



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ

DEPARTMENT OF INFORMATION SYSTEMS

**WEBOVÁ APLIKACE PRO VIZUALIZACI HODNOT
V GEOGRAFICKÝCH SOUŘADNICÍCH**

WEB APPLICATION FOR VISUALIZATION OF VALUES IN GEOGRAPHICAL COORDINATES

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

PETR KAŠPAR

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. JIŘÍ HYNEK, Ph.D.

BRNO 2020

Zadání bakalářské práce



Student: **Kašpar Petr**
Program: Informační technologie
Název: **Webová aplikace pro vizualizaci hodnot v geografických souřadnicích**
Web Application for Visualization of Values in Geographical Coordinates
Kategorie: Uživatelská rozhraní

Zadání:

1. Prozkoumejte možnosti reprezentace a vizualizace geografických dat reprezentující hodnoty v konkrétních geografických souřadnicích (např. teplotní mapa, bublinová mapa, *spike* mapa, apod.). Prostudujte technologie pro tvorbu webových uživatelských rozhraní a vizualizací geografických dat (Leaflet, D3.js, apod.).
2. Prozkoumejte existující aplikaci určenou pro vizualizaci generických geografických dat. Prostudujte princip provázání dat s danou aplikací.
3. Analyzujte požadavky uživatelů na snadnou tvorbu mapových diagramů prezentující hodnoty v konkrétních geografických souřadnicích.
4. Navrhněte rozšíření aplikace z bodu 2 řešící požadavky z bodu 3.
5. Navržené rozšíření implementujte.
6. Výsledné řešení otestujte na vhodných datech. Zhodnoťte výsledky.

Literatura:

- Kachlík, J.: Grafická vizualizace geografických dat síťového provozu. Brno, 2020. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií.
- Johnson, J.: *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines*. Morgan Kaufmann Publishers/Elsevier, 2010, ISBN: 978-0-12-375030-3.
- Harris, R. L.: *Information Graphics: A Comprehensive Illustrated Reference*. Oxford University Press, 2000. ISBN: 978-0-1951-3532-9.
- Leaflet: *Leaflet API reference* [online]. 2019 [cit. 2020-09-16]. Dostupné z: <https://leafletjs.com/reference-1.7.1.html>

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 4

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Hynek Jiří, Ing., Ph.D.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2020
Datum odevzdání: 12. května 2021
Datum schválení: 23. října 2020

Abstrakt

Vizualizace dat v mapách je technika zobrazení dat reprezentujících veličiny s vazbou na prostorové prvky. Tato práce si klade za cíl vytvořit webovou aplikaci umožňující vizualizovat data v konkrétních geografických souřadnicích a poskytnout tak podklad k analýze veličiny v mnoha různých úrovních zobrazení od obecného přehledu až po jemné detaily. Knihovna Geovisto umožňuje vizualizaci generických dat a jejich vazeb, nicméně tyto vizualizace jsou územně vymezené dle států a nepracují s konkrétními souřadnicemi. To je pro některé druhy analýzy nevhodné a zároveň to znemožňuje vizualizace dat, které pracují s konkrétními geografickými souřadnicemi. Zaměřil jsem se na rozšíření této knihovny o mapové vrstvy schopné vizualizovat data v souřadnicích.

Abstract

Data visualization in maps is a technique of data presentation, which displays spatially-bound variables. The objective of this thesis is to create a web application capable of creating visualizations of data in geographic coordinates, enabling user to analyze observed phenomenon in many different levels of view, ranging from the general overview of patterns to the analysis of fine details. The Geovisto library is capable of visualization of generic data and their connections but these visualizations are limited to the territories of states and do not work with exact geographic coordinates. This is inconvenient for some kinds of data analysis and makes visualizations of data sets that work with exact coordinates impossible. I focused on extending this library by map layers capable of visualizing data in exact geographic coordinates.

Klíčová slova

datové mapy, geovizualizace, metoda teček, teplotní mapa, mapa proporciálních symbolů, bubble mapa, spike mapa

Keywords

data maps, geovisualization, dot map, heat map, proportional symbols map, bubble map, spike map

Citace

KAŠPAR, Petr. *Webová aplikace pro vizualizaci hodnot v geografických souřadnicích*. Brno, 2020. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Jiří Hynek, Ph.D.

Webová aplikace pro vizualizaci hodnot v geografických souřadnicích

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Jiřího Hynka, Ph.D. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Petr Kašpar
10. května 2021

Poděkování

Rád bych poděkoval svému vedoucímu Ing. Jiřímu Hynkovi, Ph.D. za velmi vstřícný přístup při konzultacích, poskytnutí studijní literatury a motivaci při vypracovávání této práce. Dále bych rád poděkoval panu Tomáši Bradáčovi a panu Janu Šulcovi za konzultace při tvorbě barevných gradientů určených pro uživatele se ztrátou barvocitu.

Obsah

1	Úvod	3
2	Reprezentace a vizualizace geografických dat	5
2.1	Data	5
2.1.1	Kvalitativní a kvantitativní data	5
2.1.2	Třídní intervaly	6
2.1.3	Dimenze dat	7
2.1.4	GeoJSON	8
2.2	Vizualizace geografických dat	9
2.2.1	Univariantní a multivariantní mapy	10
2.2.2	Kartogram	11
2.2.3	Connection map	11
2.3	Tematické mapy pro zobrazování v geografických souřadnicích	11
2.3.1	Dot density map	11
2.3.2	Metoda proporciálních symbolů	11
2.3.3	Spike map	13
2.3.4	Isogram map	13
2.3.5	Teplotní mapa	14
2.4	Měřítko a shlukování symbolů	15
2.5	Hustota a zkreslení dat	16
2.5.1	Hustota dat	17
2.5.2	Data ink	17
3	Nástroje pro vizualizaci geografických dat	18
3.1	Uživatelská rozhraní v prostředí webových aplikací	18
3.1.1	Technologie pro tvorbu webových rozhraní	18
3.2	Nástroje pro tvorbu geografických vizualizací	19
3.2.1	Autorské systémy	20
3.2.2	Programové řešení a knihovny	21
4	Knihovna Geovisto	23
4.1	Uživatelské rozhraní a mapové vrstvy	23
4.2	Architektura	24
4.3	Datový model	25
5	Analýza	26
5.1	Cílová skupina	26
5.2	Analýza vstupních dat	26

5.3	Výstup analýzy	27
6	Návrh řešení	28
6.1	Vstupní data	28
6.2	Zvolené tematické mapy	29
6.3	Uživatelské rozhraní	30
6.4	Mapové vrstvy	31
7	Implementace	37
7.1	Použité technologie	37
7.2	Rozšíření stávající architektury	37
7.3	Implementace mapových vrstev	38
7.3.1	Přiřazování barev dle kategorií	40
7.4	Dot map	40
7.5	Spike map	41
7.6	Bubble map	42
7.7	Heat map	44
8	Testování	46
8.1	Testování vstupních dat	46
8.2	Uživatelské testování	47
8.2.1	Testování základní funkcionality	47
8.2.2	Testování gradientů barvoslepými uživateli	47
9	Závěr	49
	Literatura	50

Kapitola 1

Úvod

Data jsou často nějakou formou vázána na určité místo. Obzvláště při velké diverzitě nebo hustotě je analýza dat zobrazených pomocí geografické vizualizace daleko efektivnější než při použití obyčejné tabulky či statistických výpočtů. Při správném použití geovizualizačních technik je možné data na první pohled odlišit dle zobrazované veličiny a její intenzity. Takto zobrazená data slouží nejen k prohlížení a porovnávání výsledků, ale také odhalují intenzitu v různých stupních detailu analýzy a tím umožňují nalezení možných neočekávaných výskytů hodnot a usnadňují nalezení příčiny hledaného jevu. Data vizualizovaná tímto způsobem se dnes používají v mnoha odvětvích jako je meteorologie, doprava nebo v různých politických a sociologických průzkumech.

I přes to, že je geografická vizualizace hojně využívána, možnosti tvorby takovýchto vizualizací jsou omezené. Existující nástroje se povětšinou omezují na modifikaci předem definovaných šablon. Pokud se přeci jen jedná o pokročilé autorské nástroje, jsou běžně zpoplatněné a nenabízejí přílišné možnosti ohledně dekompozice a využití v aplikacích třetích stran. Naprostá většina uživatelů zároveň nedisponuje programátorskými a matematickými dovednostmi vyžadovanými pro tvorbu vlastních geografických vizualizací za pomoci pro tento účel vytvořených knihoven.

Aplikace Geovisto si klade za cíl vytvořit kompromis umožňující vytvářet vizualizace uživatelům pomocí webového uživatelského rozhraní a zároveň poskytující platformu pro vývojáře, kteří chtějí tento produkt využít, modifikovat či rozšířit pro vlastní potřeby. Vizualizace tohoto nástroje je nicméně omezena na administrativně vymezené celky dle států. Existující nástroj je vytvořen jako webová aplikace v JavaScriptu s použitím knihoven React, D3.js a Leaflet. Data jsou zapsána ve formátu JSON a je možné si zvolit zobrazované hodnoty pomocí poskytnutí jména klíče daného záznamu. Data jsou zobrazována ve vrstvách, kde každá vrstva je nezávislá na ostatních a je možné zobrazená data filtrovat.

Cílem této práce je definovat, popsat a analyzovat existující geovizualizační techniky pro zobrazení dat vymezených územím i přesně danými geografickými souřadnicemi a následně implementovat zobrazení v geografických souřadnicích a aplikaci Geovisto o tuto funkcionalitu rozšířit. Pro tento účel bude nutné vytvořit nový datový model umožňující zadání přesných geografických souřadnic místo identifikátorů států a vytvořit novou nezávislou vrstvu pro každý druh prezentace dat. Výsledkem bude webová aplikace, která zobrazuje data v geografických souřadnicích pomocí *dot map*, *spike map*, *bubble map* a *teplotní mapy*, kde každý způsob zobrazení je vhodný pro jiný druh analýzy a dat.

V kapitole 2 se budu zabývat možnostmi reprezentace dat v konkrétních geografických souřadnicích, pravidly tvorby takovýchto zobrazení a jejich popisem a analýzou. Kapitola 3 je věnována popisu nástrojů použitelných pro tvorbu webových uživatelských rozhraní

a vizualizací geografických dat. V kapitole 4 se zaměřím na popis existující aplikace a na princip provázání dat s aplikací, která je zobrazuje. Kapitola 5 se bude zabývat analýzou dat určených k vizualizaci a analýzou požadavků na snadnou tvorbu mapových diagramů, jednoduchost konfigurace a snadnost používání. V kapitole 6 se zaměřím na popis návrhu rozšíření pro existující aplikaci z kapitoly 4. V kapitolách 7 a 8 se budu věnovat implementačním detailům navrženého rozšíření, respektive testování rozšíření na vhodných datech. V závěru diskutuji dosažené výsledky a možná další rozšíření práce.

