

Univerzita Palackého v Olomouci
Přírodovědecká fakulta
Katedra geoinformatiky

IMPLEMENTACE PROCESNÍ WEBOVÉ SLUŽBY

Bakalářská práce

Daniel URBAN

Vedoucí práce Mgr. Rostislav Néték, Ph.D.

Olomouc 2020
Geoinformatika a geografie

ANOTACE

Tato bakalářská práce se zejména zabývá implementací JavaScript knihovny Turf.js do funkční mapové aplikace ve webovém prostředí. V teoretické části se autor zabývá použitými technologiemi a metodami, vymezí pojem procesní webová služba a zabývá se popisem stávajících řešení na poli mapových aplikací ve webovém prostředí, které v sobě implementují prostorové operace a analýzy. V praktické části je vytvořená aplikace s implementovanou knihovnou Turf.js otestována a srovnána s alternativními řešeními. Na závěr autor uvádí možný směr vývoje aplikace s navrhuje další možnosti rozvoje.

KLÍČOVÁ SLOVA

Procesní webová služba; WPS; Turf.js; Mapbox GL JS; implementace; aplikace

Počet stran práce: 42

Počet příloh: 3 (z toho 2 volné a 1 elektronická)

ANOTATION

This bachelor thesis mainly deals with the implementation of the JavaScript library Turf.js into a functional map application in a web environment. In the theoretical part, the author deals with the technologies and methods used, defines the concept of web processing service and deals with the description of existing solutions in the field of map applications in the web environment, which implement spatial operations and analysis. In the practical part, the created application with the implemented Turf.js library is tested and compared with alternative solutions. Finally, the author states the possible direction of application development and suggests further development options.

KEYWORDS

Web processing service; WPS; Turf.js; Mapbox; implementation, application

Number of pages 42

Number of appendixes 3

Prohlašuji, že

- bakalářskou práci včetně příloh, jsem vypracoval samostatně a uvedl jsem všechny použité podklady a literaturu.

- jsem si vědom, že na moji bakalářskou práci se plně vztahuje zákon č.121/2000 Sb. - autorský zákon, zejména § 35 – využití díla v rámci občanských a náboženských obřadů, v rámci školních představení a využití díla školního a § 60 – školní dílo,

- beru na vědomí, že Univerzita Palackého v Olomouci (dále UP Olomouc) má právo nevýdělečně, ke své vnitřní potřebě, bakalářskou práci užívat (§ 35 odst. 3),

- souhlasím, aby jeden výtisk bakalářské práce byl uložen v Knihovně UP k prezenčnímu nahlédnutí,

- souhlasím, že údaje o mé bakalářské práci budou zveřejněny ve Studijním informačním systému UP,

- v případě zájmu UP Olomouc uzavřu licenční smlouvu s oprávněním užití výsledky a výstupy mé bakalářské práce v rozsahu § 12 odst. 4 autorského zákona,

- použít výsledky a výstupy mé bakalářské práce nebo poskytnout licenci k jejímu využití mohu jen se souhlasem UP Olomouc, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly UP Olomouc na vytvoření díla vynaloženy (až do jejich skutečné výše).

V Olomouci dne

Daniel Urban

Děkuji vedoucímu práce Mgr. Rostislavu Nétkovi, Ph.D. za podněty, připomínky a cenné rady při vypracování práce.

UNIVERZITA PALACKÉHO V OLMOUCI

Přírodovědecká fakulta
Akademický rok: 2018/2019

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Daniel URBAN
Osobní číslo: R17431
Studijní program: B1301 Geografie
Studijní obor: Geoinformatika a geografie
Téma práce: Implementace procesní webové služby
Zadávající katedra: Katedra geoinformatiky

Zásady pro vypracování

Cílem práce je prozkoumat možnosti využití a nasazení procesních služeb ve webové kartografii. Student se zaměří na knihovnu turf.js umožňující provádět prostorové operace ve webovém prohlížeči, ověřit technické, datové, geoinformatické a kartografické možnosti, především se zaměří na implementaci a otestování. Výstupem bude případová studie demonstrující různé operace nad různými typy dat. Dále student provede porovnání s alternativními možnostmi jako geoprocessingové nástroje v ArcGIS Online či GIScloud, a vymezení vůči obecnému principu webových procesních služeb (WPS). Celá práce, tj. text včetně všech příloh, posteru, výstupů, zdrojových i vytvořených dat, map, programových kódů a databází, student odevzdá v digitální podobě na paměťovém nosiči (CD, DVD, SD karta, flash disk) s popisem (jméno, název, KGI, rok). Text práce s přílohami odevzdá ve dvou svázaných výtiscích na sekretariát katedry ve stanoveném termínu. O práci student vytvoří webovou stránku v souladu s pravidly dostupnými na stránkách katedry. Práce bude zpracována podle obecných zásad (Voženílek, 2002) a závazné šablony pro kvalifikační práce na KGI. Povinnou přílohou práce bude poster formátu A2.

Rozsah pracovní zprávy: max. 50 stran
Rozsah grafických prací: dle potřeby
Forma zpracování bakalářské práce: tištěná

Seznam doporučené literatury:

Kraak M.J., Brown A. Web Cartography: Developments and Prospects, 2003. Mapová knihovna Leaflet dostupné online: <https://leafletjs.com/> Zhang Ch, Liping D. et al (2018): Exploring cloud-based Web Processing Service: A case study on the implementation of CMAQ as a Service.
Environmental Modelling & Software, Pages 29-41. Muehlenhaus I. Web Cartography: Map Design for Interactive and Mobile Devices, 2013. Nétek R., Burian T. (2018) Free and open source v geoinformatic. Univerzita Palackého v Olomouci. 118s. ISBN: 9788024452913
Turf.js – dostupné online: <https://turfjs.org/>
Voženílek, V.: Diplomové práce z geoinformatiky. Olomouc, Univerzita Palackého v Olomouci, 2002. 31 s.

Vedoucí bakalářské práce: Mgr. Rostislav Nétek, Ph.D.
Katedra geoinformatiky

Datum zadání bakalářské práce: 6. května 2019
Termín odevzdání bakalářské práce: 6. května 2020

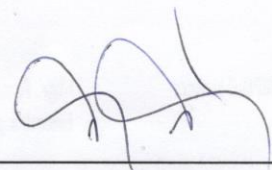
ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Pracovní úkol	1. Úvod
Pracovní úkol	2. Teoretická část
Pracovní úkol	3. Praktická část
Pracovní úkol	4. Závěr
Pracovní úkol	5. Literatura

Podpis vedoucího katedry

[Faint signature]

L.S.



doc. RNDr. Martin Kubala, Ph.D.
děkan

prof. RNDr. Vít Voženílek, CSc.
vedoucí katedry

OBSAH

SEZNAM POUŽITÝCH ZKRATEK	9
ÚVOD	10
1 CÍLE PRÁCE.....	11
2 VYMEZENÍ POJMŮ	12
2.1 Webová kartografie, webová mapa a další.....	12
2.2 Knihovna.....	12
2.3 Open Geospatial Consortium	12
2.4 Web Processing Service	13
3 POUŽITÉ TECHNOLOGIE.....	15
3.1 HTML	15
3.2 CSS	15
3.3 JavaScript.....	16
3.3.1 Mapbox GL JS.....	16
3.3.2 Turf.js	16
3.4 GeoJSON.....	17
4 STAV SOUČASNÉ PROBLEMATIKY	18
4.1 ArcGIS Online	18
4.2 GIS Cloud	19
4.3 CARTO	20
4.4 PyWPS.....	21
5 IMPLEMENTACE	22
5.1 Návrh a struktura aplikace	22
5.2 Implementace základní funkcionality.....	23
5.2.1 Přidání mapového pole do aplikace	23
5.2.2 Ovládací prvky	23
5.2.3 Nahrávání vstvy	24
5.2.4 Přepínání vrstev	24
5.3 Implementace operací z knihovny Turf.js.....	25
5.4 Nadstavbové funkce	27
5.5 Ošetření uživatelského vstupu	29
5.6 Dokončení a ladění chyb.....	29
6 TESTOVÁNÍ VÝKONNOSTI.....	30
7 SROVNÁNÍ S ALTERNATIVNÍMI ŘEŠENÍMI.....	35
8 VÝSLEDKY	37
8.1 Výsledky vývoje aplikace	37
8.2 Výsledky testování	37
8.3 Výsledky srovnání s alternativními možnostmi	39
9 DISKUZE.....	40
10 ZÁVĚR	42
POUŽITÁ LITERATURA A INFORMAČNÍ ZDROJE	
PŘÍLOHY	

SEZNAM POUŽITÝCH ZKRATEK

Zkratka	Význam
3D	3 Dimensions
BSD	Berkeley Software Distribution
CSS	Cascading Style Sheets
CSV	Comma-separated values
DHTML	Dynamic Hypertext Markup Language
ESRI	Environmental Systems Research Institute
EULA	End User License Agreement
GB	Gigabyte
GHz	Gigahertz
GIS	Geografický informační systém
GPS	Global Positioning System
GPX	GPS Exchange Format
HDD	Hard Disk Drive
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IETF	Internet Engineering Task Force
JPG	Joint Photographic Experts Group
JSON	JavaScript Object Notation
MIT	Massachusetts Institute of Technology
MPL	Mozilla Public License
OGC	Open Geospatial Consortium
OsGeo	Open Source Geospatial Foundation
PDF	Portable Document Format
PHP	Hypertext Preprocessor
SHP	Shapefile
SSD	Solid State Disk
TIF	Tag Image File
TIN	Triangulated Irregular Network
W3C	World Wide Web Consortium
WiFi	Wireless Fidelity
WPS	Web Processing Service

ÚVOD

Procesní webová služba (WPS, anglicky Web processing service) standardizuje provádění prostorových operací, síťových analýz a práci s geografickými daty ve webovém prostředí pod záštitou organizace OGC. Ačkoliv termín procesní webová služba není tak známý, jedná se o přesun analytických funkcí desktopových software jako je například ArcGIS Online do webového prostředí.

Obecně lze v dnešní době pozorovat nástup cloudových aplikací, které běží na serveru a uživatel nemusí instalovat žádný software na svoji počítačovou sestavu. Takové aplikace mnohdy dokáží zastoupit desktopový software a poradí si i s náročnými operacemi.

Autor bakalářské práce prozkoumává možnosti JavaScriptové knihovny Turf.js, která umožňuje provádět prostorové operace a analýzy ve webovém prostředí a mohla by tak v budoucnu plně nahradit desktopový software.

1 CÍLE PRÁCE

Stěžejním cílem této bakalářské práce je implementace JavaScript knihovny Turf.js, která umožňuje provádět prostorové operace, analýzy a práci s daty, a vytvořit mapovou aplikaci ve webovém prostředí.

Před samotnou implementací je nutné definovat procesní webové služby a prozkoumat možnosti uplatnění ve webové kartografii. Dále je nutné zjistit současný stav problematiky a prozkoumat alternativní možnosti jakou jsou geoprocessingové nástroje v ArcGIS Online či GIS Cloud.

Po implementaci knihovny Turf.js bude provedeno testování, která se zejména zaměří na měření rychlosti provádění jednotlivých operací. Na závěr bude vytvořená aplikace srovnána s alternativními řešeními a zhodnoceny její klady, zápory a možnosti rozšíření do budoucna.

V neposlední řadě bude v rámci bakalářské práce vytvořen poster formátu A2 a webové stránky, které budou shrnovat cíle, metody, výsledky práce a závěr. Na webových stránkách bude možno si stáhnout text práce ve formátu PDF a zmiňovaný poster. Webové stránky budou vytvořeny v souladu s pravidly dostupnými na stránkách katedry.

2 VYMEZENÍ POJMŮ

Vzhledem k rozvíjejícím se technologiím a stále častějšímu používání pojmů jako mapový portál, mapová aplikace apod. je vhodné několik na počátku vymezit několik pojmů, aby v této bakalářské práci následně nedošlo k omylům a nejistotám. Níže je uvedeno základní vymezení, tato práce si nicméně neklade za cíl přesné definování uvedených pojmů. Dále jsou v této kapitole vymezeny pojmy, které se objevují v bakalářské práci, zejména je pak vysvětlen pojem procesní webová služba.

