formulář, nýbrž přihlašovací jméno a ikona pro odhlášení), *getUserSettings()* - funkce sloužící k volání služby k získání osobních nastavení uživatele (vizte kapitolu 6), *getListWIA()* - slouží k získání seznamu uživatelem zadaných what-if analýz (vizte kapitolu 7)



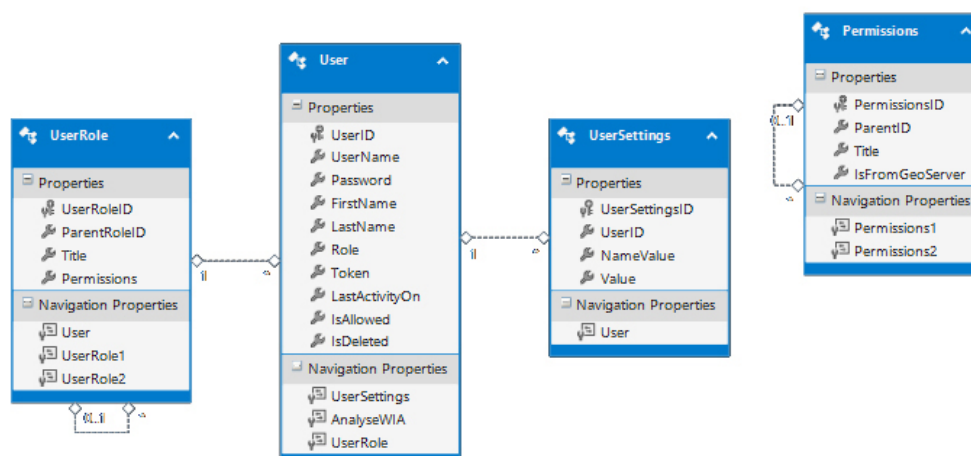
Obrázek 3: Nová komponenta registrace do systému

### 5.3 Server side - WCF

Na straně serveru autorizaci zajišťuje balík tříd „Authorization“. Tento balík obsahuje tři základní třídy *AuthDatabase*, *AuthDomain*, *AuthGoogle* a rozhraní *IAuthorization*. Dědičnost těchto tříd je znázorněna na obrázku 4. Další nedílnou součástí je MSSQL databáze, která slouží k ukládání všech relevantních dat souvisejících s autentizací a autorizací uživatele. Manipulaci s touto databází zajišťuje podpůrná třída *Database* (vizte 5.3.2) Struktura relační MSSQL databáze je zobrazena na obrázku 5.



Obrázek 4: Diagram dědičnosti tříd sloužících k autorizaci



Obrázek 5: Diagram části databáze spojené s autentizací a autorizací

### 5.3.1 IAuthorization

Toto rozhraní definuje dvě základní funkce. Funkce *ChangePassword(userName, oldPassword, newPassword)* slouží ke změně hesla uživatele a funkce *LogIntoSystem(userName, password)* plní funkci přihlášení daného uživatele do systému. Toto rozhraní implementují všechny tři třídy z balíku „Authorization“ - *AuthDomain*, *AuthGoogle* a *AuthDatabase*, kde každá z těchto tříd implementuje stejnou funkčnost zděděných metod způsobem, odpovídající danému způsobu autentizace.

*AuthGoogle* je třída připravená k možnosti rozšíření přihlášení do systému Floreon+ prostřednictvím účtu u společnosti Google. Tato třída aktuálně není v systému využita, je předdefinována pro budoucí účely dalšího vývoje systému. Třída implementuje rozhraní *IAuthorization*, a sama vlastní vlastnosti *Domain* sloužící k uložení údajů o doméně a *ConsumerSecret* sloužící k uchování ověřovacího klíče.

*AuthDomain* je třída připravená k rozšíření možnosti přihlášení do systému skrze LDAP (celouniverzitní jednotné heslo). Stejně jako třída „AuthGoogle“ obsahuje tato

třída metody z rozhraní *IAuthorization* a navíc vlastnosti *ADsPath* sloužící k uchování cesty k LDAP a *Domain* sloužící k uložení údajů o doméně.

*AuthDatabase* slouží k připojení do relační MSSQL databáze. Funkce zděděné z rozhraní využívají hlavně funkci z třídy *Database*. Implementované metody této třídy slouží k přihlášení a změně hesla.

### 5.3.2 Database

Tato třída obsahuje veškeré metody vykonávané nad databází. Metody implementované v této třídě jsou využívány ve všech třídách dědicích z rozhraní *IAuthorization* a služby *FloreonClientService*. Dále vypsané metody se podílejí na autentizaci.

- *getUser(userName)* - sloužící k nalezení uživatele na základě uživatelského jména. Funkce vrací objekt uživatele.
- *getUserByToken(token)* - funkce k nalezení uživatele pomocí posledního uživatelského tokenu. Funkce vrací objekt uživatele.
- *createUser(data)* - vytvoření nového uživatele na základě dat získaných z registračního formuláře. Takto vytvořenému uživateli je automaticky přidělena základní role.
- *UpdateUserToken(user, token)* - funkce sloužící k aktualizaci záznamu uživatele. Ukládá do objektu uživatele token předaný v argumentu funkce. Jedná se o proceduru bez návratové hodnoty.
- *UpdateUserPassword(user, password)* - aktualizace uživatelského hesla. Nalezení objektu uživatele podle dat z argumentu funkce a přiřazení nového hesla.
- *LogoutUser(token)* - metoda k odhlášení uživatele ze systému.
- *UpdateUserActivity(token)* - metoda sloužící k prodloužení živnosti tokenu při odezvě uživatele. Konkrétně při každém kliknutí v rozhraní systému *Floreon+* je tato funkce použita a expirační doba tokenu prodloužena na základní hodnotu.
- *UpdateUserActivation(userId)* - metoda volána k aktivaci uživatele. Jakmile se registruje nový uživatel, musí svůj účet aktivovat kliknutím na vygenerovaný odkaz v doručeném e-mailu. Po kliknutí na tento odkaz se volá tato funkce služby. Funkce vyhledá uživatele podle ID a nastaví účet na aktivovaný stav.
- *IsValidUserToken(userName, token)* , *IsValidUserToken(token, hasCompareExpiration)* - funkce k ověření zda je uživatelský token validní. Funkce vyhledá uživatele, načte jeho poslední známý token a ověří, zda je časová známka stále aktuální. Funkce navrácí boolean hodnotu.