Kapitola 2

Reprezentace a vizualizace geografických dat

Dle [22] je grafické zobrazení dat způsob umožňující zobrazení velkého množství dat na malém prostoru tak, aby bylo možné je snadno porovnávat a pozorovatel tak získal co nejvíce vědomostí za co nejkratší čas. Geovizualizace je grafickým zobrazením dat na jakékoli podkladové mapě, tohoto zobrazení je dosaženo kombinací kartografických a statistických technik.

V úvodu této kapitoly se budu věnovat popisu dat, jejich klasifikaci a možnostmi zobrazení. Dále je nutné vysvětlit využití a druhy map, přičemž důraz bude kladen na podmnožinu map zvanou tematická mapa. Ve další části se budu zabývat způsoby vhodnými pro zobrazování v konkrétních geografických souřadnicích. Protože existuje velké množství možných způsobů zobrazení a jejich popis by překročil uvedený rozsah práce, zaměřím se pouze na techniky implementované v této práci. V závěru této kapitoly popíši pravidla tvorby statistických grafů zajišťující maximalizaci výpovědní hodnoty výsledného zobrazení.

2.1 Data

Vhodná vizualizační technika se liší podle povahy vizualizovaných dat. Před samotnou vizualizací je tedy nutné nejprve správně pochopit charakter daných dat. Data lze obecně chápat jako hodnoty různých datových typů schopných přenosu, jež reprezentují nějakou skutečnost. Vhodnou vizualizací dat je možné koncovému uživateli poskytnout podklad pro snadné získání informace – tedy dat, která mají interpretovatelný význam.

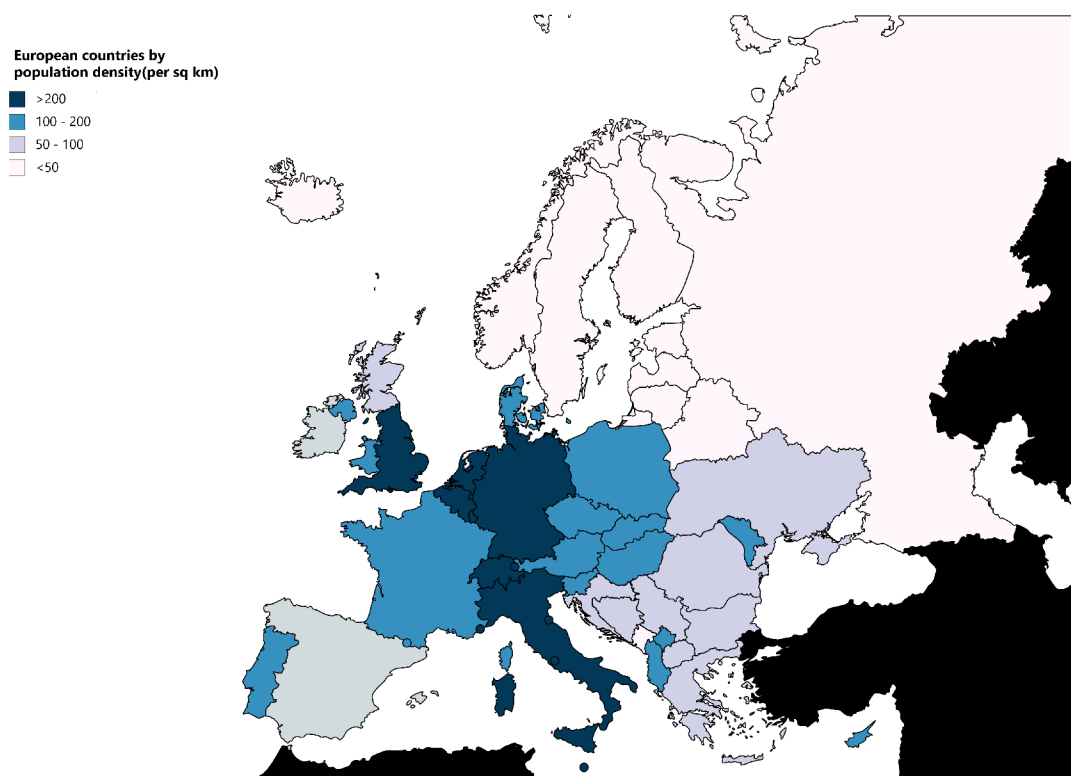
2.1.1 Kvalitativní a kvantitativní data

Data lze dělit dle dvou metrik: kvalitativní a kvantitativní data [1]. Kvalitativní data jsou taková, která popisují vlastnosti nějakých jevů. Jsou typicky vyjádřena jako výčet konečného množství kategorických hodnot. Mohou například popisovat spokojenost zákazníků s nějakou službou. Dělí se dále na data nominální, tedy data která lze uspořádat, a data ordinální, u kterých to není možné. Kvalitativní data je možné pozorovat, nebo odvodit od dat kvantitativních. V případě geovizualizace je vhodným příkladem kvalitativních dat například zobrazení subjektivního pocitu bezpečí v jednotlivých státech (v takovém případě jsou data mapována jako administrativně vymezený polygon), případně v jednotlivých domácnostech (vazba na geografické souřadnicke konkrétní domácnosti).

Kvantitativní data jsou oproti tomu definována jako data měřitelná, vyjádřená číslem, které určuje jejich intenzitu. Oproti datům kvalitativním je možné je řadit podle velikosti, či z nich například vypočítat střední hodnotu. Tento typ dat může nabývat jak diskretních, tak spojitéch hodnot. Při datech, jejichž alespoň jedna dimenze je geografická lokalita lze spojitá data chápat jako průměrnou teplotu v daném bodě, případně jako průměrnou teplotu celého státu v určitém ročním období. Diskrétní data si lze naopak představit jako počet obyvatel vizualizovaný buď jako kartogram pro celý stát, nebo pomocí *dot density map*, která zobrazí obyvatele dle nahlášené adresy.

2.1.2 Třídní intervaly

Harris píše [10], že pokud je velký soubor dat systematicky rozdělen na omezený počet skupin, nazývá se tato skupina třídní interval. To je pro efektivní vizualizaci dat klíčové. Například při zobrazování počtu obyvatel států v EU za použití kartogramu, který barevně odlišuje státy dle intenzity dané hodnoty by bylo nutné definovat rozdílnou barvu pro každý stát. Za použití třídních intervalů je možné tato data rozdělit na skupiny, kde každá obsahuje interval například dvaceti milionů a každá další je oproti předchozí posunuta o právě dvacetimilionový rozestup. Využití třídních intervalů je zobrazena na následujícím obrázku 2.1.

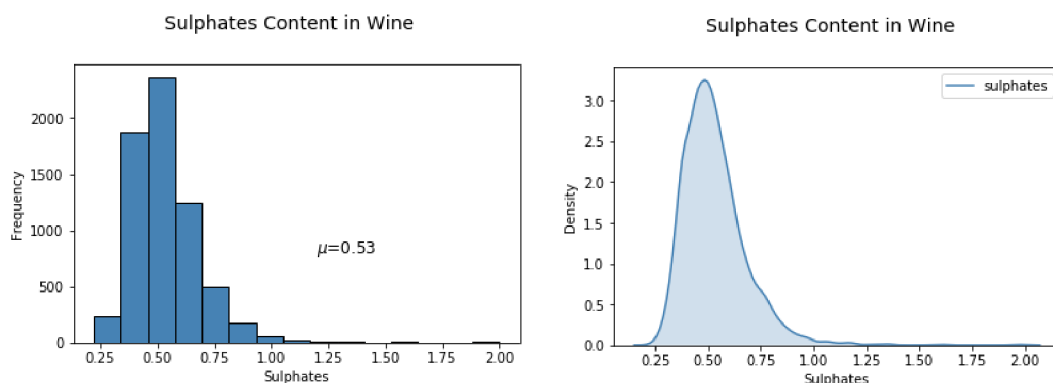


Obrázek 2.1: Hustota osídlení evropských zemí na kartogramu používajícím třídní intervaly.¹

¹Zdroj: <https://commons.wikimedia.org/w/index.php?curid=74179858>

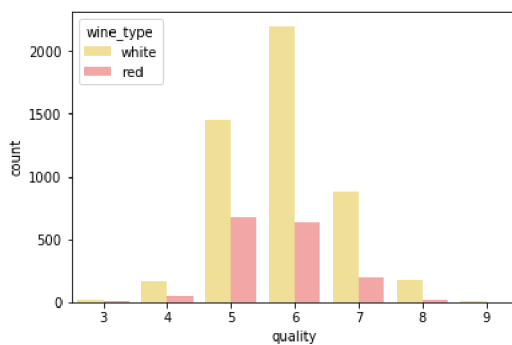
2.1.3 Dimenze dat

Dimenze dat je množina hodnot nějakého typu popisující kvantitativní data nebo kvalitativní data. Pokud mají data více než jednu dimenzi, hovoříme o multidimenzionálních datech. Nad jednotlivými dimenzemi je možné provádět operace jako je agregace, filtrování a kategorizace. Dle počtu a typu dimenzí dat se liší vhodné vizualizační médium. Jak ukazuje [19] na obrázku 2.2, pro data jednodimenzionální je vhodný histogram, kde výška každého sloupce určuje četnost právě jednoho atributu sledované veličiny.



Obrázek 2.2: Zobrazení jednodimenzionálních dat v histogramu

S nárůstem počtu dimenzí dochází k nárůstu komplexity grafu. Nejjednodušší možnost zobrazení dat se dvěma dimenzemi je složit dva sloupcové grafy, kde každý atribut dat bude odlišen jinou barvou, což je vyobrazeno na obrázku 2.3.

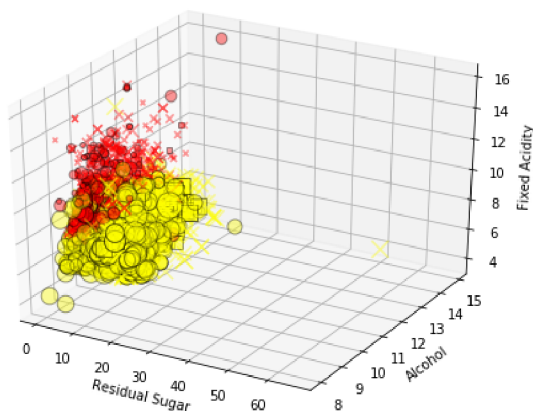


Obrázek 2.3: Složení dvou sloupcových grafů pro zobrazení dvou dimenzí dat. První dimenze je kvantitativní a je vyjádřena výškou sloupce. Druhá kategorická dimenze je zobrazena barvou sloupce.

Další možností (v závislosti na povaze jednotlivých dimenzí) může být zobrazovat pouze jeden sloupec, jehož výška určuje intenzitu veličiny jedné, a míra saturace použité barvy intenzitu veličiny druhé. Často používanou technikou je vizualizovat vícerozměrná data v prostoru – to ovšem přímo nesouvisí s počtem dimenzí dat a například trojrozměrná data mohou být zobrazena v dvourozměrném poli, pokud bude vhodným způsobem vyjádřena i třetí dimenze dat. Na obrázku 2.4 jsou šestidimenzionální data zobrazena ve trojrozměr-

ném prostoru. Tři dimenze představuje prostorové umístění dle třech os, zbylé tři dimenze sděluje použitý tvar, barva a velikost.