2.1 Webová kartografie, webová mapa a další

Mapový portál je podle Nětka (2008) popsán následovně: „*chápeme jím server, na kterém jsou pod jednotlivými odkazy umístěny elektronické mapy – tedy internetový portál, věnovaný mapám. Z technického hlediska je nejobvyklejší tzv. třívrstvá architektura, tzn., že celá aplikace se obvykle skládá ze tří částí: mapového serveru, webového serveru a rozhraní pro správu dat.*“ [0]

Mapová aplikace je podle Nětka (2008) popsána následovně: „*soubor nástrojů a funkcí, které umožňují prohlížení mapových úloh. Mapová úloha je soubor rastrových nebo vektorových vrstev, věnovaných určitému tématu, tedy položka, kterou vybíráme v menu mapového portálu.*“ [0]

Webová kartografie je podle Kraak a Brown (2001) definována jako „*obor zabývající se tvorbou, zobrazováním a údržbou map v prostředí webu. Webová mapa je produktem webové kartografie.*“ [1]

Webová aplikace je počítačový program, který využívá webové prohlížeče a webové technologie k plnění úloh přes rozhraní Internetu. [2]

2.2 Knihovna

V bakalářské práci jsou použity dvě knihovny, které jsou popsány v kapitole 3.3. Softwarová knihovna se obvykle skládá z předem napsaného kódu (v tomto případě v jazyce JavaScript), tříd, procedur, funkcí a konfiguračních dat. Vývojář obvykle přidává knihovnu do programu ručně za účelem dosažení více funkcí a zautomatizování procesu. V tomto případě jsou v knihovně Turf.js definovány pomocí funkcí prostorové operace. Všechny dostupné funkce v softwarové knihovně volat a používat aniž by byly explicitně definovány. [3]

2.3 Open Geospatial Consortium

Než se dostaneme k samostatnému pojmu WPS, je třeba vědět, co znamená OGC. Open Geospatial Consortium (OGC) je mezinárodní standardizační organizace pro oblast prostorových dat a služeb. Standardy jsou vyvíjeny v rámci relativně otevřeného procesu. Po svém schválení jsou bez omezení dostupné všem a jsou v praxi vždy implementovány.

Jednou z oblastí standardizace OGC jsou webové služby. Jejich podstatou je komunikace typu stroj – stroj, kdy spolu komunikují dva stroje na základě určitých standardizovaných protokolů (HTTP). Výhodou webových služeb je interoperabilita mezi odlišnými platformami (Java, .NET, JavaScript, PHP).

Mezi webové služby OGC patří například Web Map Service, Web Map Tile Service, Web Feature Service, Web Coverage Service nebo Web Processing Service, která je popsána níže. [4][5]

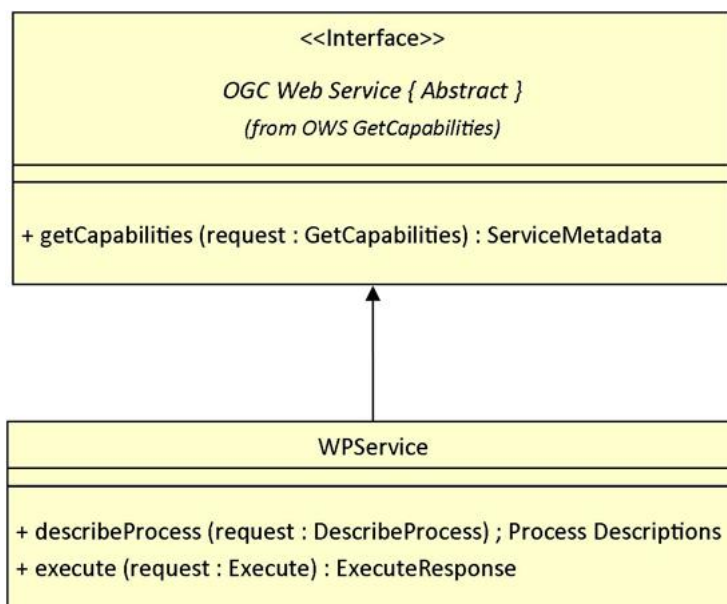
2.4 Web Processing Service

Jak je zmíněno výše, WPS (Web Processing Service, česky Procesní webová služba) je standardem organizace OGC. Služba WPS umožňuje ve webovém rozhraní pracovat s geografickými daty a provádět nad nimi prostorové operace, síťové analýzy nebo úpravy prostorových dat. WPS ve verzi 2.0 byla vydána v roce 2015. Předchozí verze 1.0.0 byla zveřejněna v červnu 2007. Standard WPS deklaruje tři operace, které může klient po serveru požadovat. Těmito operacemi jsou:

GetCapabilities – Operace poskytne klientovi metadata o konkrétních implementovaných operacích, které server nabízí. GetCapabilities poskytuje obecný popis operace a jejich názvy.

DescribeProcess – DescribeProcess dává klientovi podrobné informace o procesech, včetně požadovaných vstupů, jejich přípustných formátech a možných výstupech.

Execute – Tato operace umožní klientovi spustit implementovanou operaci, do které vstoupí zadané parametry a vrátí výslednou vrstvu.



Obrázek 1 - Schéma komunikace WPS. [6]

V praxi to může vypadat následovně. Uživatel chce vytvořit obalovou zónu (buffer) nad vstupní vrstvou. Odpověď na operaci GetCapabilities je ta, že server operaci buffer podporuje a že vstupní vrstvou mohou být body, linie nebo polygony. Odpovědí na dotaz DescribeProcess pro operaci buffer je pak možnost nastavení velikosti obalové zóny včetně výběru jednotek. Uživatel následně spustí proces operací Execute, která vrátí nově vytvořenou vrstvu obalové zóny nad vstupní vrstvou.

Web Processing Service nedeklaruje žádné konkrétní procesy, které jsou podporovány. Místo toho tyto konkrétní procesy, vstupy a výstupy definuje každá samostatná implementace WPS. WPS lze považovat za abstraktní model webové služby, který je nutné implementovat do webové aplikace. Každou operaci definuje GetCapabilities a požadavky jsou založeny na HTTP Get a Post. Standard pomocí těchto požadavků dále definuje:

- Kódování požadavku na provedení procesu

- Kódování odpovědi z provedeného procesu
- Vkládání dat a metadat do vstupů a výstupů prováděného procesu
- Referenční datové vstupy a výstupy přístupné na webu
- Podporu dlouhodobých procesů
- Informace o stavu procesu
- Informace o chybách, které nastaly během procesu
- Uložení výstupů z procesu

S postupným přechodem aplikací do cloudu lze říct, že WPS má do budoucna velký potenciál. Procesní webové služby dokáží webovým mapám přidat hodnotu ve smyslu možnosti provádění prostorových operací, analýz a práci s daty. Postupně mohou webové aplikace, které v sobě integrují procesní služby, nahradit desktopový software. Webové aplikace se standardem WPS mohou být již natolik pokročilé, že mnohdy dokáží zastat veškeré funkce, které nabízí desktopový software. WPS tedy přidává do webové kartografie přidanou hodnotu, jedná se však pouze o deklaraci a popis procesů na pozadí, nedefinuje uživatelské prostředí, kartografickou symbologii nebo interakci s uživatelem. [7][8]

3 POUŽITÉ TECHNOLOGIE

V této kapitole jsou stručně popsány technologie, které byly použity při tvorbě této bakalářské práce. Pro vytvoření mapové aplikace byly využity JavaScript knihovny Mapbox GL JS, která slouží pro vizualizaci, a Turf.js, která umožňuje provádět prostorové operace nad daty. Struktura webových dokumentů je v jazyce HTML a detailnější vzhled je definován pomocí kaskádových stylů.

3.1 HTML

HTML, nebo-li Hypertext Markup Language je značkovací jazyk, který v roce 1990 vyvinul Tim Berners-Lee. Jazyk slouží k tvorbě webových stránek zobrazovaných na World Wide Web. Tyto stránky mohou být vzájemně propojeny pomocí hypertextových odkazů. Pomocí HTML kódu lze zajistit správné formátování textu, obrázků či tabulek tak, aby je webový prohlížeč zobrazil správně. HTML kód poskytuje základní strukturu stránky, o detailnější vzhled a chování se pak starají kaskádové styly (CSS), které mohou být vloženy přímo v HTML kódu nebo v samostatném souboru s příponou .css

HTML kód se skládá z tzv. značek (tags), jejichž použití je většinou triviální. Většina tagů se skládá z úvodní značky, která obsahuje atributy, a uzavíracího tagu, kterým je značka uzavřena. [9]

Struktura značkovacího jazyka HTML:

```
<!DOCTYPE html>
<html>
<head>
<title>Název stránky</title>
</head>
<body>

<h1>Toto je nadpis.</h1>
<p>A toto je odstavec.</p>

</body>
</html>
```

HTML5 je verzi jazyku HTML a je nástupcem svého předchůdce HTML4 (další verzi může být například XHTML). HTML5 používá stejná základní pravidla a principy jako HTML4, ale navíc přidává některé nové tagy, které umožňují lépe pracovat se sémantikou, a umožňují za použití jazyku JavaScript užívání dynamických prvků. [9]

3.2 CSS

Jak již bylo zmíněno v předchozí kapitole, CSS (Cascading Style Sheets) jsou kaskádové styly, které se starají o vzhled, případně chování jednotlivých prvků v HTML dokumentu. Styl každého prvku určuje jakým způsobem se zobrazí a jak se bude chovat. Typickým příkladem může být barva, umístění prvku, velikost či font písma nebo průhlednost. Slovo kaskádové, které je v názvu, značí základní vlastnost CSS – jednotlivá pravidla se mohou vzájemně překrývat, což zvyšuje jejich efektivnost. Pokud jsou kaskádové styly správně použity, umožňují kompletní oddělení vzhledu dokumentu od jeho obsahu, který je napsán v HTML. [10]

Vznik kaskádových stylů se datuje k roku 1997 a jejich vytvoření iniciovala organizace W3C, která se stará o vývoj internetových standardů. [11][12]
Struktura kaskádových stylů:

```
p {  
  color: red;  
  text-align: center;  
}
```

Aktuální používaná verze kaskádových stylů je verze 3 (**CSS3**). Její vývoj trvá více než 10 let a stále nebyl ukončen. Proti starším verzím podporuje nové funkce, jako například animace, a usnadňuje vývojáři stylování webových stránek. [13]

3.3 JavaScript

JavaScript je objektově orientovaný skriptovací jazyk, který slouží především k vytváření interaktivních webových stránek. Příkladem použití může být kontrola vyplnění formulářů, rozbalovací menu, měření statistik návštěvnosti nebo efekty po přejetí myši. V kontextu mapových portálů bychom mohli mluvit například o přepínání podkladových map či přibližování a oddalování mapového pole. Na rozdíl od serverově orientovaných programovacích jazyků, kterým je například PHP, běží JavaScript na straně klienta. [14][15]

Spojením jazyku JavaScript s HTML a CSS vzniká tzv. DHTML (Dynamic Hypertext Markup Language) a má za cíl tvorbu dynamických a interaktivních webových stránek. [16] V jazyce JavaScript pracují některé mapové knihovny, které jsou zmíněny níže a které byly implementovány do mapové aplikace, která je součástí bakalářské práce.

3.3.1 Mapbox GL JS

Mapbox GL JS je JavaScript knihovna, kterou vyvíjí společnost Mapbox a která slouží k tvorbě a vizualizaci dat v interaktivní mapě ve webovém prostředí. Jedná se o součást multiplatformního ekosystému Mapbox GL, jehož součástí je například Mapbox Mobile. Knihovna poskytuje spoustu nástrojů pro vizualizaci a interaktivitu s uživatelem a pro vykreslení používá technologii WebGL, která umožňuje zapojit do vykreslení i grafický čip počítačové sestavy. Navíc je možné rozšířit knihovnu o několik pluginů (například geokódování). Mezi další zvažované knihovny, které vizualizují data a umožňují tvorbu interaktivních map, patří například Leaflet nebo OpenLayers.

Aktuální verze knihovny ke dni 23.5.2020 je 1.10.1. Jedná se o freemium produkt a uživatel si musí vygenerovat Access token. Verze zdarma umožňuje až 50 000 načtení mapy měsíčně. Po překročení limitu se cena za 1 000 zobrazení pohybuje od 3 do 5 dolarů. [17][18][19]

Pro implementaci knihovny byla použita rozsáhlá dokumentace s množstvím příkladů z oficiálních webových stránek.