### 5.3.3 FloreonClientService

FloreonClientService je vystavená služba obsahující metody, které lze využívat z externích zdrojů. Tato podkapitola se věnuje popisu jen nejdůležitějších funkcí této služby, sloužících k autentizaci do systému Floreon+. Metody jsou volány ze strany klienta, resp. z JavaScriptu pomocí Ajaxu. Jednou z nejpoužívanějších metod je metoda *LogIntoSystem(data)*. Tato metoda si zavolá funkci ze třídy Database k získání objektu uživatele pomocí uživatelského jména. Následuje ověření, zda byl uživatel nalezen a dále také kontrola o jakou autentizaci se jedná - zda autentizaci k databázi, doméně a nebo Googlu. V poslední fázi této metody, než dojde k navrácení objektu obsahujícího informace o přihlášení daného uživatele, dochází ke kontrole, zda je uživatelský token validní. Nevypršela-li platnost tokenu, pak se uloží do návratového objektu token daného uživatele. Je-li token po expiraci, dojde k vygenerování nového tokenu. Před ukončením metody se volá metoda třídy Database *UpdateUserToken(user, token)*.

Další důležitou metodou této služby je *RegisterToSystem(data)*, která zajišťuje registraci nového uživatele. Zde se v prvních krocích zjišťuje, zda již neexistuje uživatel zadaného uživatelského jména. Je-li uživatelské jméno v pořádku, dochází k volání metody ze třídy Database k vytvoření nového uživatele. Dále dochází k vygenerování aktivačního URL a odeslání informačního e-mailu uživateli spolu s vygenerovaným aktivačním URL odkazem. Tento odkaz slouží k úplnému dokončení registrace uživatele.

Služba dále poskytuje metody k získání uživatelských nastavení, k jejich uložení, dále k získání/uložení seznamu what-if analýz, k získání informací o dostupnosti mapových vrstev, apod.

## 5.4 Autentizace na Geoserveru

Autentizaci na Geoserveru zajišťuje modul „Key authentication“, který poskytuje jednoduchý autentizační protokol navrhnutý pro OGC klienty, kteří nepodporují žádný bezpečnostní protokol, ani základní ověřování HTTP. Pro tyto klienty modul umožňuje minimální formu autentizace připojením jedinečného klíče do URL adresy, který je používán jako pevný autentizační token. Tento způsob přístupu je bezpodmínečně spojován s použitím šifrovaného protokolu HTTPS. Vygenerovaný authkey v URL adrese reprezentuje jednoho konkrétního uživatele, takže je nutné jej přidat do každého autorizovaného požadavku. Skrze tuto metodu může uživatel volat funkce na GeoServeru jako *GetMap()*, *GetFeatureInfo()*, atd. Floreon+ využívá „Web Service provider“, který se připojuje přímo na popsanou WCF a získá si z ní název role uživatele. Každá role na WCF pak odpovídá právě jednomu uživateli na Geoserveru a definuje jeho práva.

## 6 Uživatelské rozhraní a personalizace

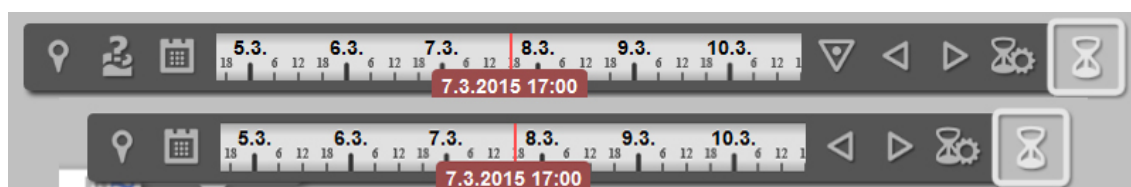
Tato kapitola se věnuje hlavně úpravě uživatelského rozhraní systému a personalizaci systému. Pojmem personalizace systému se rozumí proces, kdy uživatel s příslušnou identifikací a autorizací si může ke svému účtu nastavit výchozí stav systému. Nemusí se nutně jednat pouze o vzhled uživatelského rozhraní, ale například výchozí souřadnicový systém, časové rozmezí kroku na časové ose, apod.

### 6.1 Časová osa

V původní verzi systému byla možnost posunu po časové ose pouze tlačítky vlevo a vpravo, kde posun byl dán pevně o 1 hodinu. Tlačítko pro nastavení časové osy bylo nefunkční, tudíž velikost kroku nebylo možné měnit.

Časová osa byla obohacena o širší funkcionalitu. Přibylo tlačítko „Ted“, kde uživatel nemusí aktuální čas zdlouhavě klikat šipkami, ale tímto tlačítkem jednoduše načte aktuální čas. Dále byla implementována možnost kliknutí na časovou osu, jak by uživatel zcela intuitivně očekával. Při kliknutí na datum a přibližný čas v časové ose se tedy vizuální data v systému aktualizují (jsou-li k dispozici) na zvolený termín. Tlačítko „Nastavení časové osy“ nabylo svou funkcionalitu - slouží k nastavení velikosti kroku při posunu po časové ose šipkami. Uživatel si zde může zvolit jednotku (nejmenší jednotkou jsou minuty, největší roky) a počet jednotek při jednom kroku. Krok nelze nastavit v jednotkách sekund neboť taková možnost by byla irelevantní z důvodu většího časového intervalu u získávání dat z externích zdrojů.