Wine Residual Sugar - Alcohol Content - Acidity - Total Sulfur Dioxide - Type - Quality



Obrázek 2.4: Šestidimenzionální data ve trojrozměrném prostoru. Tento obrázek vyobrazuje obsah zbytkového cukru ve víně, obsah alkoholu, kyselost, celkové množství oxidu siřičitého, typ vína a kvalitu.²

Při vizualizaci geografických dat v datových mapách se jedná vždy o data multidimenzionální. Při vizualizaci v kartogramu je jednou dimenzí identifikátor polygonu, ke kterému se vizualizovaná data vztahují. Pokud uvažujeme konkrétní geografické souřadnice, jsou zde pro umístění veličiny do mapy dimenze dokonce dvě – zeměpisná šířka a zeměpisná délka.

2.1.4 GeoJSON

Geografická data je možné zadávat přímo pomocí zeměpisné šířky a délky, nebo nepřímo pomocí reference na identifikátor, kde tento identifikátor představuje nějaký geografický objekt. Může se jednat o bod, spojení bodů, polygon a podobně. Takovéto objekty je potom nutné nějakým způsobem definovat – je možné využít vlastní formát, nebo nějaký obecný standard používaný pro tento účel.

GeoJSON je otevřený standardní formát pro zobrazení jednoduchých prostorových dat spolu s dalšími neprostorovými atributy. Je založen na formátu JavaScript Object Notation. GeoJSON text je složen z jediného GeoJSON objektu, který musí být jedním z GeoJSON typů: `Feature`, `FeatureCollection`³, `GeometryCollection`, `Point`, `Polygon`, `LineString`, `Multipoint`, `MultiPolygon` a `MultiLineString`. S výjimkou GeoJSON typů `Feature` a `FeatureCollection`, které jsou označovány jako *Feature Objects*, se jedná o takzvané *Geometry Objects*⁴.

Geometrické objekty reprezentují body, křivky a polygony. Takové objekty musí mít vlastnost udávající `type`, který musí mít hodnotu identifikující některý z geometrických objektů. Dále musí mít vlastnost `coordinates`, která určuje umístění objektu v prostoru. Hodnota této vlastností se liší dle použitého objektu. Například geometrický objekt typu

²Obrázky 2.2, 2.3 a 2.4 byly převzaty z článku [19]

³V případě `FeatureCollection` a `GeometryCollection` slovo *Collection* nemá žádný sémantický význam, jedná se vždy o uspořádaná pole, nikoliv o neuspořádanou kolekci.

⁴Slovní spojení *Geometry Objects* a *Feature Objects* budou pro účely této práce dále překládány do češtiny jako *geometrické objekty*, respektive *rysové objekty*.

`Point` bude mít pouze jeden údaj který vyjadřuje jeho polohu, naproti tomu objekt typu `LineString` bude mít pole pozic. Pozice jako taková je pole čísel mající alespoň dva prvky, tedy zeměpisnou délku a šířku. Navíc je možné přidat volitelný třetí prvek značící nadmořskou výšku.

Rysové objekty reprezentují prostorově vázaná data. Rysový objekt může obsahovat čtyři vlastnosti: `type`, jehož hodnota musí být rovna řetězci `Feature`, `geometry`, která by měla být některým z geometrických objektů, případně může být rovna JSON hodnotě `null`, dále vlastnost `properties`, která je reprezentována jako JSON objekt a obsahuje neprostorová data vázaná na výše zmíněný geometrický objekt, a volitelnou vlastnost `id`, která představuje jednoznačný identifikátor daného rysového objektu.

GeoJSON objekty `FeatureCollection` a `GeometryCollection` jsou takové objekty, které mají pouze jednu vlastnost, a to vlastnost `features`, respektive `geometries`. V obou případech se jedná o JSON pole obsahující v prvním případě rysové objekty, ve druhém případě objekty geometrické.

Jednoduchý příklad zápisu GeoJSON je na obrázku 2.5. Více o formátu se lze dočíst ve specifikaci [2].

```
1 {
2   "type": "FeatureCollection",
3   "features": [
4     {
5       "type": "Feature",
6       "properties": {
7         "kraj": "Jihocesky"
8       },
9       "geometry": {
10        "type": "Point",
11        "coordinates": [
12          14.326171874999998,
13          49.439556958940855
14        ]
15      }
16    }
17  ]
18 }
```

Obrázek 2.5: `FeatureCollection` ve vlastnosti `features` obsahuje jeden objekt typu `Feature`, který v `properties` obsahuje data vázaná na objekt `geometry`.

2.2 Vizualizace geografických dat

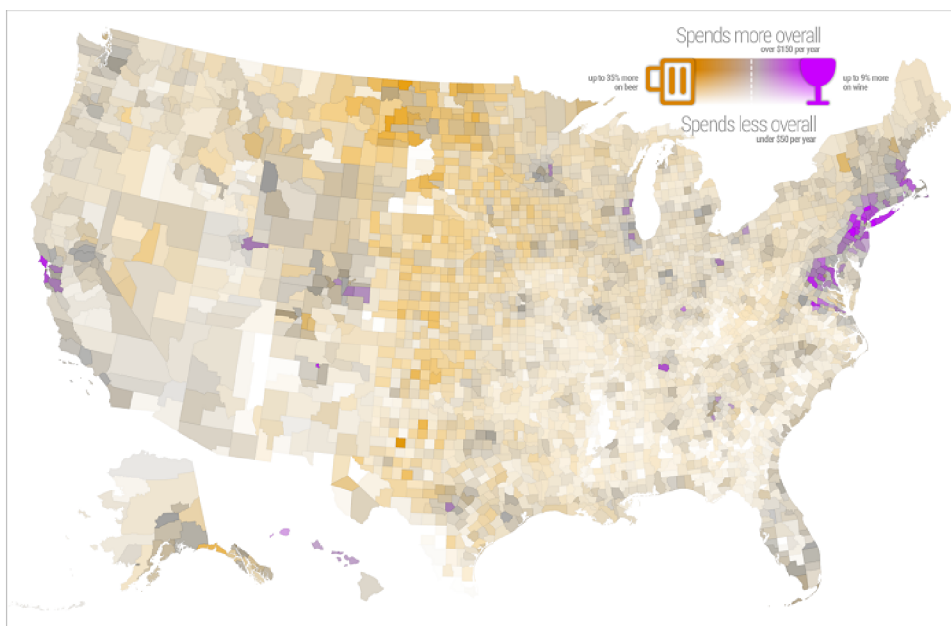
Pro rychlé získání hodnoty dat je vhodná tabulka. Pokud chceme tato data provnávat navzájem, je ovšem vhodnější data zobrazit pomocí diagramu. V případě, že je nutné zjistit, jak jsou data distribuována geograficky, neexistuje jiná možnost, než využít mapu. Mapy lze dle Roberta L. Harrise [10] rozdělit na šest hlavních druhů: mapy statistické, deskriptivní, topografické, *flow* mapy, mapy vyjadřující počasí a mapy pro speciální účely – například

mapa znázorňující trasy autobusových linek. Pro účely této práce se budeme zabývat téměř výhradně statistickými mapami.

Tematická mapa, v literatuře též nazývaná statistická mapa [10], případně datová mapa [22], je takový druh mapy, který zobrazuje data jejichž alespoň jeden atribut udává geografickou lokaci. Dle Throwera [21] je rozdíl mezi mapou běžně využívanou například v dopravě a tematickou mapou ten, že tematická mapa využívá zobrazené hranice a lokace pouze jako referenční místa sloužící pro orientaci, nejedná se o hlavní náplň mapy jako takové. Tematická mapa tedy obsahuje topografický podklad – daný obecně reliéfem krajiny, lidskými sídly a politickými či administrativními hranicemi – a tematický obsah. Topografický obsah mapy nazývaný podkladová mapa, v angličtině *base map*, se mění v závislosti na požadavích tematické vrstvy. Obecným pravidlem je udržovat zobrazené informace v podkladové mapě na nutném minimu tak, aby nedocházelo k odvádění pozornosti od zobrazených statistických dat.

2.2.1 Univariantní a multivariantní mapy

Univariantní mapy jsou mapy takové, které zobrazují v mapě pouze jednu veličinu. Oproti tomu multivariantní mapy zobrazují několik veličin na jednom místě. Jedná se tedy o mapy zobrazující data s jednou, respektive několika dimenzemi. Multivariantní zobrazení je vhodné ve chvílích, kdy je nutné zobrazit takové datové dimenze, které mezi sebou mají nějaký vztah a usnadní tak jejich zkoumání. Při vytváření takového zobrazení je nicméně nutné dbát na to, aby poskytnutím velkého množství informace na jednom místě nebyla snížena čitelnost. V případě velkého množství dimenzí tedy může být efektivnější jednotlivé dimenze zobrazit v několika samostatných univariantních mapách.



Obrázek 2.6: Zobrazení kategorické dimenze, která představuje vybraný alkoholický nápoj spolu s kvantitativní dimenzí reprezentující průměrnou utrácenou částku v jednotlivých okresech USA pomocí multivariantní mapy.⁵

2.2.2 Kartogram

Kartogram, v angličtině *choropleth map* [5], je taková tematická mapa, která zobrazuje data agregovaná na oblasti reprezentující určitý územní celek. Takovéto oblasti mohou být odlišeny například barvou, mírou saturace jedné barvy, stínováním či šrafováním. Datová hodnota je typicky relativní a je vztažena k ploše vybrané oblasti, může jít například o hustotu osídlení. Lze se ovšem setkat i s kartogramy, kde zobrazené údaje nejsou plošně vztaženy v poměru k územnímu celku, ale jedná se pouze o zobrazení hodnoty naměřené v dané oblasti. Příkladem může být průměrný věk obyvatel v jednotlivých krajích ČR. Kartogram je zobrazen na obrázcích 2.1 a 2.6.

2.2.3 Connection map

Anglický termín *connection map* představuje mapu spojení. Jedná se o mapu která představuje spojení mezi několika body na mapě. Takováto mapa se dá rozdělit na dva druhy: *route map* a *flow map*. *Route map* zobrazuje přesnou cestu mezi dvěma (nebo více) body, nicméně neposkytuje téměř žádné, nebo vůbec žádné informace o tom, co se mezi těmito body přesouvá. Oproti tomu *flow map* nezobrazuje přesnou cestu, ale zaměřuje se na předmět přesunu, přesouvané množství a jeho zdroj a cíl [10].

2.3 Tematické mapy pro zobrazování v geografických souřadnicích

Pro určité druhy analýzy není vhodné využívat mapy, které údaje agregují na územní celky. Proto existují statistické mapy pro zobrazování v přesných geografických souřadnicích.