3.3.2 Turf.js

Pro prostorové analýzy a operace byla použita knihovna Turf.js. Jedná se o pokročilou knihovnu s otevřeným zdrojovým kódem v jazyce JavaScript, která umožňuje provádět prostorové operace v prohlížeči. Je distribuována pod licencí MIT a podle webové stránky GitHub, kde je zdrojový kód poskytnut, je kód nejvíce aktualizován kanadskými programátory Denisem Carrierem a Morganem Herlockerem. [22]

Knihovna podle oficiálních webových stránek nabízí přes 100 operací rozdělených do několika skupin jako například měření, transformace, převod jednotek, agregace nebo interpolace. Základním podporovaným formátem je GeoJSON. [20][21]

3.4 GeoJSON

GeoJSON je otevřený formát pro výměnu a kódování geografických dat založený na JavaScript Object Notation (JSON). GeoJSON byl poprvé specifikován v roce 2008, v roce 2015 byla specifikace nahrazena novou, kterou vytvořila skupina IETF (Internet Engineering Task Force). Současná podoba standardu je ze srpna roku 2016. Formát používá jako souřadnicový systém World Geodetic System 1984 a jednotky desetinných stupňů. Extenzí formátu GeoJSON je formát TopoJSON. [23][24]

GeoJSON umožňuje uložení následujících prostorových dat:

- Body
- Linie
- Polygony
- Multipolygony
- Multilinie
- Prvky

Informace jsou ukládány do datových typů:

- String (textový řetězec)
- Číslo (celočíselné nebo reálné)
- Boolean (logická hodnota)
- Prázdná hodnota (Null)
- Pole, objekt

Struktura formátu GeoJSON:

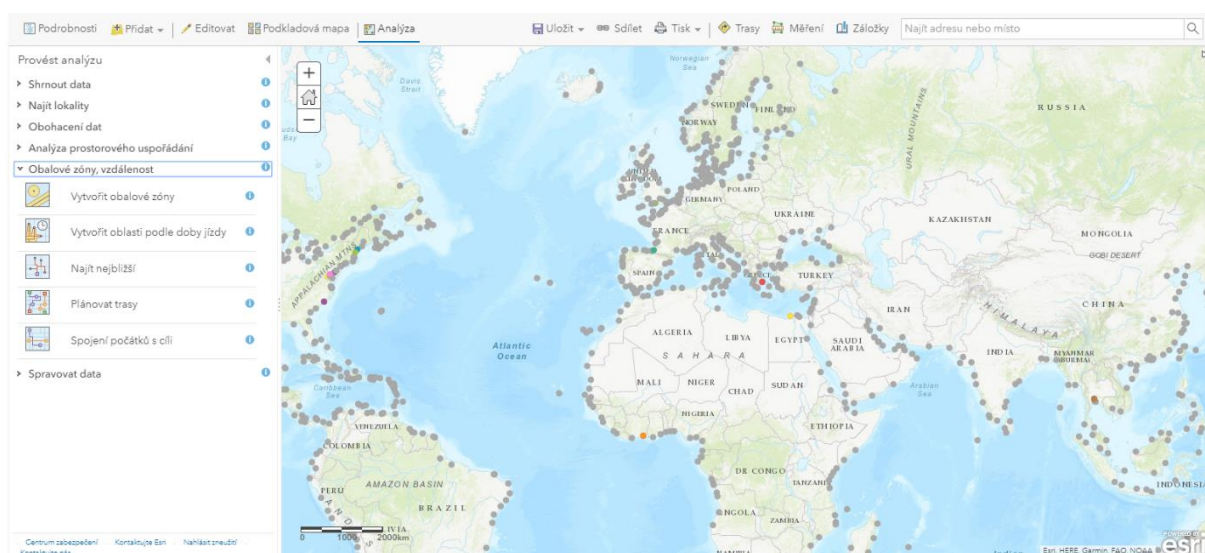
```
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "properties": {
        "Nazev": "Praha"
      },
      "geometry": {
        "type": "Point",
        "coordinates": [
          14.425048828125,
          50.0923932109388
        ]
      }
    }
  ]
}
```

4 STAV SOUČASNÉ PROBLEMATIKY

Tato kapitola se zabývá průzkumem stávajících řešení v oblasti mapových aplikací ve webovém prostředí, které umožňují provádět prostorové operace.

4.1 ArcGIS Online

ArcGIS Online je nástroj pro publikaci, prohlížení a sdílení dat, interaktivních map a aplikací v prostředí internetu od společnosti ESRI, která vyvíjí například desktopový software ArcGIS Pro. Společnost založil v roce 1969 Jack Dangermond v Redlands v USA. Do aplikace ArcGIS Online je možné nahrát data několika způsoby – podporované jsou například webové služby WMS či soubory ve formátu SHP, GeoJSON, CSV či GPX. Součástí aplikace je i několik analytických nástrojů, jejich výčet naleznete níže. Za každou provedenou operaci jsou uživatelé odečteny body, které si musí zakoupit. Podporované jsou téměř všechny webové prohlížeče a to i v mobilních verzích. [26]



Obrázek 2 - Prostředí aplikace ArcGIS Online.

Operace jsou rozděleny do následujících kategorií:

Shrtnout data

- Seskupit body
- Připojit prvky
- Souhrn hodnot v okolí
- Souhrn hodnot v rozmezí
- Souhrn středové hodnoty a rozptýlení

Najít lokality

- Najít stávající lokality
- Odvodit nové lokality
- Vyhledat centroidy
- Najít podobné lokality
- Zvolit nejlepší zařízení

- Vytvořit viditelný povrch
- Vytvořit povodí
- Sledovat po proudu

Obohacení dat

- Obohatit vrstvu

Analýza prostorového uspořádání

- Vypočítat hustotu
- Hot spot analýza
- Najít odlehlé prvky
- Najít shluky bodů
- Interpolovat body

Obalové zóny, vzálenost

- Vytvořit obalové zóny
- Vytvořit oblasti podle doby jízdy
- Najít nejbližší
- Plánovat trasy
- Spojení počátků s cíli

Spravovat data

- Sloučit hranice
- Extrahovat data
- Generovat pravidelné mřížky polygonů
- Spojit vrstvy
- Překrýt vrstvy

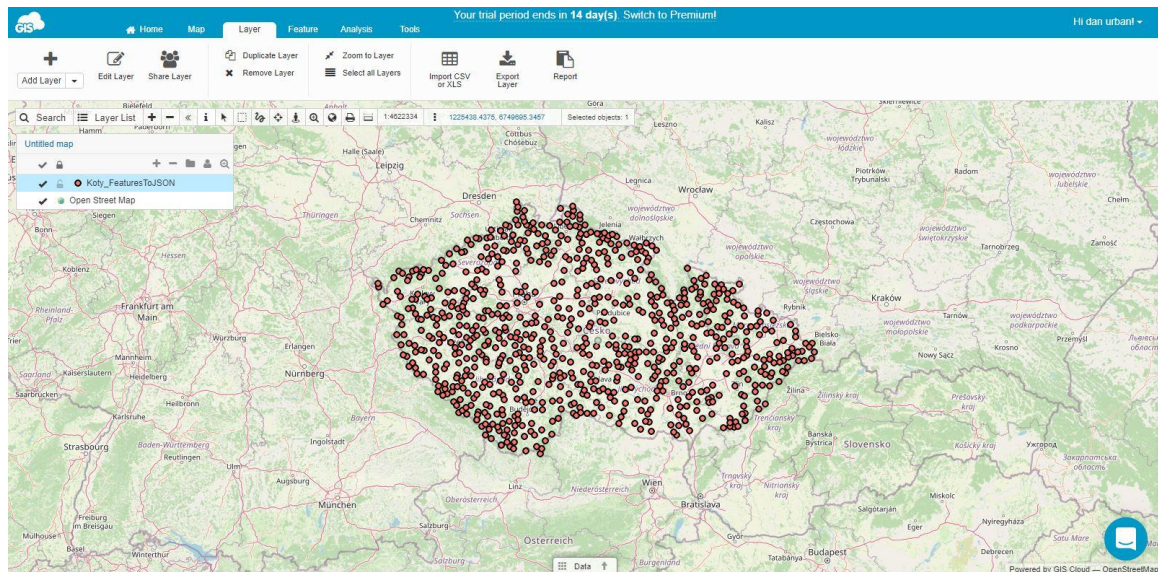
ArcGIS Online nabízí celkem 29 prostorových operací a veškeré výpočty se odehrávají na serverech společnosti ESRI. [25]

4.2 GIS Cloud

GIS Cloud je aplikace od stejnojmenné evropské společnosti. Podporuje řadu vektorových formátů (SHP, GPX, TIF, ...), rastrových formátů a webových služeb (WMS). GIS Cloud je možné používat ve 14-denní trial verzi, následně je editor zpoplatněn 95 dolary za měsíc. GIS Cloud také nabízí mobilní data collector pro sběr dat v terénu. Ohledně webových procesních služeb je na tom platforma hůře – podporuje pouze čtyři následující operace:

- Buffer
- Heat map
- Near
- Area

Mezi další funkce patří například správa souborů a databází, publikování a export map nebo její tisk. [27]

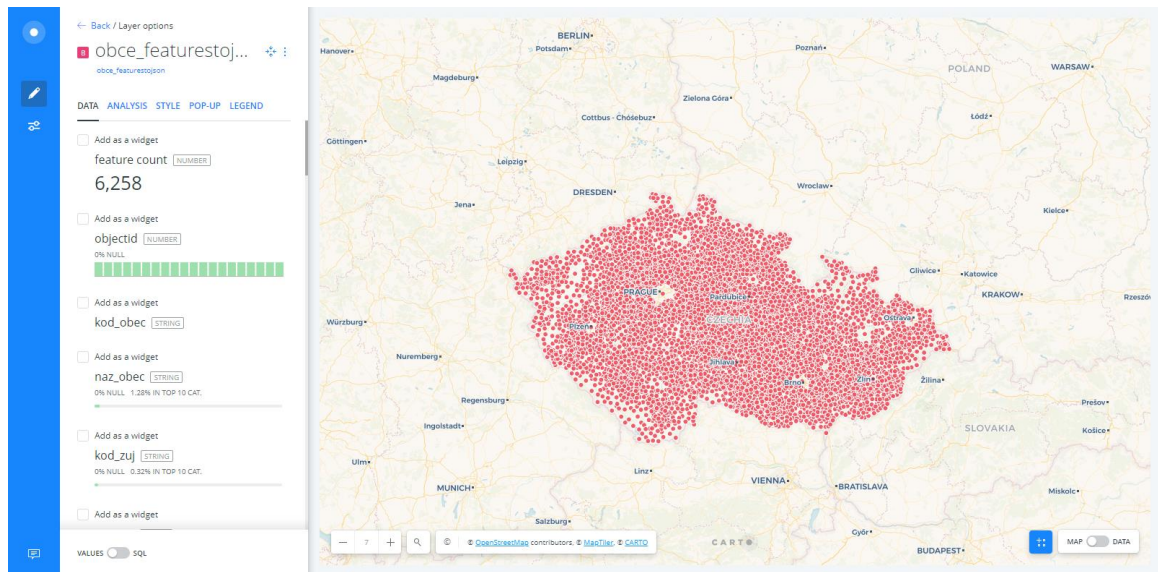


Obrázek 3 - Prostředí aplikace GIS Cloud.

4.3 CARTO

Jednou z platform společností CARTO je mapový portál pro správu a analýzu dat. CARTO podporuje všechny základní formáty jako SHP, GeoJSON, CSV, KML, GPX nebo geodatabáze. Platforma je zdarma do vytvoření 6 map nebo 10 datasetů a nabízí celkem 15 operací: [28]

- Link Second layer
- Geocode
- Add columns from 2nd dataset
- Intersect and aggregate
- Create travel/Distance buffers
- Create centroids of geometries
- Create polygons from points
- Filter by column value
- Select points in polygon
- Create lines from points
- Subsample percent of rows
- Calculate clusters of points
- Detect outliers and clusters
- Predict trends and volatility
- Find nearest



Obrázek 4 - Prostředí aplikace CARTO.

4.4 PyWPS

Další implementací procesní webové služby je PyWPS. PyWPS je OSGeo projekt, který je napsán v programovacím jazyku Python, a jeho začátky se datují do roku 2006. PyWPS založil Jáchym Čepický v rámci svého projektu „Propojení GRASS GIS s UMN MapServerem“, který byl podporován německou nadací pro životní prostředí. V roce 2015 PyWPS oficiálně vstoupil do inkubačního procesu OSGeo. PyWPS byl původně hostován serverem Wald, dnes je však ve skupině GeoPython na serveru GitHub. Od roku 2016 také funguje nová doména PyWPS.org. PyWPS ve verzi 3.x podporuje standard WPS ve verzi 1.0.0, verze PyWPS 4.x, která je stále ve vývoji, již podporuje WPS 2.0.0 a je šířen pod licencí MIT. [29][30]

5 IMPLEMENTACE

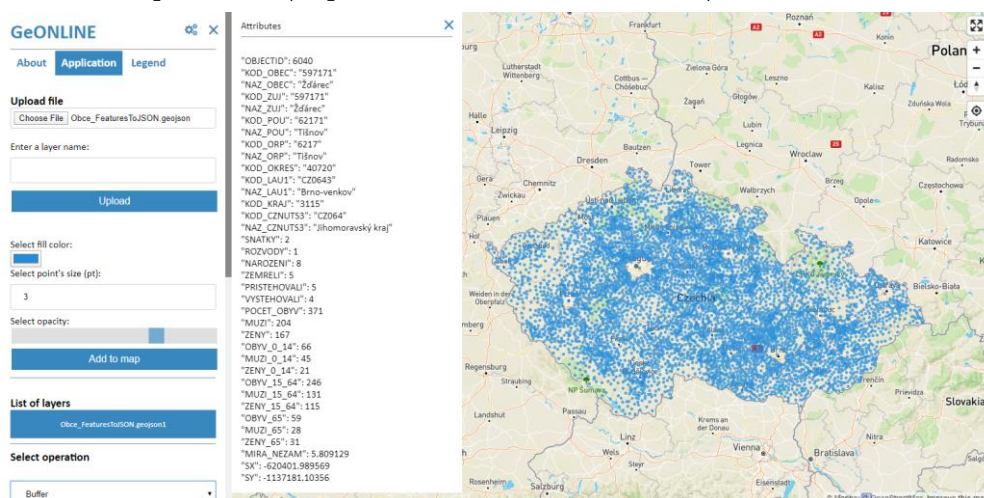
Hlavním výstupem bakalářské práce je návrh a vývoj aplikace, ve které bude možno provádět prostorové operace s geografickými daty a která v sobě bude implementovat knihovnu Turf.js. Pro aplikaci byl zvolen spojením dvou slov „geography“ a „online“ název „GeONLINE“. Pro implementaci byly použity prostředky a technologie jako například HTML, JavaScript a další, které jsou popsány v kapitole 3. Veškerá implementace probíhala z oficiálních stránek Mapbox a Turf.js. Rady a návody hledal autor práce na diskuzních fórech, zejména StackOverflow.