Dále přibyly komponenty se seznamem uživatelem vytvořených what-if analýz (vizte kapitolu 7). Přihlášený uživatel si v tomto seznamu může vytvářet, editovat a mazat své what-if analýzy. Porovnání komponenty oproti původní verzi je zobrazeno na obrázku 6



Obrázek 6: Úprava časové osy

### 6.2 Souřadnicový systém

Původní verze zobrazovala souřadnice pouze ve formátu WGS84 / Pseudo-Mercator. V souvislosti s tímto faktem odpadla v původní verzi možnost volby výchozího formátu souřadnic pro registrované uživatele. Do systému Floreon byla tedy třeba dodělat komponenta a funkcionalita pro převod současného souřadnicového systému do dalších, nej-

používanějších formátů souřadnic. Na tuto skutečnost navazuje nutnost vytvoření možnosti volby pro uživatele v jakém formátu chtějí souřadnice získávat. Zvolené byly následující formáty.

- WGS84 / Pseudo-Mercator (původní souřadnicový systém)
- WGS84 / Geografický DMS (GPS stupně, minuty, sekundy)
- WGS84 / Geografický DD (GPS decimální)
- S-JTSK / Křovák Východ Sever
- WGS84 / UTM pás 33 Sever

K transformaci souřadnic z jednoho systému do systému jiných byla vytvořena funkce *transformCoord(type)*, která jako vstupní argument *type* bere celé číslo, kterým je identifikován typ požadované souřadnice. Tento identifikátor nezádává uživatel ručně, nýbrž jej zvolí ve formuláři, který je součástí nově vzniklé komponenty pro výběr souřadnic. Uživatel tedy vybere požadovaný výchozí typ souřadnic, kde s touto volbou je volána zmíněná funkce se správným parametrem. K samotnému přepočtu souřadnic mezi systémy poslouží funkce *Projection()* z knihovny OpenLayers. Ukázka využití funkcí knihoven OpenLayers je zobrazena ve výpisu 6.

```
var point = map.getLonLatFromPixel(map.events.getMousePosition(event));
var position_x = OpenLayers.Util.toFloat(point.lon,7);
var position_y = OpenLayers.Util.toFloat(point.lat,7);

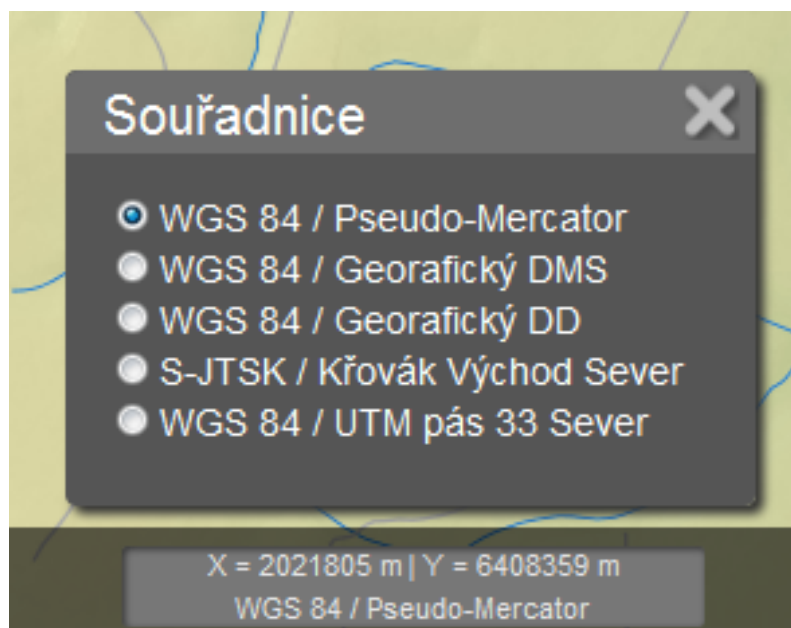
var firstProjection = '+proj=merc_a=6378137_b=6378137_lat_ts=0.0_lon_0=0.0_x_0=0.0_y_0=0.0_k=1.0_units=m_nadgrids=@null_wktext_+no_defs';

var secondProjection = "+proj=krovak_lat_0=49.5_lon_0=24.83333333333333_alpha=30.28813972222222_k=0.9999_x_0=0_y_0=0_ellps=bessel_towgs84=589,76,480,0,0,0_units=m_no_defs";
var i = proj4( firstProjection ,secondProjection,[position_x, position_y]);
return i;
```

Výpis 6: Přepočet do souřadnicového systému S-JTSK

### 6.2.1 Komponenta souřadnic

Přihlášený uživatel si může v nastavení zvolit výchozí formát souřadnic. Tato volba zůstane uložena v jeho účtu a kdykoliv se uživatel přihlásí, bude systém fungovat v zadaném formátu souřadnic. Pro rychlou změnu volby byla přidána komponenta do dolního informačního panelu. V tomto informačním panelu se souřadnice vypisují, a komponenta pro manipulaci s formátem souřadnic byla přidána na událost kliknutí na výpis souřadnic. Při události kliknutí se objeví komponenta pro výběr typu souřadnic, kdy při volbě souřadnicového systému uživatelem se tato volba projeví i v uživatelském nastavení. Náhled nově vzniklé komponenty je zobrazeno na obrázku 7.