2.3.1 Dot density map

Dot density map, známá také jako *dot distribution map*, do češtiny překládaná jako metoda teček, je taková mapa, kde jednotlivé body znázorňují výskyt určité proměnné. Každý bod (tečka) může znázorňovat jediný výskyt sledované veličiny, ale i několik výskytů [10]. Informaci o tom, jakou intenzitu jedna tečka reprezentuje je tedy vhodné předat v podobě legendy. Tato mapa může zobrazovat jak kvantitativní, tak kvalitativní data. Užitečné je barevné odlišení, které může znamenat například různé kategorie kvalitativních dat, nebo třídní intervaly intenzity kvantitativních dat.

2.3.2 Metoda proporciálních symbolů

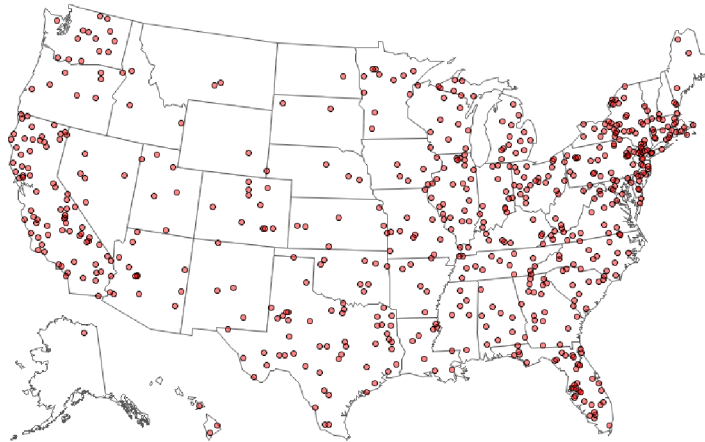
Jedná se o typ tematické mapy, která zobrazuje symboly, jejichž velikost se mění v závislosti na atributu, který reprezentují. Nejčastěji používaným symbolem je kruh, taková mapa bývá potom označována jako *bubble map*. Ačkoliv nejčastěji bývá tento typ mapy používán pro zobrazování v souřadnicích, je možné takto zobrazovat i hodnotu agregovanou v polygonu. Symboly mohou vyjadřovat buď reálnou hodnotu – v takovém případě se jedná o tzv. absolutní škálování, případně odstupňované hodnoty třídních intervalů.

Odstupňované hodnoty jsou pro čtenáře mapy jednodušší na porovnání, nicméně ztrácí se při jejich použití přesná hodnota. Oproti tomu při použití absolutního škálování bývá

⁵Zdroj: <https://gistbok.ucgis.org/bok-topics/multivariate-mapping>

⁶Zdroj: http://robslink.com/SAS/democd59/us_population_density.htm

2010 U.S. Population Dot Density map (each dot represents 500,000 people)

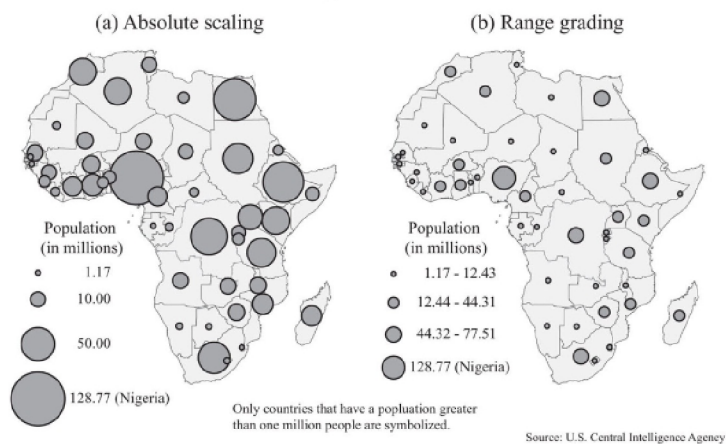


Obrázek 2.7: Metoda teček zobrazující populaci USA v roce 2010. Každá tečka symbolizuje 500 000 lidí.⁶

čtenářem často špatně odhadnuta míra změny intenzity zobrazovaného atributu vzhledem k velikosti symbolu. Při použití kruhu je vnímáný růst plochy symbolu pomalejší, než reálný naměřený růst [22]. To způsobuje, že určit, zda je jeden symbol dvakrát větší než druhý je pro většinu čtenářů velice obtížné [3].

Symbols lze škálovat i podle jiných rozměrů, než je jejich plocha. U kruhu je možné například použít růst dle průměru. Nevýhodou je, že při škálování dle průměru dochází k rychlejšímu růstu plochy kruhu. To je spojené s obecným problémem tohoto typu zobrazení, tedy že symboly se mohou navzájem překrývat, což zhoršuje výslednou čitelnost.

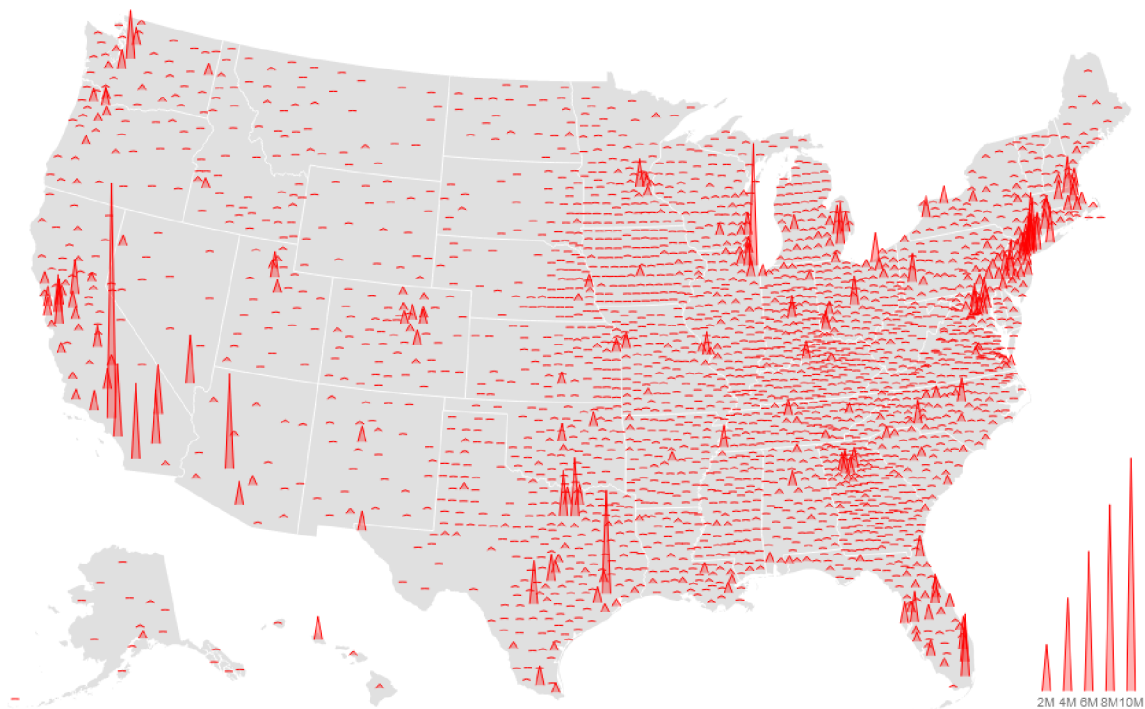
Africa Population, 2005



Obrázek 2.8: Srovnání absolutního škálování (vlevo) a odstupňovaných hodnot (vpravo) u proporciálních symbolů.⁷

2.3.3 Spike map

Jedná se o mapu využívající proporciální symboly [5] ve formě bodců, jejichž výška vyjadřuje intenzitu sledovaného atributu. To na jednu stranu usnadňuje pozorování a porovnávání intenzit v různých místech výskytu, nicméně stejně jako u metody proporciálních znaků využívající kruhy může docházet k překrývání jednotlivých bodců. Oproti té ovšem usnadňuje určení přesné geografické lokace i při vyšších intenzitách zobrazených atributů, podobně jako metoda teček. Spike map je zobrazena na obrázku 2.9.



Obrázek 2.9: Odhadovaná populace v jednotlivých okresech USA, 2016⁸

2.3.4 Isogram map

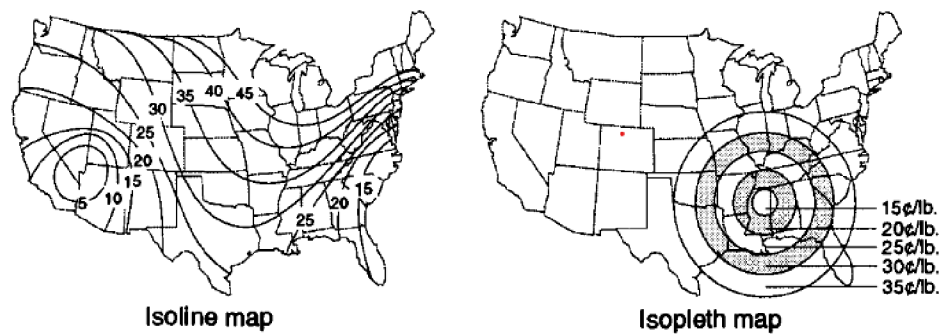
Termín *isogram map* popisuje mapu, která zobrazuje linie nebo pásma podél kterých mají všechny body stejnou hodnotu [10]. V případě, že se jedná o linie, je takováto mapa označována jako metoda isolinií. Pravá izolinie se používá pro zobrazení kvantitativních spojitých dat, tedy primárně pro znázorňování přírodních jevů, jako je teplota, nebo tlak vzduchu. Jednou z nejčastěji používaných izolinií je vrstevnice, která spojuje na mapě body se stejnou nadmořskou výškou. Pokud jsou zobrazená data nespojitá, lze hovořit o nepravé izolonii.

Tematická mapa, která zobrazuje pásma stejných hodnot se nazývá isopleth. Typicky takováto pásma reprezentují jednotlivé třídní intervaly, které bývají odlišeny barvou, nebo mírou barevné saturace. Vzhledem se tato mapa podobá kartogramu, ale vizualizovaná data nejsou nanášena do předdefinovaných polygonů, nýbrž do oblastí, které jsou součástí vstupních dat [18].