5.1 Návrh a struktura aplikace

Před implementací bylo nutno rozhodnout několik důležitých otázek. Jelikož knihovna Turf.js umožňuje pouze provádění prostorových operací a nedokáže vizualizovat data a vytvořit samotnou webovou mapu, bylo nutné rozhodnout jakou JavaScript knihovnu autor použije pro vizualizaci a interakci s uživatelem. Mezi vhodné knihovny patří například Leaflet, Mapbox GL JS nebo Openlayers. Autor zvolil knihovnu Mapbox GL JS ve verzi 1.7.0 pro svou komplexnost, široký výběr pluginů, propracovanou dokumentaci s příklady použití a v neposlední řadě hrála roli spolupráce týmu Mapbox s knihovnou Turf.js. Na oficiálních stránkách Mapbox jsou totiž dostupné tutoriály jak implementovat knihovnu Turf.js do knihovny Mapbox GL JS.

Dalším krokem bylo rozhodnout, zda bude webová mapa postavena na technologiích HTML, CSS a JavaScript nebo k nim bude přidáno databázové řešení. Kvůli časové náročnosti implementace databázové řešení od toho bylo nakonec upuštěno. Ač by databázové řešení umožnilo lepší uložení dat nebo možnost přidání uživatelských účtů, bylo autorem rozhodnuto o zvolení jednodušší a robustnější varianty pro implementaci. V tomto případě se ale například veškeré vrstvy po aktualizování stránky smažou a jsou uloženy pouze v paměti webového prohlížeče.

Autor zvolil, že se celá aplikace bude zobrazovat pouze v jednom okně. Byl vybrán konvenční vzhled webových map s postranním panelem, kde většinu místa zabírá mapové pole. Aplikace je napsána v anglickém jazyce. Důvodem je potenciálně větší základna uživatelů a rozšířenost jazyku ve světě. Autor se domnívá, že znalost anglických pojmů v geoinformatice je mezi odbornou sférou na tak dobré úrovni, že pojmy není třeba překládat (např. Buffer na Obalovou zónu).



Obrázek 5 - Prostředí aplikace GeONLINE.

Aplikace je rozdělena na mapové pole, které zabírá hlavní část displeje a postranní panel, kterým lze aplikaci obsluhovat. Postranní panel je rozdělen do 3 záložek. První About slouží jako krátké představení aplikace a návod na používání, záložka Application je hlavní část postranního panelu. Nachází se v ní nahrávání souborů, seznam vrstev a volba prostorových operací. Poslední záložka Legend slouží k zobrazení jednotlivých kartografických znaků k dostupným vrstvám. Pro přepínání mezi jednotlivými záložkami slouží funkce OpenAboutApp, OpenContent a OpenLegend.

5.2 Implementace základní funkcionality

Podkapitola shrnuje způsob implementace základní funkcionality aplikace. Popisuje jakým způsobem se vykresluje mapové pole do aplikace, dále přidání ovládacích prvků a v poslední řadě nahrání vlastního souboru a následné přepínání vrstev.

5.2.1 Přidání mapového pole do aplikace

Tato podkapitola popisuje přidání mapového pole do aplikace. Na obrázku 6 je ukázka vložení mapy do aplikace. Pro přidání mapy je nejprve nutné stáhnout balíček Mapbox GL JS a vložit jej pomocí tagu `<script src="mapbox-gl.js"></script>` do hlavičky HTML souboru. Dále je potřeba vložit do těla HTML souboru div, do kterého se mapa vykreslí - `<div id="map"></div>`. Hlavní částí je vložení samotného skriptu, který mapu vykreslí. Skript je uložen v externím souboru s názvem map.js a je na něj odkázáno v HTML souboru pomocí tagu `<script src='map.js'></script>`. V kódu je nejprve vložen accessToken, který je nutné vygenerovat na stránkách Mapbox. Následuje definice proměnné map, ve které je definován styl, nebo-li podkladová mapa, počáteční pozice, nastavení počátečního přiblížení a div, ve kterém se mapa vykreslí.

```
//Základní definice mapy
mapboxgl.accessToken = 'pk.eyJ1IjoidXJ1YW45NiIsImEiOiJjazEwZD11bWswNDdqM2hxZHhhejJzc2xpIn0.83Yupx3wE-Le2P9BRjzCkg';
var map = new mapboxgl.Map({
  container: 'map', //Div, ve kterém se vykreslí mapa
  style: 'mapbox://styles/mapbox/streets-v11',
  center: [15.468749999999998, 49.63917719651036], //Počáteční pozice
  zoom: 5 //Zoom level
});
```

Obrázek 6 - Základní definování mapy.

5.2.2 Ovládací prvky

Další základní funkcí, kterou aplikace GeONLINE nabízí, je možnost ovládání mapového pole pomocí ovládacích prvků. Ty jsou umístěny v pravém horním rohu mapové aplikace. Pomocí nich tak lze ovládat přiblížení a oddálení mapového pole, zapnutí celoobrazového režimu, geolokaci a úhel pohledu na mapu. Na obr. č. 7 je k dispozici zdrojový kód implementující ovládací prvky.

```
map.addControl(new mapboxgl.FullscreenControl()); //Tlačítko Fullscreen
map.addControl(new mapboxgl.NavigationControl()); //Tlačítka navigace a ovládání mapy

//Tlačítko geolokace
map.addControl(
  new mapboxgl.GeolocateControl({
    positionOptions: {
      enableHighAccuracy: true
    },
    trackUserLocation: true
  })
);
```

Obrázek 7 - Implementace ovládacích prvků.

5.2.3 Nahrávání vrstvy

V této podkapitole je popsáno nahrávání vlastních souborů do aplikace, což je jedním z hlavních stavebních kamenů celé aplikace. Do aplikace lze nahrát soubor ve formátu GeoJSON. Po kliknutí na tlačítko UPLOAD je podmínkou zjištěno, zda je přípona souboru .geojson a zda byl opravdu vybrán soubor. Následně je zavolána funkce, která metodou JSON.parse přečte soubor a uloží jej do proměnné result, která je typu objekt.

```
//Zjistí zda se jedná o bodovou, liniovou nebo polygonovou vrstvu
if (editLayer.checked == true) {
  if (((result.features[0].geometry.type) == "Point") || ((result.features[0].geometry.type) == "MultiPoint")){
    document.getElementById('upload-point').style.display = 'block';
    document.getElementById('upload-line').style.display = 'none';
    document.getElementById('upload-polygon').style.display = 'none';
    document.getElementById('uploadLayer').style.display = 'block';
  }
}
```

Obrázek 8 - Podmínka, která zjišťuje, zda se jedná o bodovou vrstvu.

Následně funkce zjišťuje, zda se jedná o bodovou, liniovou nebo polygonovou vrstvu. Na obrázku č. 8 je podmínka, která se z prvního prvku v proměnné result dotáže na typ geometrie a následně vloží vrstvu do mapy s předem definovanou symbologií. Pokud uživatel v nastavení zaškrtně možnost vlastní symbologie, před přidáním do mapy se objeví nový formulář s dodatečným nastavením symbologie. Poté je do pole toggleableLayersIds, které slouží jako úložiště veškerých vrstev, zapsáno na nultém indexu jméno vrstvy a na prvním indexu data z proměnné result. Uživatel není nucen zadat jméno vrstvy. Jestliže jej nevyplní, přiřadí se vrstvě jméno podle názvu vstupního souboru. V názvu se na konci nachází číslo, které se s postupně nahrávanými vrstvami inkrementuje. Zajištěn je tím tak unikátní název každé vrstvy, který slouží zároveň jako identifikátor a musí být jedinečný.

5.2.4 Přepínání vrstev

Vrstvy se zobrazují v postranním panelu pod titulem List of layers. Vrstvě je přiřazen název z pole toggleableLayersId. Zároveň s nahráním vrstvy do mapy se vrstvě přidá třída active, která ji v postranním panelu zobrazí modrou barvou. Po kliknutí na danou vrstvu odebere vrstvě třídu active a nastaví ji vlastnosti visibility na hodnotu none. V opačném případě, kdy uživatel požaduje znovuzobrazení vrstvy, nastaví funkce hodnotu visibility na visible a přiřadí vrstvě opět třídu active.

```
var link = document.createElement('a');
link.href = '#';
link.className = 'active';
link.textContent = id;

link.onclick = function(e) {
  var clickedLayer = this.textContent;
  e.preventDefault();
  e.stopPropagation();

  var visibility = map.getLayoutProperty(clickedLayer, 'visibility');

  if (visibility === 'visible') {
    map.setLayoutProperty(clickedLayer, 'visibility', 'none');
    this.className = '';
  } else {
    this.className = 'active';
    map.setLayoutProperty(clickedLayer, 'visibility', 'visible');
  }
};

var layers = document.getElementById('layers');
layers.appendChild(link);
document.getElementById('no-legend').style.display = 'none';
document.getElementById('no-layers').style.display = 'none';
```

Obrázek 9 - Přepínání vrstev v postranním panelu.

5.3 Implementace operací z knihovny Turf.js

Tato podkapitola popisuje nejdůležitější část bakalářské práce. Do aplikace bylo implementováno celkem 14 operací, z toho 2 v záložce Measurement, 9 v Analysis and transformation a 3 v kategorii Random. Výběr byl zvolen tak, aby byl co nejrozmanitější a nabídl přidanou hodnotu od alternativních řešení.

V postranním panelu bylo pod titulem Select operation vytvořeno rolovací menu, ve kterém se nacházejí všechny implementované operace. Zvolením jedné z nich se rozbalí formulář s dodatečnými atributy a základní možností editace symbologie. Každou operaci lze vykonat pouze nad určitým typem dat (bod, linie, polygon) a vhodnost jednotlivých vrstev se zajišťuje stejným způsobem jako u nahrávání souborů, v tomto případě funkcí SuitableLayer. V případě operace Points within polygon jsou vstupem 2 vrstvy a s funkcí SuitableLayer je volána i funkce SecondSuitableLayer. Výsledkem je pole ListOfSuitableLayers, který se vypíše do rolovacího seznamu pod zvolenou operaci. Po vybrání operace, vrstvy a zvolení atributů se po kliknutí na tlačítko SUBMIT zavolá funkci Execute. Ta pomocí metody getElementById() zajistí všechny vstupní parametry a zavolá funkci z knihovny Turf.js, která provede prostorovou operaci. Výsledná vrstva je uložena do pole toggleableLayersId, tedy i do seznamu vrstev v postranním panelu a je zobrazena v mapovém poli. Mimo to je vytvořen odkaz pro stažení nové vrstvy a je zapsán kartografický znak do legendy. Všechny operace fungují na stejném principu a jejich základní charakteristika je k nalezení níže.

```
else if (choice === "linetopolygon") {  
  
    var outputName = document.getElementById("linetopolygon-output-name").value;  
    LayerName(outputName);  
    var lineToPolygoncolor = document.getElementById("linetopolygon-color").value;  
    var lineToPolygonOpacity = parseFloat(document.getElementById("linetopolygon-opacity").value); //nutno Float místo Int  
    var lineToPolygonOutlineColor = document.getElementById("linetopolygon-outline-color").value;  
    var lineToPolygon = turf.lineToPolygon(toggleableLayerIds[a][1]);  
  
    map.addLayer({  
        'id': outputName + counter,  
        'type': 'fill',  
        'source': {  
            'type': 'geojson',  
            'data': lineToPolygon  
        },  
        'layout': {  
            // "line-join": "round",  
            // "line-cap": "round"  
        },  
        'paint': {  
            'fill-color': lineToPolygonColor,  
            'fill-opacity': lineToPolygonOpacity,  
            'fill-outline-color': lineToPolygonOutlineColor  
        }  
    });  
  
    toggleableLayerIds.push([outputName + counter, lineToPolygon]);  
  
    counter = counter + 1;  
  
    DownloadLayer(lineToPolygon, outputName);  
}
```

Obrázek 10 - Kód zajišťující průběh operace Line to polygon.

Calculate area

Do operace Calculate area může vstoupit pouze polygonová vrstva, výsledkem je vypočítaná plocha všech polygonů ve vrstvě. Uživatel má možnost si jako výstupní jednotku vybrat kilometr čtvereční nebo metr čtvereční. Výstupní hodnota se objeví ve vyskakovacím okně prohlížeče.