Obrázek 7: Komponenta k přepočtu souřadnicových systémů

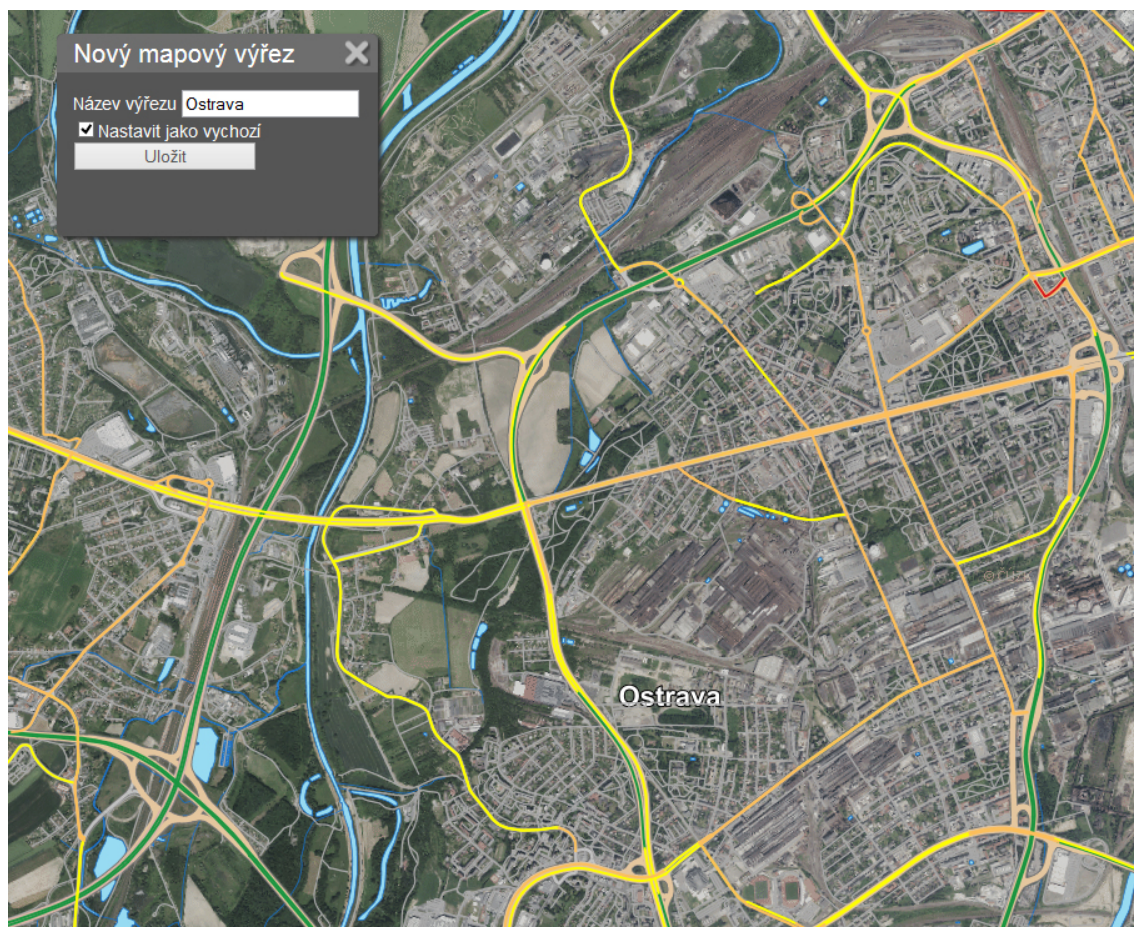
### 6.3 Mapové výřezy

Komponenta vytvoření až pěti mapových výřezů byla přidána, aby uživatel měl možnost vytvořit si záložky lokací, spolu s informacemi o spuštěných vrstvách. Tato funkčnost umožňuje uživateli vytvořit, editovat a smazat pět nejlépe viditelných lokací na mapě. Kromě faktu, že uživatel se takto pět uložených výřezů zobrazí jako tlačítka ve stavovém řádku, si také uživatel může jeden tento výřez zvolit jako výchozí. Takto zvolený výřez posléze bude domovským zobrazením při přihlášení uživatele.

#### 6.3.1 Implementace mapových výřezů

Mapový výřez se skládá ze tří základních informací - zeměpisná šířka a délka, zoom a seznam aktivních vrstev mapy. Na základě těchto tří informací lze uložit mapový výřez. Chce-li uživatel vytvořit nový výřez, musí si požadovanou lokaci nalézt na mapě, přiblížit si danou lokaci na dostatečnou velikost a zvolit v komponentě vrstev, které vrstvy mají být aktivní. Při kliknutí na tlačítko „Přidat mapový výřez“ se objeví komponenta k pojmenování vytvářeného mapového výřezu a volba, zda chce právě vytvořený mapový výřez nastavit jako výchozí. Posléze je volána funkce `saveCutoffMap(nameCut, defaultValue)`, která v parametrech předává informace z formuláře - tedy pojmenování nově vytvořeného výřezu a volbu, zda se má jednat o výchozí výřez. Tato funkce uloží do pole

výřezů na příslušný index informace o názvu výřezu, zeměpisnou šířku, délku a úroveň přiblížení. Tyto informace jsou uchovávány v datovém typu string, kde jsou jednotlivé informace odděleny čárkou. Takto vytvořená informace se uchová v poli mapových výřezů, které je součástí uživatelských nastavení, a je tedy metodou *saveUserSettings()*, využívající službu WCF *SetUserSettingsList*, uloženo v databázi k příslušnému uživateli. Náhled vzniklé komponenty je zobrazen na obrázku 8.



Obrázek 8: Ukázka komponenty mapových výřezů

#### 6.4 Komponenta ovládaní vrstev

Zásadní restrukturalizací prošel modul pro vypínání a zapínání jednotlivých mapových vrstev. Jednotlivé mapové vrstvy slouží k zobrazení/skrytí specifických dat na mapě. Systém Floreon má aktuálně tyto mapové vrstvy.

- Floreon

- Simulované záplavy
- Říční síť
- Měřicí stanice

Vrstva „Simulované záplavy“ vizualizuje do mapového podkladu geografická data se záznamy povodňových aktivit. Uživatel si může zobrazit i historické údaje a sledovat rozliv říční sítě při povodních v minulosti. Vrstva „Říční síť“ zobrazí vektorovou prezentaci říční sítě a vrstva „Měřicí stanice“ zobrazí na mapě všechny stanice Moravskoslezského kraje umístěné na jednotlivých vodních proudech, a jejichž úkol je měřit stav hladiny a hlídat okolní relevantní faktory.