⁷Převzato z [3]

⁸Zdroj: <https://observablehq.com/@d3/spike-map>

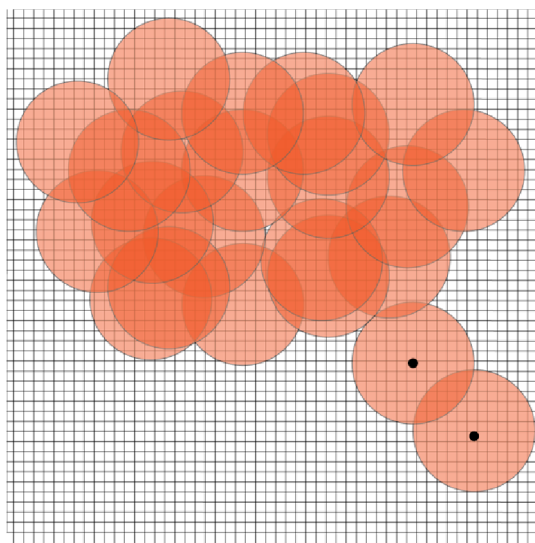
⁹Převzato z [10]



Obrázek 2.10: Srovnání isolinií a isoplethu.⁹

2.3.5 Teplotní mapa

Teplotní mapa je dalším druhem tematické mapy, kterou lze chápat jako podmnožinu isoplethu. Nicméně, místo přechodů mezi jednotlivými pásmy, které reprezentují třídní intervaly, jsou hodnoty zobrazeny barvou určitého barevného spektra, která je zvolena pomocí interpolace hustoty bodů. Toto spektrum může být vyjádřeno buď paletou barev, nebo mírou saturace jedné barvy. Teplotní mapy zobrazují hustotu bodů v rastrovém prostoru – pro všechny body je zvolena vzdálenost, která je následně použita jako poloměr pro vykreslení kruhu kolem každého bodu. S vyšším počtem překrývajících se kruhů roste hustota a takové místo je na mapě intenzivněji vybarveno. Tento postup je zobrazen na obrázku 2.11.

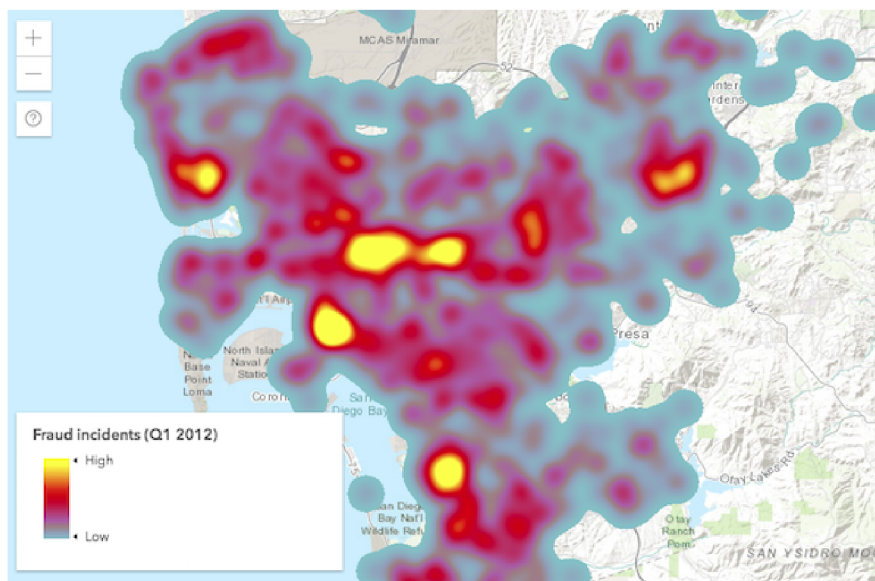


Obrázek 2.11: Ukázka překrývání jednotlivých kruhů, na základě kterých se určuje výsledná intenzita.¹⁰

Hlavním problémem této metody je, že při nesprávně zvolené velikosti poloměru může docházet k vyhodnocení vysoké hustoty i v místech, kde ve skutečnosti žádné body nejsou. Jak píše Mike DeBoer [4], řešením může být (kromě vhodně zvoleného poloměru) na vý-

¹⁰Převzato z [4]

sledné mapě zobrazovat i body, ze kterých byla mapa vytvořena. U interaktivních map je typické, že poloměr vlivu každého bodu se mění na základě míry přiblížení.



Obrázek 2.12: Každý pixel na této teplotní mapě má intenzitu určenou jeho blízkostí k jednomu nebo více vstupním bodům.¹¹

2.4 Měřítko a shlukování symbolů

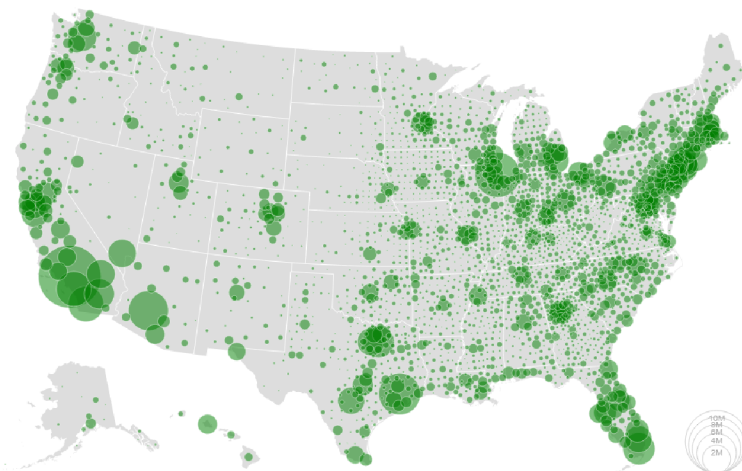
Měřítko u symbolů vizualizujících data na podkladových mapách je důležité pro přehlednost a rychlé chápání zobrazených dat. Obzvláště při použití interaktivních map je třeba s měřítkem pracovat velmi opatrně. Protože v případě webových geografických vizualizací se velikost symbolů běžně zadává jako počet pixelů, je při přiblížení nebo oddálení mapy vhodné zobrazované symboly zvětšovat, respektive zmenšovat tak, aby nezakrývaly příliš velké množství podkladové mapy a zároveň byly čitelné i při velkém přiblížení [7]. Stejně tak u teplotní mapy je vhodné měnit v závislosti na míře přiblížení velikost poloměru pro vyhodnocení intenzity v daném místě. Druhou možností je využít shlukování spolu se změnou hodnoty, jaké hodnoty zobrazená značka nabývá.

U map využívajících proporciální symboly, které typicky zobrazují veličiny kvantitativní je shlukování jednou z možností, jak se vyhnout nežádoucímu překrývání symbolů a s tím spojenou ztrátou čitelnosti. Nicméně, u map tohoto typu je nutné počítat s tím, že značka vzniklá sloučením několika menších značek reprezentujících kvantitativní data se musí proporciálně zvětšit s ohledem na to, jaké velikosti nabyla zobrazená veličina po sloučení z více zdrojů. Pokud by se vhodně neupravila velikost značky reprezentující shluk, došlo by k výraznému zkreslení zobrazených dat.

Kromě shlukování je možné symboly nechat překryté a pouze je vykreslovat průhledné. Tento přístup je pro vizualizaci v konkrétních souřadnicích obzvláště výhodný, neboť na-

¹¹Zdroj: <https://developers.arcgis.com/javascript/latest/api-reference/esri-renderers-HeatmapRenderer.htm>

rozdíl od shlukování neskrývá geografický rozměr zasažených značek. Způsob překryvání implementovaný pomocí knihovny D3.js je zobrazen na obrázku 2.13.



Obrázek 2.13: Ukázka bubble mapy, která místo shlukování využívá překryvání s průhledností symbolů.¹²

2.5 Hustota a zkreslení dat

Zobrazení není zkreslené, pokud vizuální zobrazení je konzistentní s jeho numerickou reprezentací. Vizuální zobrazení může znamenat jak fyzicky měřenou plochu, tak čtenářem vnímaný vizuální efekt. Nicméně vnímání je závislé na konkrétním čtenáři, jeho zkušenostech, jeho cíli a také na kontextu [16]. Je proto vhodné používat takové zobrazení, u kterého u většiny čtenářů nedochází k dezinterpretaci (příklad možné dezinterpretace je uveden v sekci 2.3.2). Edward Tufte ve své práci [22] zmiňuje dva důležité principy pro minimalizaci zkreslení, a tedy dodržení grafické integrity u vizuální reprezentace:

- Reprezentace čísel fyzicky měřená na ploše obrazu by měla být přímo proporční k reprezentovaným numerickým kvantitám.
- Jasně, detailní a důsledné popisování by mělo být použito k odstranění grafického zkreslení a nejasností, například napsáním vysvětlivek k datům přímo do grafiky.

Míru porušení prvního principu indikuje tzv. *Lie Factor*. Ten je možné spočítat pomocí vzorce:

$$\text{Lie Factor} = \frac{\alpha}{\beta}$$

kde α je rovna *velikosti efektu zobrazeného v grafice* a β je rovna *velikosti efektu v datech*. Pokud je výsledný Lie Factor roven jedné, data nejsou nijak zkreslena. V praxi je často přeháněna velikost efektu zobrazeného v grafice a tak Lie Factor rovný dvěma či pěti není nijak výjimečný.

¹²Převzato z: <https://observablehq.com/@d3/bubble-map?collection=@observablehq/maps>.

2.5.1 Hustota dat

Lidské oko je schopné zaregistrovat pozoruhodný počet rozdílů na malém prostoru. V kartografii je proto hlavním jmenovatelem stručnost spolu s vysokým rozlišením. Pokud podělíme počet vstupů v datové matici plochou datového grafu, získáme údaj zvaný *datová hustota* [22]. Velice málo statistických grafů dosahuje datové hustotě map, mapa je médiem nesoucím velké množství informací na relativně malém prostoru. Pokud je vstupních dat příliš velké množství, lze použít redukcující techniky, jako je zprůměrování, vyhlazování nebo shlukování. Případně lze smršťovat měřítko – například menší tečky při použití metody teček, nebo jemnější barevné přechody u teplotních map. Kromě zmenšení symbolů na mapě je také možné zvýšit intenzitu atributu, který symbol reprezentuje. To je ukázáno na obrázku 2.7.

2.5.2 Data ink

Pojem *data ink*, do češtiny přeložitelný jako datový inkoust¹³, je nesmazatelné jádro grafiky, které se mění spolu s daty [22]. Souvisejícím termínem je tzv. Data-Ink Ratio, tedy poměr použitého datového inkoustu ku inkoustu použitému k vykreslení celé grafiky. Při tvorbě grafů by měla být snaha tento poměr maximalizovat, tedy většina zobrazené plochy by měla reprezentovat data. Platí tedy, že grafický element by měl hlavně zobrazovat data, nicméně může plnit i grafickou funkci. Je ovšem nevhodné upřednostňovat dekorativní funkci před výpovědní hodnotou.

¹³Slovo *inkoust* v tomto případě znamená jakýkoliv prostředek použitý k vykreslení grafiky, například se tedy může jednat o pixely na počítačovém displeji.

Kapitola 3

Nástroje pro vizualizaci geografických dat

Uživatelské rozhraní lze chápat jako část systému, prostřednictvím které lidé komunikují se systémem samotným. Člověk je ve své komunikaci omezen svými smysly, je tedy nutné rozhraní na straně stroje upravit tak, aby bylo člověku umožněno komunikovat co nejpřirozeněji. Příkladem uživatelského prostředí může být volant spolu s pedály, řadicí pákou a tachometrem u osobního vozu, nebo klávesnice u počítače. Protože je tato práce zaměřena na uživatelská rozhraní v prostředí webu, budu se dále zabývat výhradně tímto druhem uživatelského rozhraní. První část této kapitoly je věnována popisu webových uživatelských rozhraní a technologiemi pro jejich tvorbu. Ve druhé části kapitoly se budu zabývat rozbohem existujících nástrojů pro tvorbu geografických vizualizací.