Calculate length

Do operace Calculate length vstupuje pouze liniová vrstva, výsledkem je vypočítaná délka všech liniových prvků ve vstupní vrstvě. Uživatel má možnost si jako výstupní jednotku vybrat mezi kilometry, mílemi, stupni a radiány. Výstup se objeví ve vyskakovacím okně prohlížeče.

Buffer

Funkce vytvoří obalovou zónu nad vstupní vrstvou, kterou můžou být body, linie i polygony. Výstupem jsou polygony. Uživatel zadá velikost obalové zóny a jednotky, ve které velikost udává. Na výběr jsou kilometry a míle.

Envelope

Funkce vytvoří nad vstupní bodovou vrstvou nejmenší možný pravoúhelník tak, aby obsahoval všechny vstupní body. Výstupem je polygonová vrstva.

Get centroid coords

Operace Get centroid coords vypočítá těžiště, tedy bod, ze vstupní vrstvy, kterou můžou být body, linie nebo polygony. Operace počítá střední hodnoty všech vstupních bodů, což snižuje účinek malých ostrovů apod. při výpočtu na těžiště. Výstupem je bodová vrstva, navíc se ve vyskakovacím okně prohlížeče objeví souřadnice bodu.

Line to polygon

Vstupní vrstvou, jak již název napovídá, jsou linie. Výstupní vrstvou jsou polygony.

Points within polygon

Tato operace má dvě vstupní vrstvy - první je bodová a druhá polygonová. Operace následně vybere všechny body, které se překrývají s polygonovou vrstvou a uloží je do výstupní bodové vrstvy.

Polygon to points

Vstupní vrstvou jsou polygony, výstupem body. Operace vrátí všechny lomové body polygonu jako bodovou vrstvu.

Simplify polygon

Vstupní vrstvou do operace Simplify polygon jsou polygony, výstupem taktéž. Operace používá knihovnu Simplify.js od Vladimíra Agafonkina (tvůrce knihovny Leaflet) a je využit algoritmus Ramer-Douglas-Peucker. Uživatel má možnost zvolit míru zjednodušení polygonové vrstvy a to na stupnici od 0 do 1, kde vyšší číslo znamená větší zjednodušení. Další možností nastavení je export do vysoké kvality. [31][32]

TIN

Funkce vytvoří z bodové vrstvy TIN (Triangular irregular networks), tedy polygonovou vrstvu. Polygony jsou vytvořeny tak, že jsou spojeny vrcholy vstupní bodové vrstvy. K výpočtu trojúhelníků je použit algoritmus Delaunayho triangulace. Do jednotlivých polygonů jsou v atributové tabulce zapsány atributy a, b, c, které obsahují hodnoty vrcholů trojúhelníku. Vstupní atribut volí uživatel před provedením operace. Vhodným atributem je například výška bodu (z-value), výsledkem pak může být 3D model reliéfu. [33]

Voronoi polygons

Do operace vstupuje bodová vrstva a výstupem je vrstva polygonů. Voronoi polygony jsou vytvořeny tak, že každé místo v mnohoúhelníku je blíže vnitřnímu bodu než k jakémukoliv jinému. Operace využívá algoritmus d3-voronoi. Do atributové tabulky polygonové vrstvy se nazapíše atributy ze vstupní bodové vrstvy. [34]

Create random points

Operace create random points generuje vrstvu s požadovaným počtem náhodných bodů. Body se vytváří náhodně po celém světě a uživatel nemá možnost nastavit zónu, ve které se body vytváří. Je možnost nastavit počet bodů, který není nijak omezen. Uživatel nicméně musí počítat, že může být limitován výkonem svého počítače a bude tomu odpovídat čas výpočtu.

Create random lines

Tato operace generuje liniovou vrstvu s požadovaným počtem linií. Uživatel má možnost nastavit počet linií a počet lomových bodů každé linie, kde výchozí hodnota je 1000. Operace je výpočetně náročná a uživatel musí počítat, že může být limitován výkonem svého PC. Generování 1000 linií s ponecháním výchozí hodnoty počtu lomových bodů je ekvivalentní pro funkci Create random points s vytvořením 1 000 000 bodů.

Create random polygons

Operace Create random polygons generuje polygonovou vrstvu s předem definovaným počtem polygonů. Uživatel má možnost nastavit počet polygonů a tato volba není nijak limitována.

5.4 Nadstavbové funkce

Autor v této podkapitole popisuje implementaci funkcí, které mají uživateli zvýšit komfort při užívání aplikace a zároveň dávají aplikaci přidanou hodnotu.

Stažení vrstvy

Zároveň s nově vytvořenou vrstvou je pod titulem List of layers vytvořen odkaz pro stažení. Funkce OdkazNaStazeni je volána po provedené operaci z knihovny Turf.js a vstupují do ní 2 argumenty - vrstva s daty a její jméno. Pomocí funkce encodeURIComponent(JSON.stringify()) jsou uložena data z vrstvy do proměnné data a pomocí metody document.createElement() je vytvořen odkaz, který je umístěn v ikoně pro stažení vrstvy.

```
//Funkce vytvoří odkaz na stažení, odkaz v podobě ikony
function DownloadLayer(dataZOperace, nazev) {
  var data = "text/json;charset=utf-8," + encodeURIComponent(JSON.stringify(dataZOperace));

  var a = document.createElement('a');
  a.href = 'data:' + data;
  a.download = nazev + '.geojson';
  a.innerHTML = '';

  var container = document.getElementById('layers');
  container.appendChild(a);
}
```

Obrázek 11 - Funkce zajišťující vytvoření odkazu ke stažení jednotlivých vrstev.

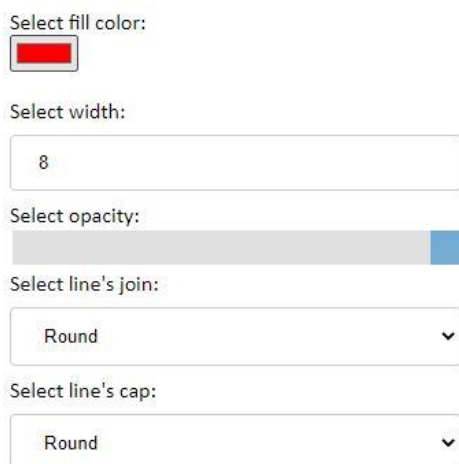
Zobrazení atributů

Další funkcí je zobrazení atributů po kliknutí do mapového pole. Po kliknutí do mapy je zavolána funkce, která zobrazí div, ve kterém budou atributy zobrazeny. Následně je seznam atributů rozdělen pomocí funkce split tak, aby byl každý název atributu i s hodnotou na novém řádku. Kliknout lze jak na uživatelská data, tak do mapového podkladu, kde se zobrazí atributy z databáze společnosti Mapbox.

Nastavení symbologie

Knihovna Mapbox GL JS umožňuje nastavení symbologie pro tyto možnosti: background, fill, line, symbol, raster, circle, fill-extrusion, heatmap a hillshade. Pro bodové vrstvy byla vybrána možnost circle. Pro implementaci byly zvoleny tyto základní parametry - barva, průhlednost a rádius. Pro polygony byly implementovány parametry

barva, průhlednost a barva vnějšího obrysu. Pro linie byly implementovány parametry barva, šířka, druh spojení linií a ukončení linie.



Select fill color:

Select width:

Select opacity:

Select line's join:

Select line's cap:

Obrázek 12 - Možnosti kartografické vizualizace pro liniovou vrstvu.

Legenda

Součástí aplikace GeONLINE je legenda, která se aktualizuje po každé nově nahrané nebo vytvořené vrstvě. Nachází se v postranním panelu pod záložkou Legend. Aplikace přidá do záložky název vrstvy. K němu připojí HTML značku span, která dostává parametry jako výstupní vrstva v mapě.

```
document.getElementById('legend-div').innerHTML += "<h4>" + outputname + counter + "</h4><br>";  
document.getElementById('legend-div').innerHTML += "Symbol: <span style=\"width:50px;height:"+randomlineswidth+"pt; background-color:\" +  
randomlinescolor + "; opacity:\" + randomlinesopacity + "; display:inline-block;\">
```

Obrázek 13 - Kód, který přidá jméno vrstvy s kartografickým znakem do záložky Legend

Skrýtí postranního panelu a nastavení

Poslední funkcí je skrytí postranního panelu a základní možnost nastavení. Skrýt postranní panel lze křížkem v jeho pravé horní části a zobrazit jej uživatel může po kliknutí na šipku v levé části displeje. Kliknutím na křížek nebo šipku se volá funkce OpenNav, respektive CloseNav a mění atributy kaskádových stylů, například šířku postranního panelu z 0px (pixelu) na 340px. Zároveň se vedle křížku pro skrytí postranního panelu nachází ikona nastavení. Po kliknutí je zavolána funkce OpenSettings, která uprostřed displeje zobrazí div s možnostmi nastavení. První možností je editace symbologie před nahráním vrstvy do mapy. Pokud je tato možnost odškrtnuta, vrstva se nahraje s předem definovanou symbologií. Druhou možností nastavení je zobrazení atributů po kliknutí do mapového pole.

```
function OpenNav() {  
    document.getElementById("navbar").style.width = "340px";  
    document.getElementById("navbar").style.overflow = "scroll";  
    document.getElementById("navbar").style.padding = "15px";  
}  
  
function CloseNav() {  
    document.getElementById("navbar").style.width = "0";  
    document.getElementById("navbar").style.overflow = "hidden";  
    document.getElementById("navbar").style.padding = "0px";  
}
```

Obrázek 14 - Funkce OpenNav a CloseNav, které ovládají zobrazení a skrytí postranní lišty.

5.5 Ošetření uživatelského vstupu

Podkapitola se zaměřuje na ošetření uživatelských vstupů. Do aplikace je možné nahrát soubor pouze ve formátu GeoJSON. Pokud tomu tak není, zobrazí se uživateli chybová hláška ve vyskakovacím okně prohlížeče. Obdobně je tomu tak pokud uživatel nevybere žádný soubor a pokusí se kliknout na tlačítko UPLOAD.

Dalším šetřením uživatelského vstupu prošel výběr vrstev u operací. Pokud skript zjistí, že pro danou operaci není k dispozici žádná vhodná vrstva a uživatel se pokusí operaci zahájit, zobrazí se mu chybová hláška. V poslední řadě byl ošetřen vstup z hlediska nastavení symbologie. Barvu lze zvolit pouze na barevné paletě, průhlednost vrstvy je možné nastavit posuvníkem mezi hodnotami 0 a 1. Pokud je to nutné, jsou na vstupních parametrech nastaveny minimální a maximální povolené hodnoty, které lze zadat.

5.6 Dokončení a ladění chyb

Tato podkapitola pojednává o závěrečném ladění aplikace a odstraňování chyb. Autor po implementaci všech funkcí a operací musel odstranit gramatické chyby nebo upravit kaskádové styly tak, aby byla aplikace přehledná, dobře čitelná a upoutala uživatele.

V poslední řadě muselo být na závěr opraveno několik chyb, které se při používání aplikace objevovaly. Autor si je vědom toho, že i přes snahu odstranit veškeré chyby, se některá z chyb může objevit i v době odevzádní bakalářské práce.