- Rodos

- Silniční síť
- Rychlost dopravního toku
- Kamery (OVA.NET)
- Dopravní informace

Systém Floreon+ je ve spolupráci s Centrem pro rozvoj dopravních systémů RODOS [15] schopen zobrazovat informace o dopravní situaci v České republice.. Vrstva „Silniční síť“ zobrazuje na mapě vektorovou prezentaci všech oficiálních pozemních komunikací (automobilové komunikace) a vrstva „rychlost dopravního toku“ na tyto komunikace zobrazuje v reálném čase zatížení těchto komunikací a jejich průjezdnost. Vrstva „Kamery (OVA.NET)“ působí pouze na oblasti města Ostravy, kde umožňuje zobrazení kamer na každé větší křižovatce v tomto městě. Taktéž vrstva „Dopravní informace“ zobrazuje na komunikacích aktuální výluky, objížďky, aj.

- Znečištění

- Případová studie

Tato část systému není plnohodnotně zprovozněná. Jedná se o vrstvu s daty k vizualizaci simulace úniku škodlivé látky do ovzduší.

- Podkladová mapa

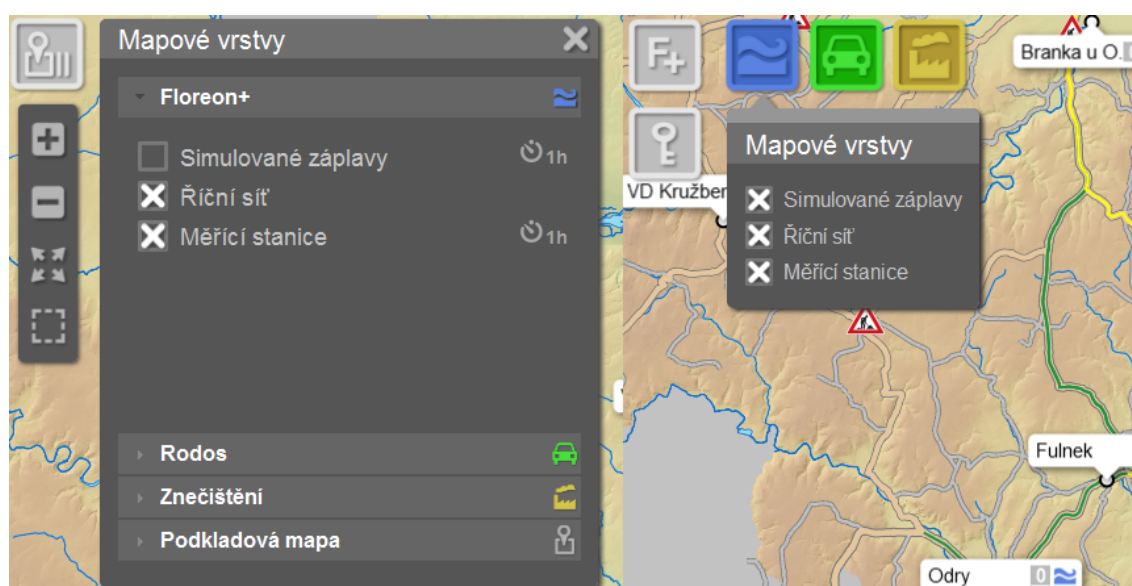
- Topografie
- Letecká
- Základní

V této části komponenty lze zvolit mapový podklad pro mapovou kompozici. Aktuálně systém Floreon+ poskytuje zobrazení ve třech základních podkladových



mapách - „Topografie“ reprezentuje podklad znázorňující výškové rozdíly v terénu, „Letecká“ podkladová mapa je kompozice leteckých snímků, dohromady tvořící pohled na mapu z ptáčích perspektivy a „Základní“ podklad zobrazuje pouze územní členění, znázorňující městskou zástavbu, pozemní komunikace a vodstvo.

Pro větší přehlednost a integritu byly všechny tyto možnosti umístěny na jednotném místě ve vhodné komponentě k zobrazení. Oproti původnímu uspořádání (vizte obrázek číslo 9) těchto ovládacích prvků, byla vytvořena jednotná ikona v levém horním rohu webového rozhraní, sloužící k přístupu do komponenty, ve které pro každou jednotlivou skupinu vrstev je samostatný panel. Umístění se stalo pro uživatele více intuitivní a designově zajímavější.



Obrázek 9: Nová komponenta mapových vrstev (vlevo) a stará (vpravo)

## 6.5 Lokalizace jazyka

Veškeré texty v prostředí webového rozhraní systému jsou lokalizovány a spolu s načtením základního dokumentu webové stránky překládány do patřičného jazyka. Aktuálně se webové prostředí automaticky překládá do češtiny, rozšíření o další jazyky je v aktuální fázi řešení systému zbytečné.

### 6.5.1 Algoritmus překladu

Algoritmus zajišťující překlad se nachází v souboru *lang.js*, který obsahuje dvě funkce:

- *switchLanguage(lang, baseElement)* sloužící k překladu elementů „span“, „label“ nebo „title“ označených třídou „lang“
- *switchHelpContent(baseElement)*, sloužící jako pomocná funkce při překladu elementů „div“ s třídou „lang“

V prvním kroku překladu je třeba načíst si veškerý obsah překladového slovníku (vizte podkapitolu 6.5.2) do datové struktury pole. V dalším kroku je volána metoda *find()* s argumentem definujícím třídu „lang“ a cyklus „each“, který způsobí, že pro každý takový nalezený případ, se vykoná část kódu v těle cyklu. Uvnitř cyklu dochází k zjišťování povahy elementu, z důvodů další manipulace s obsaženým textem. Po zjištění povahy elementu následuje volání vhodných metod, pro zjištění textu v elementu, nalezení tohoto textu jako klíče v překladovém slovníku a nahrazení původního textu v elementu hodnotou přiřazenou danému klíči z překladového slovníku.

### 6.5.2 Překladový slovník

Překladový slovník je reprezentován souborem *lang\_dict.js*. Obsahem tohoto souboru je objekt LANG\_RESOURCES s polem výrazů použitých v elementech základního strukturálního souboru *index.html* a s příslušnými výrazy reprezentující klíč v daném jazyce. Aktuálně obsahuje pouze české výrazy.