3.1 Uživatelská rozhraní v prostředí webových aplikací

Velké množství aplikací pro vizualizaci dat je mimo desktopové aplikace dostupné v podobě webových aplikací. Webová aplikace je takový druh softwaru, který běží na webovém serveru. K takovéto aplikaci je nutné přistupovat typicky pomocí webového prohlížeče, nebo (méně běžně) jako k nativním aplikacím na mobilním zařízení. Uživatelské rozhraní slouží jako prostředek pro komunikaci mezi uživatelem a webovou aplikací. Pro tvorbu uživatelských rozhraní je možné využít standardů jako HTML či CSS, programovacích jazyků jako je JavaScript či využít možností některých z knihoven nebo frameworků přímo zaměřených na tvorbu uživatelských rozhraní.

3.1.1 Technologie pro tvorbu webových rozhraní

HTML a CSS

Základním stavebním kamenem každé webové stránky je dokument ve značkovacím jazyku Hypertext Markup Language. Tento jazyk je obecně známý pod zkratkou HTML a je vyvíjen dle standardu konsorcia W3C a skupiny WHATWG [15]. Pomocí vnořených elementů tvoří HTML dokument stromovou strukturu. Stromová reprezentace HTML, která umožňuje přístup k jednotlivým uzlům (přičemž dokument samotný je považován za uzel) se nazývá *Document Object Model*. Jazyk HTML neobstarává vizuální prezentaci obsahu, ta je definována odděleně pomocí takzvaných kaskádových stylů (*Cascading Style Sheets*), neboli CSS [17].

Dříve bylo možné vkládat do webových stránek grafiku pouze formou statických obrázků, případně pro animace využít některý z pluginů prohlížeče – například Adobe Flash Player. Ten byl posléze nahrazen modernějšími technologiemi, kterými disponovalo HTML5 – SVG¹ a Canvas. SVG je vektorový grafický formát založený na XML, který je nezávislý na rozlišení [14]. Poskytuje také SVG DOM API, pomocí kterého je možné grafiky takto vykreslené dynamicky modifikovat. Oproti tomu Canvas je JavaScriptové API umožňující kreslit programově. Jedná se o bitmapovou grafiku, výkon je tedy závislý na rozměrech obrázku, nikoliv složitosti. SVG podporuje v základu animace, Canvas podobného efektu dosahuje vykreslováním za použití časovačů.

JavaScript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk vyvíjený podle specifikace [6]. Velkou výhodou využití jazyka JavaScript je z hlediska uživatelské přívětivosti možnost dynamické interakce s dokumentem pomocí reakcí na události. Události lze chápat jako vnější okolnosti, které mohou nastat činností prohlížeče nebo uživatele. Lze tedy například zareagovat na to, že se HTML dokument načel a je možné do něj přidat nějaká data, nebo přehrát animaci. Z hlediska uživatelských rozhraní je nicméně zajímavá hlavně možnost reagovat na uživatele. Každá událost má jméno, například `onClick` pro kliknutí myši na určitý element, nebo `onLoad`, která nastává při výše zmíněném načtení dokumentu. Každý element může mít následně atribut se jménem konkrétní události a hodnotou tohoto atributu by měl být kód v JavaScriptu, který se vykoná, jakmile nastane daná událost pro konkrétní element. Dalším možným způsobem je vytvořit tzv. *event listener*, který má přidělený konkrétní element a událost a na tu následně reaguje provedením svého kódu.

Využití jazyka JavaScript umožňuje reagovat na činnost uživatele v reálném čase bez nutnosti požadavek zaslat na server, znovu načíst stránku a zobrazit obdrženou odpověď. Místo toho je možné na události reagovat přímo v prohlížeči, případně zaslat na server asynchronní dotaz a dle obdrženého výsledku přizpůsobit zobrazenou stránku, opět bez nutnosti kompletního překreslení. Pomocí jazyka JavaScript lze vytvořit takové webové aplikace, které po úvodním načtení nepotřebují připojení k internetu a jejich kód se po celou dobu běhu programu provádí výlučně v prohlížeči (například různé hry v prohlížeči často fungují na tomto principu) [8].

3.2 Nástroje pro tvorbu geografických vizualizací

Různé aplikace podporující práci s mapami, jako Google Maps² nebo Waze³, jsou využívány denně mnoha uživateli po celém světě. Nicméně, tento typ aplikací je často omezen pouze na plánování trasy a umístování bodů do mapy, jsou velice omezeny v umístování (a zobrazování) statistických rozměrů umožňujících analýzu zobrazených dat. Pokud je třeba provádět analýzu metody teček, je možné takovéto mapy použít, nicméně není to jejich hlavní případ užití. V případě, že je ke geografické lokaci přidána další dimenze reprezentovaná nějakou kvantitativní veličinou, není možné tento typ map využít a je třeba využít nástrojů pro tento účel vytvořených.

¹Scalable Vector Graphics <https://developer.mozilla.org/en-US/docs/Web/SVG>

²<https://www.google.cz/maps>

³<https://www.waze.com>

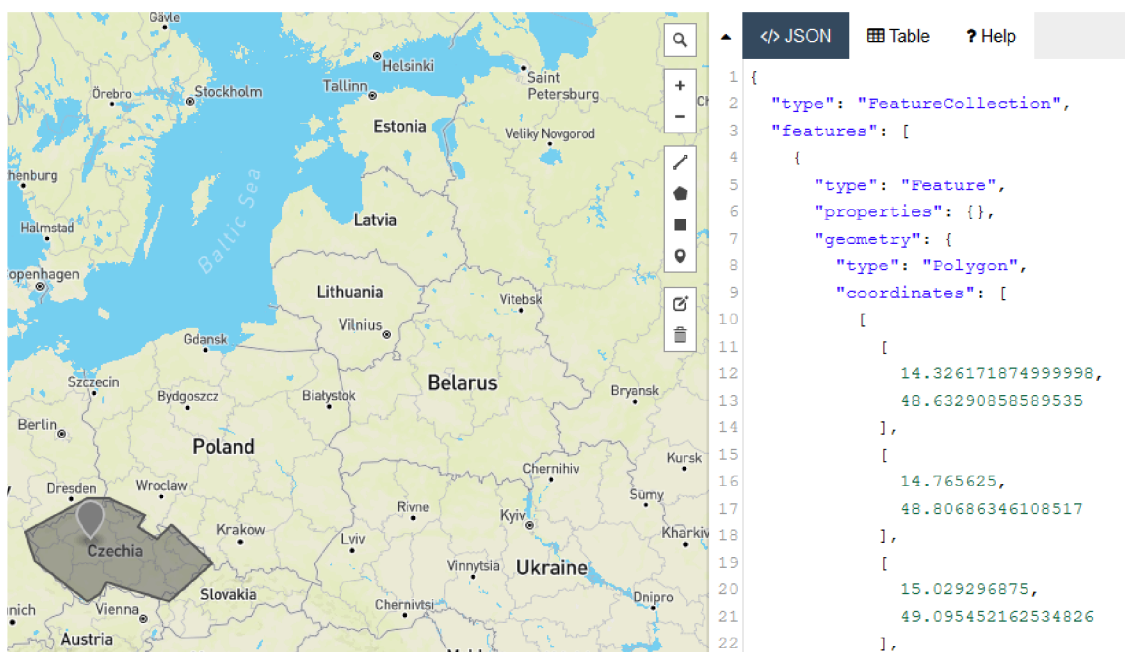
Pro tvorbu interaktivních vizualizací se uživateli nabízí dvě možnosti: Je možné využít autorské systémy, nebo si konkrétní systém sám naprogramovat. Každý z těchto přístupů má své výhody a nevýhody co se týče nutných znalostí uživatele a dosažitelného výsledku.

3.2.1 Autorské systémy

Autorské systémy poskytují základní stavební bloky, pomocí kterých je možné sestavit výslednou vizualizaci. Uživatel je schopný v těchto systémech vytvořit vizualizaci a tu exportovat a publikovat bez znalostí programování. Existují jak obecné autorské systémy, tak systémy zaměřené na konkrétní skupinu uživatelů – například projekt NewsViews umožňující novinářům vytvářet geovizualizace [9]. Jak uvádí [12], u autorských systémů jsou kladeny nároky na schopnost uživatele sestavit vhodný grafický návrh, což může být limitující. Tento problém není natolik výrazný u specificky zaměřených autorských systémů, neboť typicky nabízí méně možností grafického zobrazení.

geojson.io

Jedná se o webovou⁴ aplikaci primárně určenou pro tvorbu GeoJSON objektů. Neposkytuje možnost zobrazovat jiné dimenze, než geografické. Pokud je ovšem cílem vygenerovat například polygony které budou reprezentovat hranice států, jedná se o užitečný nástroj – pomocí geojson.io je možné si dané hranice vytvořit pomocí umístování bodů na mapě a následně je možné si výsledný GeoJSON soubor stáhnout a využít v jiné aplikaci. Uživatelské rozhraní geojson.io je zobrazeno na obrázku 3.1.



Obrázek 3.1: Při editaci mapy (umístování bodů, tvorba polygonů) se mění GeoJSON zápis v editoru vpravo. Stejně tak je možné provádět zápis v editoru a výsledek bude patrný na mapě.

⁴<https://geojson.io/>

ArcGIS

Z kategorie autorských systémů je možné zmínit například geografický informační systém ArcGIS. Tento systém poskytuje nástroje pro tvorbu vyspělých vizualizací a tyto vizualizace umožňuje vkládat na webové stránky pomocí pluginů. Stejně jako výše zmíněné nástroje nabízí podporu formátu GeoJSON, ale také soubory typu KML a další geoprostorové formáty. ArcGIS je dostupný jako desktopový systém, webový systém, platforma pro vývojáře a další. Protože hlavní náplní této práce je rozšíření aplikace Geovisto, nebudu se tímto (a dalšími autorskými systémy) dále zabývat.

3.2.2 Programové řešení a knihovny

Při využití programování je možné dosáhnout přesně zamýšleného výsledku, nicméně jsou kladeny vysoké nároky na odbornost uživatele. Existující nástroje určené pro vizualizaci dat mohou práci programátora usnadnit a poskytnout mu potřebné komponenty.