6 TESTOVÁNÍ VÝKONNOSTI

V rámci této bakalářské práce bylo nutné otestovat implementovanou knihovnu Turf.js. Mapová aplikace byla uložena na serveru katedry geoinformatiky a testování probíhalo na třech odlišných počítačových sestavách s následujícími konfiguracemi:

Počítačová sestava č. 1

Processor: Intel® Core™ i5-4300U 1.9GHz (Turbo Boost 2.9GHz)

Operační paměť: 8GB

Disk: SSD

Grafická karta: dedikovaný grafický čip AMD Radeon HD 8750M

Internetové připojení: 20Mbit/s download, 5Mbit/s upload (WiFi)

Počítačová sestava č. 2

Processor: Intel® Core™ i5-7300HQ 2.5GHz (Turbo Boost 3.5GHz)

Operační paměť: 8GB

Disk: SSD

Grafická karta: dedikovaný grafický čip NVIDIA GeForce GTX 1050

Internetové připojení: 30Mbit/s download, 3Mbit/s upload (WiFi)

Počítačová sestava č. 3

Processor: Intel® Core™ i5-5500U 2.4GHz (Turbo Boost 3.0GHz)

Operační paměť: 8GB

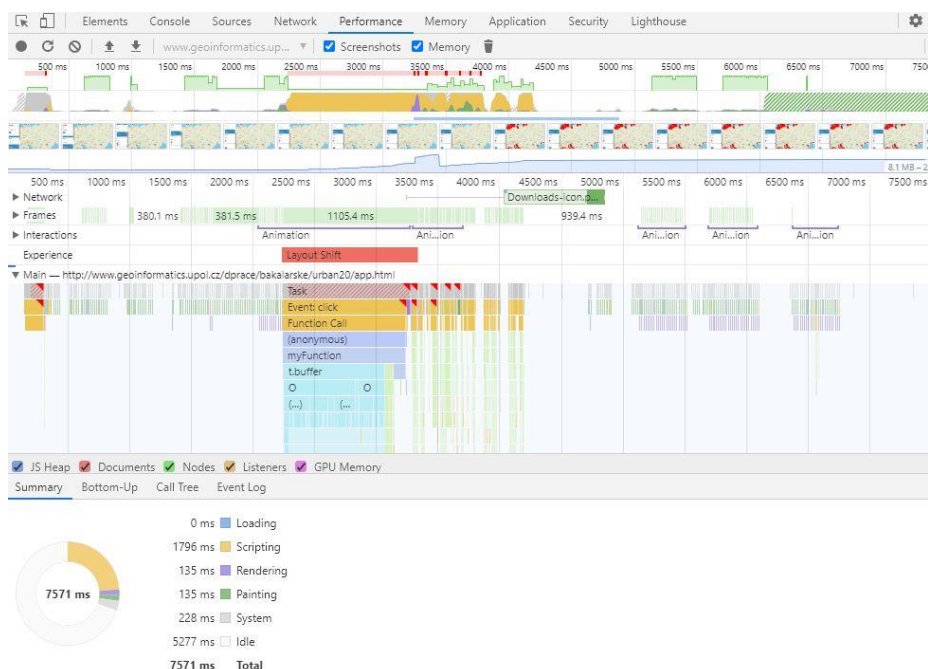
Disk: HDD

Grafická karta: integrovaný grafický čip Intel® HD Graphics 5500

Internetové připojení: 20Mbit/s download, 20Mbit/s upload (WiFi)

Na každé sestavě byla testována rychlost provedení tří operací a to Create random points (vytvoření náhodného počtu bodů), Buffer (obalová zóna) a TIN (triangulační síť). Během měření rychlosti byly využity webové prohlížeče Google Chrome ve verzi 81.0.4044.138, Mozilla Firefox ve verzi 76.0.1 a Microsoft Edge ve verzi 44.18362.449.0, které jsou jedny z nejpoužívanějších webových prohlížečů. Webový prohlížeč Safari od společnosti Apple byl zanedbnán, jelikož testování probíhalo pouze na operačním systému Microsoft Windows 10. [35]

Pro měření doby vykreslení byly použity integrované nástroje webových prohlížečů (záložka Performance). Tyto nástroje umožňují detailně sledovat průběh zpracování a vykreslení webové stránky. Testování probíhalo vždy při maximálním oddálení mapového pole, aby byla zajištěna srovnatelnost výsledků. Při přiblíženém mapovém poli na určitou oblast se totiž generovaná vrstva vykreslila zpravidla dříve než při pohledu na celý svět. Každá operace byla na jednotlivých sestavách provedena 10x. Z těchto hodnot byly vyloučeny nejmenší a největší naměřené hodnoty a vypočítán aritmetický průměr. Kompletní výsledky měření lze nalézt v příloze č. 3.



Obrázek 15 - Nástroje pro vývojáře v prostředí Google Chrome.

Google Chrome

Google Chrome je multiplatformní webový prohlížeč společnosti Google Inc. První verze byla vydána 2. září 2008 a prohlížeč je šířen pod licenci BSD. Původně Chrome používal renderovací jádro prohlížeče WebKit, následně však Google vytvořil alternativní verzi Blink, kterou prohlížeč využívá dodnes. Webový portál StatCounter odhaduje, že Google Chrome v květnu 2020 vlastní 63 % trhu, což z něj činí nejpoužívanější webový prohlížeč. [36][37][38]

Mozilla Firefox

Mozilla Firefox je multiplatformní webový prohlížeč, který vyvíjí společnost Mozilla Corporation. Je distribuován pod licenci MPL 2.0 a jeho první verze vyšla 23. září 2002. Prohlížeč používá renderovací jádro Gecko, které je vyvíjeno stejnojmennou společností. Gecko je nasazeno i do jiných prohlížečů jako například Waterfox, K-Meleon, Lunascape, Classilla, Conkeror nebo TenFourFox. [39][40]

Microsoft Edge

Microsoft Edge je webový prohlížeč vyvíjený společností Microsoft. První verze byla vydána 30. března 2015. Jedná se o nástupce prohlížeče Microsoft Explorer a v operačním systému Windows 10 se stal výchozím prohlížečem. Šířen je pod proprietární licenci EULA. V roce 2018 společnost oznámila, že Microsoft Edge přestane používat vlastní renderovací jádro EdgeHTML a stane se z něj prohlížeč na bázi projektu Chromium, na němž staví i Google Chrome nebo Opera. [41][42]

Funkčnost mapové aplikace byla mimo jiné ověřena i v dalších webových prohlížečích, a to Opera ve verzi 68.0.3618.125 a Internet Explorer ve verzi 11.836.18362.0.

Tabulka č. 1 - Výsledky měření operace Create random points ve webovém prohlížeči Google Chrome (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	10 000	50 000	100 000	250 000	500 000	1 000 000	2 000 000
PC č. 1	2814,4	4899,6	10650,1	32126,4	73417,3	N/A	N/A
PC č. 2	2461,5	4787,3	9041,6	25449,9	57605,0	N/A	N/A
PC č. 3	3058,1	8061,3	17338,9	40933,4	81088,0	N/A	N/A

Tabulka č. 2 - Výsledky měření operace Create random points ve webovém prohlížeči Mozilla Firefox (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	10 000	50 000	100 000	250 000	500 000	1 000 000	2 000 000
PC č. 1	1022,6	2179,1	3327,4	10145,1	16605,8	43595,3	N/A
PC č. 2	752,3	1900,1	4087,1	8050,9	11139,0	34097,9	N/A
PC č. 3	984,9	1705,8	3177,9	7689,8	17029,4	68137,0	N/A

Tabulka č. 3 - Výsledky měření operace Create random points ve webovém prohlížeči Microsoft Edge (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	10 000	50 000	100 000	250 000	500 000	1 000 000	2 000 000
PC č. 1	986,0	2558,6	4070,4	12610,3	29917,3	N/A	N/A
PC č. 2	1280,0	2697,4	4114,4	11100,6	26267,5	71055,5	N/A
PC č. 3	1506,5	2953,4	4210,0	15047,1	40132,3	N/A	N/A

Tabulka č. 4 - Výsledky měření operace Buffer ve webovém prohlížeči Google Chrome (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	1 000	2 000	5 000	10 000	25 000	50 000	100 000
PC č. 1	1380,0	1744,6	2261,5	4349,4	9425,0	15857,8	44969,9
PC č. 2	1233,5	1679,0	2511,3	3818,9	8063,8	17068,6	33311,4
PC č. 3	1590,1	1900,0	2408,6	7258,5	12443,3	23066,4	50842,6

Tabulka č. 5 - Výsledky měření operace Buffer ve webovém prohlížeči Mozilla Firefox (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	1 000	2 000	5 000	10 000	25 000	50 000	100 000
PC č. 1	1376,9	1680,6	2582,8	3988,6	8359,4	13996,9	28311,0
PC č. 2	1181,0	1327,1	2180,8	3467,5	7998,8	11870,1	24981,8
PC č. 3	1281,1	1404,4	3091,1	4670,1	7752,6	15140,4	30680,1

Tabulka č. 6 - Výsledky měření operace Buffer ve webovém prohlížeči Microsoft Edge (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	1 000	2 000	5 000	10 000	25 000	50 000	100 000
PC č. 1	2116,4	2327,8	2296,1	8293,8	18697,4	37004,5	N/A
PC č. 2	2001,1	2266,6	3714,9	7500,5	14837,0	28614,5	N/A
PC č. 3	2119,4	2343,3	4489,3	11037,9	22764,6	48172,1	N/A

Tabulka č. 7 - Výsledky měření operace TIN ve webovém prohlížeči Google Chrome (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	1 000	2 000	5 000	10 000	25 000	50 000	100 000
PC č. 1	788,4	952,3	1783,9	5890,5	7749,1	12750,3	27854,5
PC č. 2	777,0	1016,5	2971,3	6712,6	7504,0	12891,3	26810,1
PC č. 3	900,8	1092,4	2032,3	7060,1	10292,9	17969,6	34384,0

Tabulka č. 8 - Výsledky měření operace TIN ve webovém prohlížeči Mozilla Firefox (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	1 000	2 000	5 000	10 000	25 000	50 000	100 000
PC č. 1	393,5	715,6	1182,1	1684,4	3468,8	6641,6	12990,0
PC č. 2	387,0	671,3	936,6	1400,4	3017,0	5414,6	10542,4
PC č. 3	444,9	906,6	1480,6	2219,4	4260,4	8917,1	17653,6

Tabulka č. 9 - Výsledky měření operace TIN ve webovém prohlížeči Microsoft Edge (naměřené hodnoty jsou uvedeny v milisekundách a jsou zaokrouhleny na 1 desetinné místo).

Sestava	Počet bodů v datové sadě						
	1 000	2 000	5 000	10 000	25 000	50 000	100 000
PC č. 1	432,8	893,8	1484,3	2048,8	4891,5	10808,9	26751,8
PC č. 2	422,5	896,1	1481,9	1950,5	4809,8	8885,6	23027,1
PC č. 3	480,3	989,6	1698,4	2365,0	6145,6	15264,8	28150,4

Při měření rychlosti hrálo roli několik faktorů. Prvním je hardwarová konfigurace počítačových sestav, kde se jako nejrychlejší jevíly počítačové sestavy č. 1 a 2, nejpomalejší sestavou je č. 3. Jelikož knihovna Mapbox GL JS podporuje akceleraci grafických karet, může být příčinou absence dedikované grafické karty u sestavy č. 3, která má jen slabší integrovaný čip v procesoru. Jak již bylo zmíněno výše, dalším faktorem je například přiblížení mapového pole nebo další spuštěné aplikace v operačním systému. Z tohoto důvodu musely být webové prohlížeče nainstalovány až před testováním a neobsahovaly žádné pluginy ani jiná uživatelská přizpůsobení. Na rychlost má výrazný vliv i to, zda je aplikace GeONLINE uložena na serveru nebo na lokálním disku každé počítačové sestavy - na lokálním disku je provádění operací výrazně rychlejší.

Mezi testovanými webovými prohlížeči vynikal v rychlosti provedení operací prohlížeč Mozilla Firefox, který používá renderovací jádro Gecko. Například při operaci Create Random points s 500 000 generovaných bodů trvalo prohlížeči Google Chrome

vykonat operaci na všech sestavách více než 57 vteřin, v prohlížeči Mozilla Firefox 11 až 17 sekund a v Microsoft Edge se hodnoty pohybují od 26 do 40 sekund.

Operace Create random points nebyla dokončena v prohlížeči Google Chrome při generaci 1 000 000 a 2 000 000 bodů. Microsoft Edge zvládl provést operaci s 1 000 000 bodů na sestavě č. 2, Mozilla Firefox u všech tří sestav. U operace Buffer měl problém pouze prohlížeč Microsoft Edge při počtu 100 000 bodů v datové sadě. Všechny ostatní operace proběhly v pořádku, pouze někdy se stalo, že webový prohlížeč přestal na několik sekund odpovídat.

Na základě těchto měření tedy může autor práce doporučit webový prohlížeč Mozilla Firefox.

Dále autor testoval funkci, která umožní uživateli stáhnout vytvořenou vrstvu. Pro tyto účely byly operací Create random points vytvořeny vrstvy s 10 000, 50 000, 100 000, 250 000, 500 000 a 1 000 000 body. Měření probíhalo na počítačové sestavě č. 1.

Tabulka č. 10 - Výsledky měření velikosti exportovaného souboru ve formátu GeoJSON.

Formát	Velikost testovaného souboru					
	10 000	50 000	100 000	250 000	500 000	1 000 000
GeoJSON	1,1 MB	5,6 MB	11,1 MB	27,8 MB	55,6 MB	112,9 MB

7 SROVNÁNÍ S ALTERNATIVNÍMI ŘEŠENÍMI

V této kapitole je srovnána mapová aplikace GeONLINE, která implementuje knihovnu Turf.js s ostatními alternativními řešeními, které jsou popsány v kapitole 4 - ArcGIS Online, GIS Cloud a CARTO.

Tabulka č. 11 - Srovnání aplikace GeONLINE s alternativními řešeními.

	GeONLINE	ArcGIS Online	GIS Cloud	CARTO
Počet operací	14	29	4	15
Vstupní formáty	GeoJSON	SHP, GeoJSON, KML, GPX, CSV, ...	SHP, GeoJSON, KML, GPX, CSV, ...	SHP, GeoJSON, KML, GPX, CSV, ...
Legenda	ANO	ANO	ANO	ANO
Možnost zoomu	ANO	ANO	ANO	ANO
Geolokace	ANO	NE	ANO	NE
Volba podkladové mapy	NE	ANO	ANO	ANO
Export vrstev	ANO	ANO	ANO	ANO
Zobrazení atributů	ANO	ANO	ANO	ANO
Editace atributů	NE	ANO	ANO (pouze pro databáze)	ANO
Editace geometrie	NE	ANO	ANO	ANO
Vyhledávání	NE	ANO	ANO	ANO
Tisk mapy	NE	ANO	ANO	ANO
Měřítko	NE	ANO - grafické	ANO - číselné	NE
Uživatelské účty	NE	ANO	ANO	ANO

Pro srovnání bylo vybráno několik geoinformatických, kartografických a uživatelských aspektů. Nejdůležitějším aspektem je počet prostorových operací, které aplikace nabízejí. Aplikace GeONLINE podporuje 14 prostorových operací, srovnatelně s CARTO, která nabízí 15 operací. Nejméně operací nabízí platforma GIS Cloud, která navíc 2 operace nabízí pouze pro data z PostGIS databáze a nabízí tak velmi omezené možnosti prostorových operací. Naopak nejlépe s 29 operacemi si vede ArcGIS Online od ESRI.