## 7 What-if analýza

Tato kapitola je věnována what-if analýze (česky „co-kdyby“ analýza), jejíž vývoj byl upřednostněn před vývojem modulu pro vkládání krizových událostí uživateli systému Floreon. Modul what-if analýzy by se měl stát v budoucnu nedílnou součástí nástrojů využívaných záchrannými složkami jako je policie, hasiči nebo záchranáři.

### 7.1 Co je What-If analýza

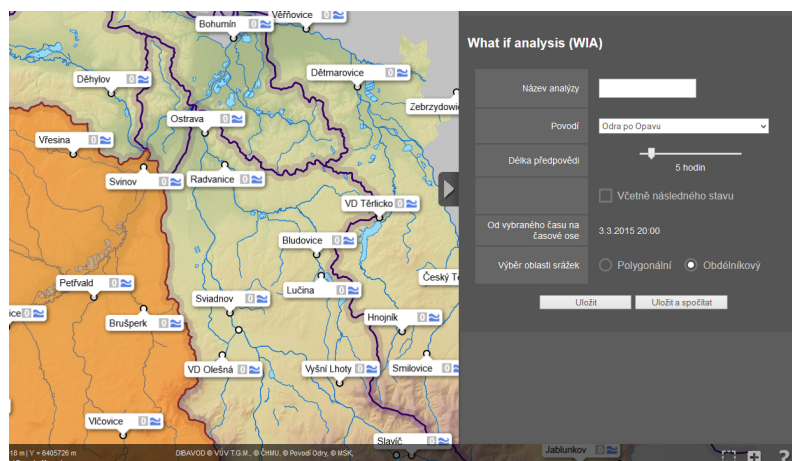
What-if analýza (zkr. WIA) je analytická technika používána při řízení rizik. Její princip stojí na hledání možných dopadů vybraných situací. Díky predikčním algoritmům a dostatku relevantních dat lze spustit simulaci možného vývoje situace a díky této predikce přizpůsobit formu řešení.

WIA systému Floreon+ je v aktuální formě vytvořena pro predikci záplav na základě informací o srážkách v okolí měřících stanic v povodích Moravskoslezského kraje. Vstupními daty je tedy povodí, časové rozmezí a oblast vybraná na mapě s informací o intenzitě srážek. Tyto informace procházejí predikčními algoritmy a výsledkem je vizualizace rozlivu na mapě včetně jednotlivých hydrogramů u měřících stanic. Nově vzniklá komponenta a její vstupní formulář lze vidět na obrázku 10

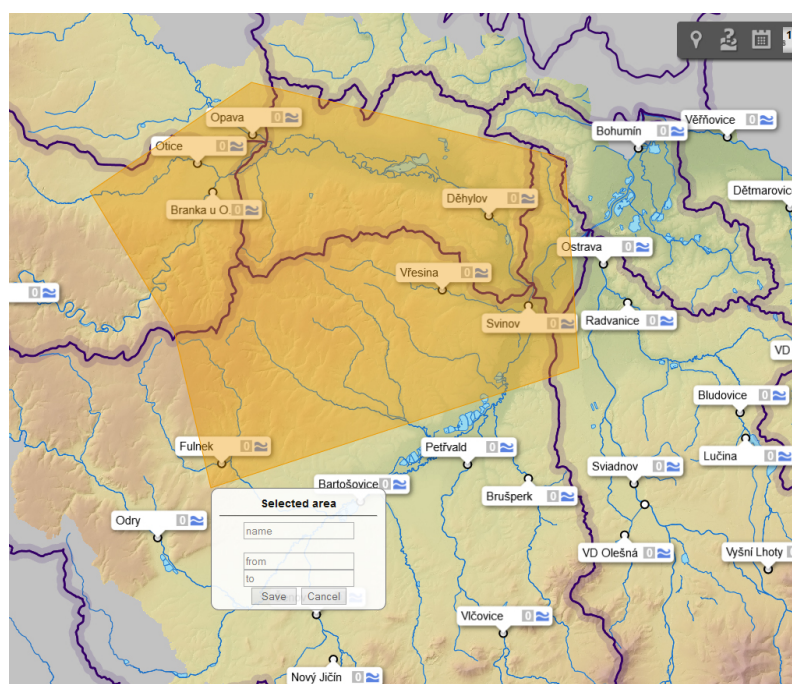
### 7.2 Polygonální výběr

Jednou z nejzásadnějších komponent what-if analýzy je polygonální výběr na mapě. Jedná se o možnost „nakreslit“ na mapovém podkladu polygon, odpovídající oblasti srážek (vizte obrázek 11). Uživateli se při vytvoření nové what-if analýzy zobrazí v pravém panelu formulář s dotazy na vstupy nutné k výpočtu predikce. Při zadávání srážek má uživatel možnost volby mezi dvěma typy polygonálního výběru, a to mezi nvrcholovým polygonem nebo obdelníkem. Spolu se zavoláním kterékoliv z těchto dvou možností, se provede vytvoření nové vrstvy v mapě. Konkrétně v implementaci systému Floreon+ se tato vrstva nazývá „analysisLayer“. Tato vrstva slouží jako kreslicí plátno pro polygony. Pro vytvoření nového polygonálního výběru se volá funkce `OpenLayers.Control.DrawFeature(layer,object)`, která vrátí objekt polygonu. Jelikož je potřeba zjistit, které měřící stanice leží v kterém polygonu, je třeba pracovat s vrstvou obsahující tyto stanice. Pomocí funkce `getLayersByName(layerName)` objektu `map`, který reprezentuje hlavní mapový objekt, lze uložit do proměnné objekt vrstvy s měřícími stanicema. Jakmile uživatel dokreslí polygon a dvojklikem ukončí zadávání vrcholů polygonu, je volána funkce `beforefeatureadded`, která spustí část kódu ze svého těla ještě před tím, než se nově vzniklý polygon uloží do kreslicí vrstvy. V rámci této funkce se spouští metoda na zobrazení zadávacího formuláře rozsahu srážek. Jakmile uživatel zadá název vzniklého polygonu, rozsah srážek v jednotkách milimetrů za hodinu a dá informaci uložit, spouští se algoritmus pro zjišťování všech měřících stanic ve vytvořeném polygonu. Algoritmus pro výběr polygonu využívá hlavně funkci `intersect(layer)` implementovanou v knihovně OpenLayers. Pomocí této funkce lze získat objekty, které se protínají z jedné a druhé vrstvy. Tento proces se opakuje v cyklu, a v každé iteraci vznikne nový objekt

„WIAItem“, který se naplní daty z objektu měřících stanic a daty zadanými uživatelem. Takto získané objekty se uloží do pole, které je posléze v návratové hodnotě odesláno. Toto pole si dále přebírá funkce na výpis vybraných atributů do tabulky, která umožňuje dodatečnou editaci záznamů.