Leaflet

Jedná se o nástroj podporující tvorbu geografických vizualizací. Leaflet je JavaScriptová knihovna zaměřená na tvorbu multiplatformních interaktivních map. Poskytuje podporu formátu GeoJSON a rozhraní pro práci s mapovými vrstvami. Kromě mapových vrstev nabízí také UI prvky pro interaci s mapou, jako jsou například *popup* okna. V základu disponuje nástroji pro snadnou implementaci podkladové mapy, na kterou lze následně nanášet geografické objekty, jako jsou body, čáry, nebo polygony. Zároveň jde o snadno rozšiřitelný nástroj s velkou komunitou, díky čemuž je k dispozici velké množství pluginů a rozšíření. Typickým způsobem užití je navázání mapy na HTML element, který má mapu zobrazovat a následně je na tuto mapu možné přidávat další vrstvy, značky a podobně. Podkladová vrstva je na mapu přidána pomocí URL API s požadovanou podkladovou mapou. Leaflet podporuje práci s rastrovými i vektorovými vrstvami, přičemž u vektorů je možné použít jak HTML `<canvas>`, tak SVG elementy.

D3.js

D3.js je JavaScriptová knihovna určená k zpracování dat a manipulaci DOM objektů [11]. Dále obsahuje matematické funkce užitečné například pro škálování proporciálních symbolů, nebo geometrické funkce vhodné pro jinak pracné (obzvláště u elementů typu `path`, kde výsledek je popsán jako posloupnost bodů, čar a křivek) sestavování SVG elementů. Oproti knihovně Leaflet se jedná o práci na nižší úrovni abstrakce, protože uživatel nemá k dispozici přímo geovizualizační nástroje, ale popisuje pomocí CSS a SVG velikost a tvar zobrazených elementů. Pro účel modifikace DOM knihovna nabízí vlastní API, které oproti původnímu W3C API nevyužívá imperativní, ale deklarativní přístup a umožňuje modifikovat jak jediný uzel, tak celou skupinu uzlů, a to při využití jediného selektoru. Nevýhodu nutnosti implementace všech dílčích částí vizualizace lze eliminovat propojením s další knihovnou, například výše zmíněnou knihovnou Leaflet. Je poté možné využít vrstvy které knihovna Leaflet poskytuje a ty obohatit o polygony vytvořené za pomoci D3.js. Stejně jako Leaflet i D3.js nabízí velké množství volně dostupných pluginů.

Google maps API

Google maps nabízí vlastní JavaScriptové API pro tvorbu vizualizací. Pomocí této API je možné upravovat podkladové mapy, které Google nabízí, pomocí dalších vrstev, událostí a knihoven třetích stran. Nabízí čtyři druhy podkladových map: terénní, satelitní, silniční a hybridní. Na mapu umožňuje vkládat ukazatele a jejich vzhled modifikovat. Kromě toho podporuje zobrazování dat pomocí polygonů, nabízí podporu formátu GeoJSON a řízení pomocí událostí. Zároveň poskytuje možnost tvorby teplotních map. Tato technologie nicméně není vhodná pro tvorbu aplikace dle zaměření této práce, protože nejde o čistě klientskou aplikaci, je zde nutnost neustálého připojení k internetu⁵ a navíc disponuje pro geovizualizaci nepříliš podstatnými technologiemi, jako je Google Street View, které jsou datově náročné.

⁵Oproti přístupu, kdy se ze serveru pouze stáhne klientský JavaScript, který je následně vykonáván pouze v prohlížeči nehledě na internetové připojení.

Kapitola 4

Knihovna Geovisto

Aplikace Geovisto si klade za cíl řešit problémy zmíněné v předchozí kapitole – tedy poskytnout knihovnu použitelnou pro vizualizaci geografických dat, která obsahuje hotové vrstvy představující různé druhy tematických map a zároveň je snadno rozšiřitelná a umožňuje přetěžování již existujících vrstev a funkcí. Jedná se o modulárně strukturovaný nástroj poskytující API pro rozšíření jádra aplikace a umožňující použití pouze těch částí, které jsou pro konkrétního uživatele relevantní. Geovisto je vyvíjeno jako open-source s cílem umožnit vývojářům, jednotlivým uživatelům a firmám možnost nástroj rozšiřovat a integrovat do svého softwaru. Jedná se o knihovnu v jazyce TypeScript která exportuje React komponentu a je založena na knihovnách Leaflet a D3.js.

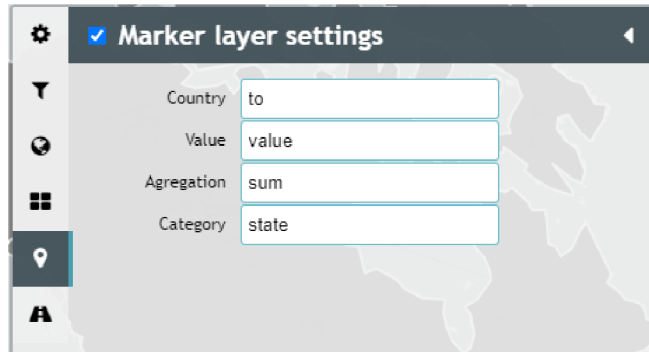
V první části této kapitoly se budu věnovat uživatelskému rozhraní, práci s ním a popisu jednotlivých mapových vrstev, které Geovisto nabízí. V druhé části kapitoly vysvětlím architekturu a datový model aplikace.

4.1 Uživatelské rozhraní a mapové vrstvy

Uživatelské rozhraní se skládá ze dvou hlavních částí: postranního panelu pro práci s vrstvami a samotné mapy, která jednotlivé vrstvy zobrazuje. Postranní panel je plug-in, nejedná se o nezbytnou součást aplikace. Umožňuje přístup k nástrojům, které Geovisto poskytuje, jako je filtrování dat a nastavení jednotlivých mapových vrstev. Práce s těmito nástroji je ovšem možná i bez použití postranního panelu.

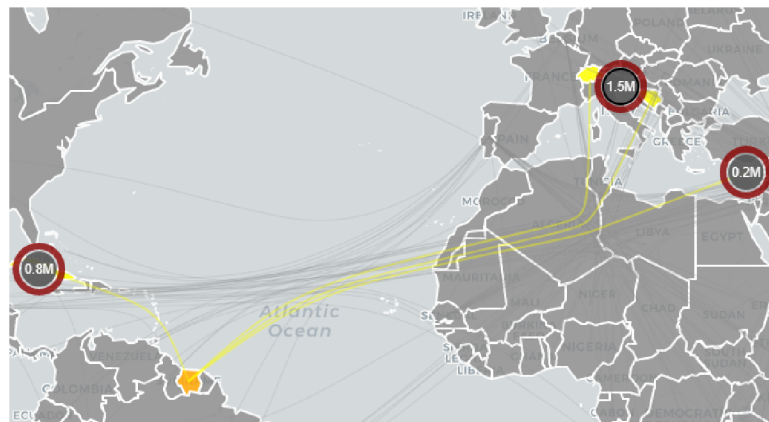
Aplikace Geovisto v době psaní této práce obsahuje čtyři mapové vrstvy: podkladovou mapu, vrstvu pro mapu značek, kartogramovou vrstvu a vrstvu mapy spojení. Každá z těchto vrstev s výjimkou podkladové mapy má vlastní nastavení mapování dat. Podkladová mapa lze pouze skrýt nebo zobrazit. Vrstvy se od sebe navzájem liší nejen použitou vizualizační technikou, ale také počtem dimenzí, které umožňují namapovat. Pro změnu klíče, pod kterým jsou hledány jednotlivé dimenze v datovém souboru slouží postranní panel, jak je zobrazeno na obrázku 4.1. Mapování dimenzí lze nicméně kromě postranního panelu taktéž určit pomocí konfiguračního souboru.

Způsob mapování dat na dimenze grafu lze demonstrovat na vrstvě značek. Vrstva značek, v aplikaci nazvaná *Marker layer*, agreguje data z dané země dle mapování a zobrazuje je formě značek, v tomto případě pomocí modifikace výsečového grafu (tzv. *donut chart*). Tato vrstva umožňuje mapovat čtyři datové dimenze. *Country* je geografický rozměr obsahující identifikátor země, podle kterého je vyhledán příslušný *centroid*, tedy geografický střed státu, a určuje, kde je výsledná značka umístěna. Dimenze *Value* určuje kvalitativní



Obrázek 4.1: Jednotlivé dimenze tematické mapy jsou mapovány na klíče v datovém souboru formátu JSON.

veličinu vyjadřující intenzitu zobrazené značky. Další dimenzí je *Agregation*, která určuje, jakým způsobem se agregují hodnoty *Value* pro záznamy se stejnou geografickou polohou určenou v *Country*. Poslední dimenzí této vrstvy je *Category* sloužící pro přiřazení kategoričkého rozměru konkrétnímu záznamu a umožňující zobrazit více kategorií v jednom místě. Značky podporují shlukování na základě míry přiblížení mapy, přičemž při najetí myši nad danou značku je vyobrazeno, jakou oblast daná značka shlukuje. Shluk lze rozložit na jednotlivé značky buď přiblížením, nebo kliknutím na konkrétní shluk, kdy k přiblížení na danou oblast dojde automaticky. Jednotlivé vrstvy lze kombinovat. Výsledný vzhled při kombinaci všech dostupných vrstev je zobrazen na obrázku 4.2.



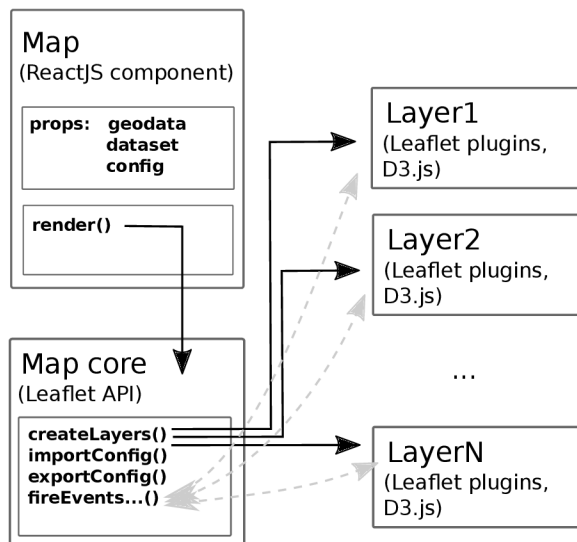
Obrázek 4.2: Zvýrazněné polygony představují kartogramovou vrstvu, *donut charts* vrstvu značek a křivky vedoucí mezi body vrstvu spojení.

4.2 Architektura

Architekturu (zobrazena na obrázku 4.3) je možné rozdělit do dvou hlavních částí: jádra a nástrojů. Jádro je obaleno v komponentě knihovny React *GeovistoMap*. Při inicializaci vykreslí komponenta jádro na základě svého vstupu.

Jádro nejprve zpracuje svá vstupní data a následně inicializuje globální stav mapy. Pokud je uživatelem poskytnut konfigurační soubor, je deserializován a globální stav mapy je modifikován dle těchto konfiguračních dat. Jádro následně vytvoří Leaflet mapu a HTML elementy pro nástroje. Pokud nastane v aplikaci událost, je tato distribuována na příslušné nástroje.

Nástroje představují prvky mapy, jako je postranní panel a jednotlivé vrstvy. Nástroje mají modulární architekturu a proto je možné je aktivovat/deaktivovat či přidávat nebo odebrat. Měly by být vzájemně nezávislé a je možné je vytvořit za pomoci rozdílných knihoven a pluginů.