Zřejmě největší slabinou aplikace GeONLINE je nabídka vstupních formátů - uživatel má možnost nahrát data pouze ve formátu GeoJSON. Ostatní aplikace nabízejí širší portfolio formátů od vektorových (SHP, GeoJSON, KML, GPX, ...), rastrových (TIF, JPG, PNG, ...) po webové služby (WMS, WMTS, ...). V tomto ohledu nemůže GeONLINE

konkurovat alternativním řešením od jiných společností. Všechny aplikace nabízí legendu, možnost přiblížení a oddálení mapového pole, export vrstev nebo zobrazení atributů jednotlivých prvků. Kromě GeONLINE nabízejí ostatní webové aplikace také editaci atributů a geometrie. Platforma GIS Cloud avšak nabízí editaci atributů pouze pro vrstvy, které byly nahrány jako databáze. Aplikace GeONLINE také na rozdíl od dalších produktů nenabízí volbu podkladové mapy, její tisk nebo vyhledávání. Společně s GIS Cloud naopak dokáže nabídnout funkci geolokace. Měřítko nabízí pouze služba ArcGIS Online a GIS Cloud. Prvně zmiňovaná má k dispozici grafické měřítko, druhá číselné měřítko.

GeONLINE je odlehčenou variantou pro alternativní řešení. Nabízí 14 prostorových operací, největším mínusem je omezené množství vstupních formátů a absence editace atributů a geometrie. Největším plusem této aplikace je její jednoduchost a rychlost. Aplikace jako jediná nenabízí možnost uživatelských účtů, což je způsobeno absencí databázového řešení. To může být považováno jak za klad pro jednoduchost aplikace, tak za zápor kvůli absenci ukládání dat. Veškerá další možná rozšíření pro aplikaci GeONLINE jsou popsána v diskuzi.

ArcGIS Online je nejkompaktnější řešení z uvedených možností. Nabízí největší počet prostorových operací a kromě absence geolokace nebo číselného měřítka má velkou podporu vstupních formátů a nabízí pokročilé možnosti kartografické vizualizace. Velkou výhodou je možné propojení s ekosystémem ESRI a s jeho dalšími produkty (ArcGIS Pro a další). Uživatelské prostředí je přívětivé a rychlé.

GIS Cloud nabízí nejomezenější výběr prostorových operací, a to jen 4. Navíc 2 z nich lze provést pouze se vstupní vrstvou v PostGIS databázi. Stejné omezení platí i u editace atributů jednotlivých prvků. Jinak se dá říct, že v ničem dalším nezaostává. Uživatelské prostředí GIS Cloud by si rozhodně zasloužilo redesign, působí zastarale a není příliš intuitivní.

CARTO se řadí se svými 15 prostorovými operacemi vedle aplikace GeONLINE. Nabízí velkou řadu vstupních formátů a kromě absence měřítka a geolokace nabízí všechny základní funkce. Uživatelské prostředí je přívětivé a vypadá moderně.

8 VÝSLEDKY

Tato kapitola popisuje výsledky, kterých bylo v této bakalářské práci dosaženo. Výsledky jsou rozděleny do tří částí. První část se zabývá popisem výsledků vývoje samotné aplikace a jak se podařilo naplnit zadané cíle. Ve druhé části se autor věnuje následnému testování rychlosti aplikace. V třetí části je prostor věnován srovnání vytvořené aplikace s alternativními možnostmi. Mimo to byla na začátku práce provedena rešerše dostupné literatury, která se tématu práce týkala. Autor v úvodu práce vymezil pojmy, které jsou v práci použity a zároveň prozkoumal stávající podobná řešení na trhu, které prostorové operace ve webovém prostředí nabízejí.

8.1 Výsledky vývoje aplikace

Hlavním cílem bakalářské práce byla implementace JavaScript knihovny Turf.js do komplexní webové aplikace. Pro tvorbu webové aplikace, která byla nazvána GeONLINE, byly použity technologie HTML, kaskádové styly a skriptovací jazyk JavaScript. Mimo knihovnu Turf.js byla taktéž použita knihovna Mapbox GL JS, která tvoří jádro aplikace a umožňuje vizualizovat data a interagovat s uživatelem. Pro implementaci bylo vybráno 14 operací, mezi kterými nechybí například tvorba obalových zón (Buffer), nepravidelné triangulační sítě (TIN), Voronoi polygonů nebo vytváření náhodných bodových, liniových či polygonových vrstev. Ke každé operaci je přidružená nabídka atributů, které upřesňují její provedení. Aplikace mimo jiné využívá podkladovou mapu od společnosti Mapbox a základní ovládací prvky z knihovny Mapbox GL JS. Uživateli je umožněno nahrávat vlastní soubory ve formátu GeoJSON.

Do aplikace byly mimo základní funkcionalitu prostorových operací navíc implementovány některé nadstavbové funkce. Mezi ně patří možnost stažení nově vytvořených vrstev ve formátu GeoJSON, zobrazení atributů jednotlivých prvků v mapovém poli, základní nastavení symbologie pro body, linie i polygony, přepínání vrstev nebo zobrazení legendy.

8.2 Výsledky testování

Dalším dílčím cílem bakalářské práce bylo testování vytvořené aplikace GeONLINE. Předmětem testování bylo změřit rychlost vykonání operací Create random points, Buffer a TIN. Testování probíhalo na 3 počítačových sestavách v následujících konfiguracích:

Počítačová sestava č. 1

Processor: Intel® Core™ i5-4300U 1.9GHz (Turbo Boost 2.9GHz)

Operační paměť: 8GB

Disk: SSD

Grafická karta: dedikovaný grafický čip AMD Radeon HD 8750M

Internetové připojení: 20Mbit/s download, 5Mbit/s upload (WiFi)

Počítačová sestava č. 2

Processor: Intel® Core™ i5-7300HQ 2.5GHz (Turbo Boost 3.5GHz)

Operační paměť: 8GB

Disk: SSD

Grafická karta: dedikovaný grafický čip NVIDIA GeForce GTX 1050

Internetové připojení: 30Mbit/s download, 3Mbit/s upload (WiFi)

Počítačová sestava č. 3

Processor: Intel® Core™ i5-5500U 2.4GHz (Turbo Boost 3.0GHz)

Operační paměť: 8GB

Disk: HDD

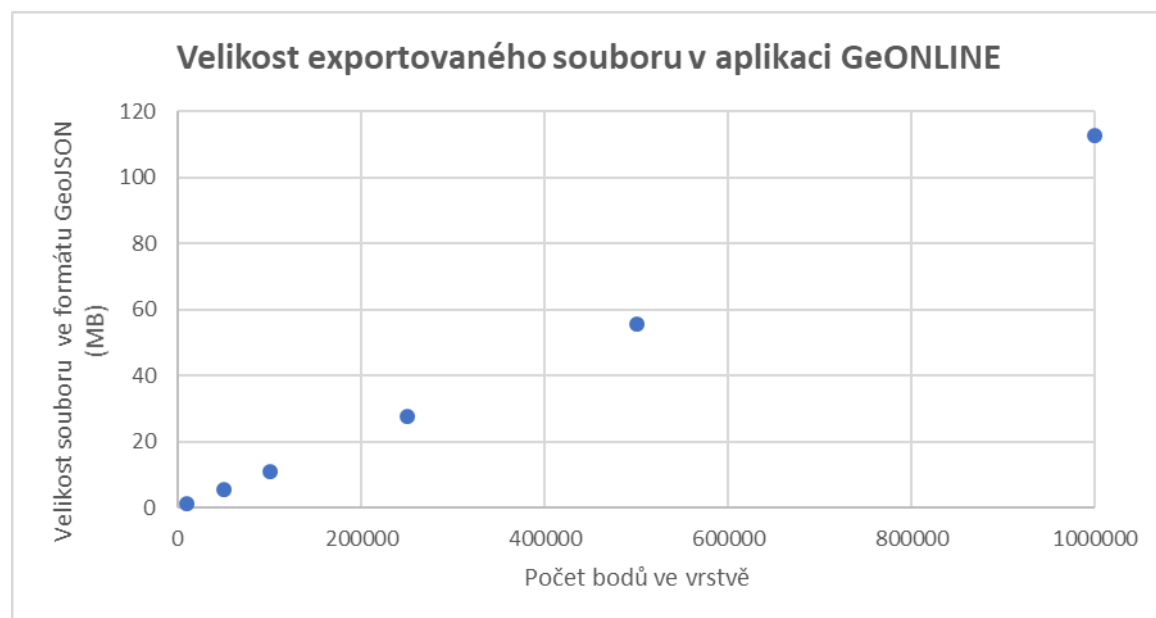
Grafická karta: integrovaný grafický čip Intel® HD Graphics 5500

Internetové připojení: 20Mbit/s download, 20Mbit/s upload (WiFi)

K testování byly použity webové prohlížeče Google Chrome, Mozilla Firefox a Microsoft Edge a jejich nástroje umožňující sledovat průběh zpracování a vykreslení webové stránky.

Jednotlivá měření probíhala na každé sestavě 10x, z těchto hodnot byly následně vyloučeny nejmenší a největší naměřené hodnoty a vypočítán aritmetický průměr. Z testovaných prohlížečů si s provedením operací vedl nejlépe Mozilla Firefox, kterému například operace TIN se 100 000 vstupními body trvalo na PC č. 1 12990 ms, na PC č. 2 10542,4 ms a PC č. 3 17653,6 ms, zatímco vykreslení v Google Chrome a Microsoft Edge trvalo ve všech případech více než 20 vteřin.

Na závěr byla testována funkce, která umožňuje uživateli stáhnout vytvořenou vrstvu ve formátu GeoJSON. Byla zvolena operace Create random points, pomocí které byly vytvořeny vrstvy s 10 000, 50 000, 100 000, 250 000, 500 000 a 1 000 000 body a následně staženy. Z grafu č. 1 lze zjistit, že velikost souboru rostla lineárně s přibývajícím počtem bodů ve vrstvě.



Graf č. 1 - Výsledky měření velikosti exportovaného souboru v aplikaci GeONLINE.

8.3 Výsledky srovnání s alternativními možnostmi

Na závěr bylo potřeba srovnat vytvořenou aplikaci GeONLINE porovnat s ostatními komerčními nástroji, které jsou popsány v kapitole 4. Pro účely srovnání vznikla tabulka č. 11, která porovnává několik vybraných geoinformatických, kartografických a uživatelských hledisek. Největším mínusem aplikace GeONLINE je nabídka vstupních formátů a absence jakéhokoliv uložení dat a přiřazení k uživatelskému účtu. Naopak největším kladem aplikace je její jednoduchost, rychlost, intuitivnost a rozmanitý výběr prostorových operací z knihovny Turf.js. Nejlépe si z pohledu počtu prostorových operací, vstupních formátů a doplňkových funkcí vedla aplikace ArcGIS Online.

Autorovi bude ctí, pokud se někdo rozhodne aplikaci dále rozšiřovat. Další možný rozvoj funkcionality je popsán v diskuzi.

9 DISKUZE

Cílem této bakalářské práce bylo prozkoumání možnosti využití procesních služeb ve webové kartografii, implementování knihovny Turf.js do mapové aplikace, její srovnání se stávajícími řešeními a testování.

Prvním problémem při implementaci bylo nahrávání vlastních souborů do aplikace. Autor původně zamýšlel nahrávání souborů do dočasné složky, která bude následně smazána po vypnutí stránky. Řešením nakonec bylo nahrání obsahu souboru do proměnné typu objekt, čímž se aplikace stala robustnější a nevytváří na serveru žádná další data ani složky. Veškerá data jsou uložena ve webovém prohlížeči na straně klienta.

Dalším problémem bylo ošetření uživatelských vstupů. Do aplikace nelze nahrát soubor v jiném formátu než GeoJSON nebo například nemůže spustit operaci pokud není zvolena žádná vhodná vrstva. Nepodařilo se však implementovat kontrolu vhodných atributů pro operaci TIN. Všechny atributy v proměnné objekt jsou totiž datového typu string (textové pole) a nebylo tedy možné ověřit zda se například jedná o číslo nebo text. Výběr atributu u operace TIN tak zůstává uživatelsky neošetřen a je možné vybrat kterýkoliv atribut.