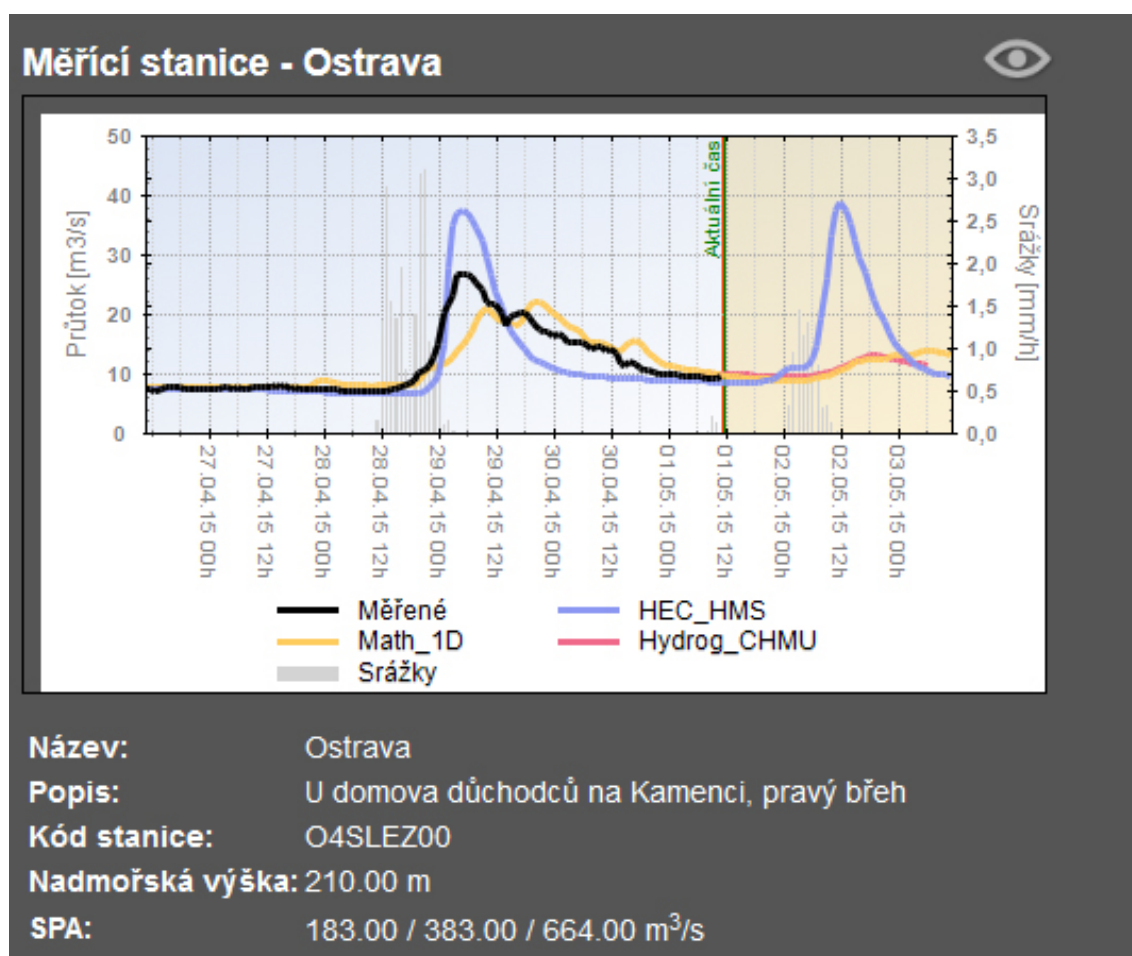
Obrázek 10: Náhled komponenty what-if analýzy