Dalším problémem bylo vytvoření rozdílných obalových zón. Obalové zóny se jeví větší směrem k pólům a naopak se zmenšují směrem k rovníku. Vzniká tak zkreslení, které by mohlo uživatele ovlivnit v interpretaci výstupů. To je způsobeno implementací Mercatorova zobrazení. Taktéž se někdy při operaci buffer objevují dlouhé polygony v horizontální rovině v okolí pólů. Autor nedokázal zjistit, čím by to mohlo být způsobeno.

V rámci bakalářské práce byla implementována knihovna Turf.js do mapové aplikace ve webovém prostředí se základní funkcionalitou. Mapová aplikace ale samozřejmě nabízí mnoho možností k rozšíření do budoucna. Tím by mohlo být připojení aplikace k databázovému řešení. Vznikla by tím například možnost vytvoření uživatelských účtů a vytváření vlastních map, které by se ukládaly do databáze. Momentálně aplikace veškerá data ztratí současně s obnovením záložky ve webovém prohlížeči. Nepochybně je také ukládání geografických dat do databáze lepší řešení než ukládání dat do objektu v jazyku JavaScript. Toto rozšíření nebylo realizováno zejména kvůli náročnosti a rozsahu bakalářské práce. Velké možnosti do budoucna nabízí i implementace více operací. Aplikace podporuje 14 operací, zatímco knihovna Turf.js nabízí více než 100 operací k možné implementaci. Aplikace by zároveň mohla podporovat více vstupních formátů. Nyní podporuje pouze formáty GeoJSON. V rešerši se autor věnoval jiným řešením implementace, kde aplikace podporovaly více formátů (například KML, CSV, GPX, SHP, ...), databáze nebo vytváření vlastních vrstev pomocí kliknutí do mapového pole. Dalším rozšířením by mohla být širší možnost nastavení symbologie jednotlivých prvků. Aplikace by například mohla barevně odlišit prvky podle daného atributu. Významným rozšířením by byla možnost editace atributové tabulky. V současné podobě aplikace dokáže zobrazit atributy jednotlivých prvků. Editace atributů je důležitým prvkem správy geografických dat.

Jedním z cílů bylo testování vytvořené aplikace. K testování byly zvoleny 3 počítačové sestavy a webové prohlížeče Google Chrome, Mozilla Firefox a Microsoft Edge. Během testování se autor snažil o měření při stejných podmínkách, nicméně se obává, že to není možné. Vlivů, které ovlivní rychlost vykonání operace je mnoho - například stáří instalace operačního systému, běh dalších aplikací a podobně. Během

testování bylo zjištěno, že nejrychleji zpracuje operace prohlížeč Mozilla Firefox, což bylo pro autora práce překvapující. Autor očekával drobné rozdílné časy výpočtů. Nečekal však, že Mozilla Firefox mnohdy vykreslila vrstvu i 2x rychleji než Google Chrome nebo Microsoft Edge. Důvodem by mohlo být odlišné renderovací jádro, které může být pro knihovnu Mapbox GL JS a Turf.js lépe optimalizováno.

Implementace WPS bude mít podle autora v budoucnosti velké využití. Aplikace by při vhodném rozšíření funkcionality mohla při některých operacích postupně nahradit desktopový software. Tím by zmizela nutnost instalace software, jeho pravidelné aktualizace a podobně, což autor vidí jako obrovskou výhodu. K aplikaci GeONLINE je potřeba mít pouze nainstalovaný webový prohlížeč a být připojený k internetu.

10 ZÁVĚR

Hlavním cílem této bakalářské práce byla implementace JavaScript knihovny Turf.js a vytvoření mapové aplikace ve webovém prostředí. Implementaci předcházelo studium tématu, především ze zahraničních publikací, na jehož základě byly v teoretické části popsány použité technologie a současný stav dané problematiky.

Výsledkem je aplikace GeONLINE, která nabízí 14 prostorových operací z knihovny Turf.js. Aplikace využívá knihovnu Mapbox GL JS a nabízí několik funkcí jako export vrstvy ve formátu GeoJSON, zobrazení atributů jednotlivých prvků, legendu, ovládací prvky mapového pole nebo základní možnost nastavení kartografických znaků. Při implementaci byl použit značkovací jazyk HTML společně s kaskádovými styly a skriptovacím jazykem JavaScript.

Dalším cílem bylo aplikaci otestovat. Testování probíhalo na 3 počítačových sestavách ve webových prohlížečích Google Chrome, Mozilla Firefox a Microsoft Edge. Testovány byly 3 operace - Create random points, Buffer a TIN. Ze zmiňovaných webových prohlížečů byl ve většině měření nejrychlejší Mozilla Firefox s renderovacím jádrem Gecko, nejpomalejší naopak Google Chrome. Autor se zaměřil také na měření velikosti exportovaných souborů ve formátu GeoJSON, které obsahovaly od 10 000 do 1 000 000 bodových prvků. Bylo zjištěno, že na rychlost provedení operace má vliv několik aspektů - hardware počítačové sestavy, rychlost internetového připojení, počet zapnutých aplikací na pozadí počítače, přiblížení mapového pole nebo počet aktivních vrstev v mapě. Vliv mělo také to, zda je zdrojový kód umístěn na serveru nebo na lokálním disku počítače.

Posledním úkolem bylo porovnat aplikaci GeONLINE s alternativními řešeními, které jsou popsány v rešerši, a zhodnotit možný vývoj aplikace do budoucna se všemi možnými vylepšeními funkcionality.

V rámci bakalářské práce byl vytvořen poster formátu A2 a webové stránky, které shrnují cíle, metody, výsledky práce a závěr. Na webových stránkách je možno si stáhnout text práce ve formátu PDF a zmiňovaný poster. Webové stránky jsou vytvořeny v souladu s pravidly dostupnými na stránkách katedry.

POUŽITÁ LITERATURA A INFORMAČNÍ ZDROJE

- [0] - NĚTEK, R. Frekvence využívání mapových metod na mapových portálech. Bakalářská práce. Univerzita Palackého v Olomouci, Přírodovědecká fakulta, Katedra geoinformatiky. Olomouc, 2008.
- [1] - KRAAK, M.-J., BROWN, A. (Eds.) *Web Cartography: Developments and Prospects*. Taylor & Francis Group, 2001. 213 s. ISBN 9780748408696
- [2] - GIBB, Robert. What is a Web Application. *Stackpath* [online]. 2016 [cit. 2020-06-04]. Dostupné z: <https://blog.stackpath.com/web-application/>
- [3] - What is a Software Library. *Techopedia* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.techopedia.com/definition/3828/software-library>
- [4] - OGC. *GISMentors* [online]. 2019 [cit. 2020-06-04]. Dostupné z: <https://training.gismentors.eu/open-source-gis/standardy/ogc/index.html>
- [5] - NĚTEK, Rostislav a Tomáš BURIAN. *Free and open source v geoinformaticce*. Olomouc: Univerzita Palackého, 2018. ISBN 978-80-244-5291-3.
- [6] - WPS Interface. In: *Landmap Geoknowledge* [online]. 2020 [cit. 2020-06-04]. Dostupné z: http://learningzone.rspoc.org.uk/images/stories/learning_materials/OGC/Images/Unit_3/WPS-Interface.png
- [7] - OpenGIS® Web Processing Service. *Open Geospatial Consortium Inc.* [online]. 2007 [cit. 2020-06-04]. Dostupné z: http://portal.opengeospatial.org/files/?artifact_id=24151
- [8] - Webové služby. *Help Service - Remote Sensing* [online]. 2017 [cit. 2020-06-04]. Dostupné z: <http://www.bnhelp.cz/produkty/webove-sluzby/>
- [9] - What is HTML (Hypertext Markup Language). *Computer Hope* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.computerhope.com/jargon/h/html.htm>
- [10] - CSS: Cascading Style Sheets. *MDN* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web/CSS>
- [11] - Kaskádové styly. *Adaptic* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.adaptic.cz/znalosti/slovnicek/kaskadove-styly/>
- [12] - W3C. *Adaptic* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.adaptic.cz/znalosti/slovnicek/w3c/>

- [13] - SLÁDEK, Jan. Webdesignérův průvodce po CSS3: nultá kapitola. *Zdroják* [online]. 2010 [cit. 2020-06-04]. Dostupné z: <https://www.zdrojak.cz/clanky/webdesigneruv-pruvodce-po-css3-nulta-kapitola/>
- [14] - JavaScript. *MDN* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- [15] - JavaScript. *Adaptic* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.adaptic.cz/znalosti/slovnicek/javascript/>
- [16] - DHTML | Introduction. *GeeksforGeeks* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.geeksforgeeks.org/dhtml-introduction/>
- [17] - Mapbox pricing. *Mapbox* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.mapbox.com/pricing/>
- [18] - Mapbox GL JS. *GitHub* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://github.com/mapbox/mapbox-gl-js>
- [19] - Mapbox GL JS. *Mapbox* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://docs.mapbox.com/mapbox-gl-js/api/>
- [20] - *Turf.js* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://turfjs.org/>
- [21] - Spatial analysis. *Mapbox* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://docs.mapbox.com/help/how-mapbox-works/geospatial-analysis/>
- [22] - Contributors to Turfjs/turf. *GitHub* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://github.com/Turfjs/turf/graphs/contributors>
- [23] - *GeoJSON* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://geojson.org/>
- [24] - RFC 7946 - The GeoJSON Format. *Internet Engineering Task Force* [online]. 2016 [cit. 2020-06-04]. Dostupné z: <https://tools.ietf.org/html/rfc7946>
- [25] - ArcGIS Online. *ARCDATA PRAHA* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.arcdata.cz/produkty/arcgis/webovy-gis/arcgis-online>
- [26] - ArcGIS Online - GIS Wiki. *ESRI* [online]. 2020 [cit. 2020-06-04]. Dostupné z: http://wiki.gis.com/wiki/index.php/ArcGIS_Online#Supported_Browsers
- [27] - GIS Cloud [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.giscloud.com/>
- [28] - *CARTO* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://carto.com/>

- [29] - PyWPS. *OSGeo* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.osgeo.org/projects/pywps/>
- [30] - Welcome to the PyWPS 4.3.dev0 documentation. *PyWPS* [online]. 2016 [cit. 2020-06-04]. Dostupné z: <https://pywps.readthedocs.io/en/latest/>
- [31] - Simplify.js - a high-performance JavaScript 2D/3D polyline simplification library. *GitHub* [online]. 2013 [cit. 2020-06-04]. Dostupné z: <http://mourner.github.io/simplify-js/>
- [32] - Turf/simplify. *GitHub* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://github.com/Turfjs/turf/tree/master/packages/turf-simplify>
- [33] - What is a TIN surface. *ESRI* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://desktop.arcgis.com/en/arcmap/10.3/manage-data/tin/fundamentals-of-tin-surfaces.htm>
- [34] - ArcGIS Desktop Help 9.2 - Voronoi polygons. *ESRI* [online]. 2007 [cit. 2020-06-04]. Dostupné z: http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=Voronoi_maps
- [35] - Browser Statistics. *W3Schools* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://www.w3schools.com/browsers/>
- [36] - Webový prohlížeč Google Chrome. *Google* [online]. 2020 [cit. 2020-06-04]. Dostupné z: https://www.google.com/intl/cs_CZ/chrome/
- [37] - Google going its own way, forking WebKit rendering engine. *Arstechnica* [online]. 2013 [cit. 2020-06-04]. Dostupné z: <https://arstechnica.com/information-technology/2013/04/google-going-its-own-way-forking-webkit-rendering-engine/>
- [38] - Browser Market Share Worldwide. *Statcounter* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://gs.statcounter.com/browser-market-share>
- [39] - Firefox. *Computer Hope* [online]. 2018 [cit. 2020-06-04]. Dostupné z: <https://www.computerhope.com/jargon/f/firefox.htm>
- [40] - Gecko FAQ. *MDN* [online]. 2020 [cit. 2020-06-04]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Gecko/FAQ>
- [41] - JO FOLEY, Mary. Google gains power over web as Microsoft rebuilds Edge browser on Chrome tech. *ZDNet* [online]. 2019 [cit. 2020-06-04]. Dostupné z: <https://www.cnet.com/news/microsoft-rebuilds-edge-browser-on-chrome-tech-google-gains-web-power/>

[42] - SHANKLAND, Stephen. Google gains power over web as Microsoft rebuilds Edge browser on Chrome tech. *Cnet* [online]. 2018 [cit. 2020-06-04]. Dostupné z: <https://www.cnet.com/news/microsoft-rebuilds-edge-browser-on-chrome-tech-google-gains-web-power/>

PŘÍLOHY

SEZNAM PŘÍLOH

Volné přílohy

Příloha 1 Poster

Příloha 2 CD

Elektronické přílohy

Příloha 3 Výsledky testování výkonnosti

Popis struktury CD

Urban20_textBP - text bakalářské práce

GeONLINE - adresář s aplikací GeONLINE

Web - adresář s webovými stránkami k práci

Vysledky_mereni - tabulka s kompletními výsledky měření výkonnosti

Poster