Obrázek 11: Náhled komponenty polygonálního výběru

### 7.3 Vizualizace simulace

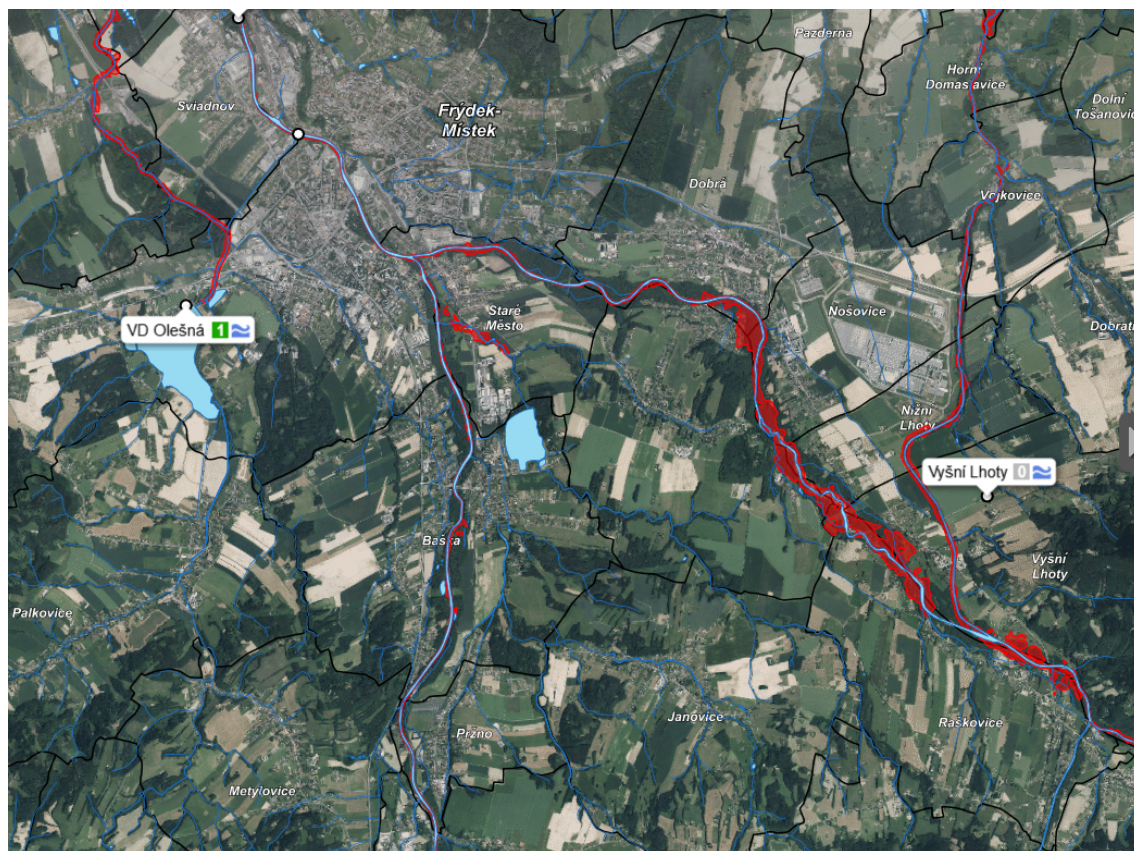
Po zadání všech relevantních vstupních dat se spustí simulace složená ze dvou hlavních částí - výpočet srážkoodtokového a hydrodynamického modelu [16]. Výsledkem srážkoodtokového modelu je vygenerovaný hydrogram (vizte obrázek 12) a výsledkem hydrodynamického modelu rozliv vizualizovaný v mapové komponentě (vizte obrázek 13).



Obrázek 12: Vygenerovaný hydrogram

Srážkoodtokový model slouží k výpočtu teoretické odezvy povodí na déšť, kde významnou roli hraje mohutnost srážky a doba trvání srážky. Výsledkem tohoto modelu je hydrogram, který je zobrazením vývoje průtoku na určitém místě řeky (v našem případě na stanici) v čase.





Obrázek 13: Vizualizace rozlivu v mapové komponentě

Rozliv je na mapě zobrazen jako oblast kolem řeky zalitá vodou při intenzitě a délce zadaných srážek.



## 8 Závěr

Webové rozhraní systému Floreon+ bylo přeuspořádáno, sjednoceno a vylepšeno. Komponenty se staly uživatelsky více intuitivní, části související s jednou funkcionalitou byly sjednoceny na vhodná místa. Za nejdůležitější části a zároveň úspěchy považují tyto - dnes je možné se do systému registrovat, vlastnit účet skrze který lze systém personalizovat - vybrat si výchozí pozici na mapě po přihlášení, výchozí formát souřadnic, výchozí časovou stopu na časové ose, vytváření a uchovávání vlastních what-if analýz. Mimo uživatelská nastavení přibyla funkční legenda, která se přepisuje podle aktuálně používaných mapových vrstev, byl upraven modul pro výběr mapových vrstev, veškeré funkce s mapovými vrstvami jsou sjednoceny na jednom místě.

Na základně těchto inovací systém umožňuje širokou škálu možností dalších rozšíření. Na odvedenou práci naváže vývoj komponenty k vytváření uživatelských událostí. V této komponentě by každý přihlášený uživatel mohl ukládat do systému vlastní informace o aktuální krizové situaci. Tyto události by se zobrazovaly na mapě všem ostatním uživatelům. U každé takto zadané události by se nastavil, nebo aktualizoval čas trvání, po uplynutí tohoto času by se tato událost stala historickou. Jedinou komplikací by mohla znamenat verifikace údajů, aby nedocházelo k vytváření irelevantních nebo nevhodných událostí. V budoucnosti lze předpokládat, že bude systém Floreon+ rozšířen na celorepublikové měřítko.

## 9 Reference

- [1] O projektu Floreon+, <http://floreon.vsb.cz/base/>
- [2] Vše o HTML5, <http://www.html5.cz/>
- [3] Seriál o jQuery, <http://jquery-navod.cz/serial/serial-o-jquery>
- [4] jQuery Tutorial, <http://www.w3schools.com/jquery>
- [5] SVG Tutorial, <http://www.w3schools.com/svg>
- [6] SVG Drawing tutorial, <http://www.inkscape.org/en/learn/tutorials>
- [7] Openlayers intro, <http://workshops.boundlessgeo.com/openlayers-intro>
- [8] Openlayers documentations, <http://trac.osgeo.org/openlayers/wiki/Documentation>
- [9] Web Map Service, <http://www.opengeospatial.org/standards/wms>
- [10] Web Feature Service, <http://www.opengeospatial.org/standards/wfs>
- [11] Captcha, <http://cs.wikipedia.org/wiki/CAPTCHA>
- [12] JavaScript Object Notation, <http://cs.wikipedia.org/wiki/JavaScriptObjectNotation>
- [13] Ajax, <http://cs.wikipedia.org/wiki/AJAX>
- [14] Windows Communication Foundation, <https://msdn.microsoft.com/en-us/library>
- [15] RODOS, <http://www.it4i-rodos.cz/>
- [16] Bedient P.B., Huber W.C., Vieux B.E., *Hydrology and floodplain analysis*, 5th edition, Pearson, Essex, 2013.

## 10 Přílohy

1. CD disk - obsahuje elektronickou formu této bakalářské práce, obrázky, diagramy.
2. Text bakalářské práce kolegy Jana Křenka na téma zátěžového testování systému Floreon+ - taktéž k nalezení na CD