Obrázek 4.3: Obalující komponenta vykresluje mapové jádro, které spravuje jednotlivé nástroje. Převzato z [12].

4.3 Datový model

Jádro pracuje se třemi druhy dat. Prvním druhem jsou **props**, které se předávají pomocí React komponenty a jedná se o způsob, jak programově modifikovat chování aplikace. Druhým druhem jsou konfigurační data, která obsahují globální stav mapy. Tato data jsou serializována ve formě JSON a obsahují informace o míře přiblížení, vzhledu, aktivních nástrojích a informace o mapování dimenzí dat v jednotlivých vrstvách. Konfigurační data je možné z uživatelského hlediska uložit a použít v budoucnu.

Posledním druhem jsou samotná vstupní data sloužící pro vizualizaci. Ta jsou rozdělena na dva typy. Prvním z nich jsou data poskytující hodnoty datových dimenzí s vazbou na určité místo dle identifikátoru. Druhým typem jsou data ve formátu GeoJSON, která představují geografickou dimenzi dat rozlišenou identifikátorem. Jedná se o polygony, na které se pomocí identifikátoru ve formě ISO-3166-alpha3 [13] odkazuje první typ vstupních dat, který na tyto polygony mapuje veličiny v něm obsažené. Kromě datového souboru s polygony obsahuje Geovisto ještě datový soubor obsahující tzv. *centroidy*, které představují geometrický střed států a slouží pro umístění značek a výchozích a cílových bodů pro mapu spojení.

Kapitola 5

Analýza

V této části se zaměřím na analýzu požadavků uživatelů na snadnou tvorbu diagramů v geografických souřadnicích. Cílová skupina je reprezentována běžnými uživateli, kteří pracují s počítačem a daty a mají potřebu tato data vizualizovat.

5.1 Cílová skupina

Cílovým uživatelem je kdokoliv s přístupem k počítači a daty s geografickým rozměrem, která je třeba vizualizovat. Jednotlivci v takto široké cílové skupině mají rozdílné zkušenosti s vizualizací, rozdílná data a s tím související rozdílné požadavky. Například novinář popisující aktuální epidemickou situaci má zcela jistě rozdílné požadavky na vizualizaci než firma zkoumající zastoupení konkurence a jejich podíl na trhu v konkrétních městech.

Rozdílní uživatelé mají rovněž jiné schopnosti a jiná omezení. Je vhodné umožnit uživatelům, kteří dovedou programovat možnost výsledné zobrazení upravit či rozšířit programově, ale zároveň je nutné, aby v základu aplikace disponovala dostatečným množstvím funkcí, pomocí kterých uživatel bez znalosti programování bude schopen plnohodnotně vizualizovat svoje data.

Jednotlivé kategorické dimenze bývají typicky odlišeny pomocí rozdílných barev. Často opomíjenou skupinou uživatelů jsou lidé s nějakou formou poruchy barvocitu. Nejčastěji se jedná o poruchy vnímání červených a zelených odstínů, které se nazývají *deuteranopie* a *protanopie*, méně častěji o neschopnost vnímat modrou a žlutou zvanou *tritanopie* [20]. Pro takové uživatele jsou tedy mapy s předdefinovanými barevnými schémata bez možnosti barvu přizpůsobit často problematické.

5.2 Analýza vstupních dat

Data budou mít předem danou strukturu, nicméně povaha daných dat není známá a nemělo by na ní záležet – aplikace by měla být schopna vizualizovat generická vstupní data. Je tedy nutné uživateli poskytnout mechanismus, pomocí kterého si může mapovat indexy ze vstupních dat na dimenze, které budou vizualizované.

Geografická lokace se udává dle dvou parametrů – zeměpisné šířky (*latitude*) a zeměpisné délky (*longitude*). V případě trojrozměrného zobrazení je možné připojit ještě údaj třetí, a to nadmořskou výšku. Tato data lze zapsat jednoduchým způsobem například ve formátu JSON, nebo XML. Je zároveň možné použít i bohatších standardů jako je GeoJSON, nicméně tento způsob zápisu je pro běžného uživatele nepříliš přehledný a zdlouhavý.

Data mají kromě geografické lokace často více než jednu datovou dimenzi – ať už kvalitativní nebo kvantitativní. Úkolem nástroje pro vizualizaci je poskytnout možnost tyto dimenze zobrazovat vhodným způsobem tak, aby byl maximalizován počet dimenzí bez ztráty čitelnosti. Každý druh tematických map se hodí pro jiný druh analýzy a výběr vhodného druhu zobrazení se tím pádem odvíjí od povahy vizualizovaných dat. Stejně tak rozdílné druhy zobrazení nabízejí odlišné množství dimenzí, které jsou schopny vizualizovat.

Obecně platí, že čím vyšší je počet dimenzí, tím komplexnější musí být výsledné zobrazení. Pro větší soubory dat je vhodnější využít jednodušší zobrazení s menším počtem dimenzí. Pro korektní zobrazení více kvalitativních dimenzí v jedné geografické lokaci je vhodné využít takový druh zobrazení, který tyto dimenze obsáhne v jednom zobrazeném symbolu. Pokud toto není možné, jsou tyto symboly vykresleny přes sebe a dochází ke snížení čitelnosti. Zároveň je nutné, aby v případě, že pro jediný záznam existuje více kvantitativních veličin, byla každá takováto veličina navázána na veličinu kvalitativní a bylo tím pádem možné tyto veličiny odlišit.

Bez ohledu na počet dimenzí lze datové mapy pracující s geografickými souřadnicemi klasifikovat dle povahy vstupních dat na mapy zobrazující:

- kvantitativní intenzitu,
- geografickou hustotu,
- agregovaná data a poměr mezi nimi,
- intenzitu odvozenou z hustoty a kvantitativních hodnot.

Výsledkem této práce by měly být mapové vrstvy splňující tuto klasifikaci. Jednotlivé druhy zobrazení by mělo být možné kombinovat pro poskytnutí více pohledů na vstupní data a tím pádem umožnění hlubší analýzy zobrazených dat.

5.3 Výstup analýzy

Ná základě analýzy vstupních dat a vhodnosti jednotlivých metod jsem dospěl k závěru, že vhodný nástroj pro vizualizaci v geografických souřadnicích je takový, který automaticky převede datový soubor na generický datový model, se kterým je aplikace schopna pracovat bez nutnosti transformovat data ze strany uživatele. Tento nástroj musí poskytovat dostatek možností vizualizovat datové soubory rozdílné povahy a to tak, aby nediskriminoval uživatele s omezeným barvocitem. Existující nástroje jsou často obtížně rozšiřitelné a jejich využití v aplikacích třetích stran může být problematické nebo nákladné. Jako největší nevýhodu existujících řešení jsem shledal omezené možnosti kombinování více druhů zobrazení v jednom pohledu. Například Tableau umožňuje kombinovat pouze dvě vrstvy, ArcGIS umožňuje kombinovat více vrstev, ale musí se jednat o stejné typy zobrazení. Tento nedostatek výrazně omezuje možnost analýzy komplexních datových souborů a znesnadňuje hledání vztahů mezi jednotlivými datovými záznamy.

Kapitola 6

Návrh řešení

V reakci na předešlou kapitolu a v ní diskutované problémy bude v této kapitole popsán návrh nástrojů umožňujících vizualizovat data v geografických souřadnicích. Tyto nástroje budou představovat mapové vrstvy *dot map*, *bubble map*, *spike map* a *heat map*. Všechny grafické návrhy v této kapitole byly vytvořeny pomocí nástroje Figma¹.

6.1 Vstupní data

Nástroje obalující jednotlivé vrstvy a jejich funkčnost budou koncipovány jako součást klientské aplikace, která kromě úvodního načtení nevyžaduje interakci se serverem. Tyto nové vrstvy budou navzájem nezávislé a budou pracovat s předem neznámými generickými daty ve formátu JSON, které mohou vypadat následovně:

```
{
  "lat": 50.0256,
  "long": 1.5235,
  "activeCasesTotal": 163,
  "activeCasesDetail": [
    {"category": "sick", "cases": 89},
    {"category": "recovered", "cases": 68},
    {"category": "deceased", "cases": 6}
  ]
}
```

Agregace vstupních dat

Uživatel si pomocí uživatelského rozhraní daného nástroje tato data bude mapovat na jednotlivé dimenze konkrétního grafu. Pokud bude více záznamů se stejnou geografickou lokací, bude záležet na konkrétní vrstvě, jak tato data zobrazí.

Dále platí, že pokud si uživatel namapuje na grafovou dimenzi takový index, který se v jednom záznamu pro jednu geografickou lokaci vyskytuje vícekrát, je tento záznam rozložen a zpracován, jako by se jednalo o několik různých záznamů. Pokud by například uživatel namapoval na dimenzi grafu `activeCasesDetail.category` z příkladu výše, bude výsledné zobrazení stejné, jako by se jednalo o takovýto set dat:

¹<https://www.figma.com/>


```

{
  "lat": 50.0256,
  "long": 1.5235,
  "activeCasesTotal": 163,
  "activeCasesDetail": {
    "category": "sick",
    "cases": 89
  }
},
{
  "lat": 50.0256,
  "long": 1.5235,
  "activeCasesTotal": 163,
  "activeCasesDetail": {
    "category": "recovered",
    "cases": 68
  }
},
{
  "lat": 50.0256,
  "long": 1.5235,
  "activeCasesTotal": 163,
  "activeCasesDetail": {
    "category": "deceased",
    "cases": 6
  }
}

```

Pokud vrstva podporuje agregaci dat, dojde následně k agregaci těchto záznamů.

6.2 Zvolené tematické mapy

V reakci na diskutované množství dimenzí v kapitole 5 jsem se rozhodl vytvořit čtyři mapové vrstvy, kdy každá slouží jinému účelu a je vhodná pro jiný druh analýzy. Jedná se o teplotní mapu (*heat map*), která je vhodná především pro vizualizaci intenzity sledovaného jevu, metodu teček (*dot map*), která slouží pro nalezení precizní polohy veličin a je vhodná pro datové soubory obsahující velké množství záznamů, kde jediným důležitým aspektem je jejich poloha a kategorie. Pro komplexní datové soubory jsou vhodné metody proporciálních symbolů – *bubble map* a *spike map*.

Teplotní mapa zobrazuje pouze dvě datové dimenze a to geografickou lokaci a intenzitu pro danou lokaci. Tato mapa se využívá pro zobrazení intenzity, která je určena dle množství výskytů veličiny na konkrétním území. Je možné měnit poloměr kruhu použitý pro výpočet této intenzity nebo rychlost klesání intenzity od středu k okrajům kruhu a provádět další modifikace, které mají vliv na výsledné zobrazení, jinak se ale tato metoda vzhledem ke své povaze nehodí pro zobrazení většího množství datových dimenzí.

Metoda teček je schopna díky své jednoduchosti zobrazovat pouze jednu² veličinu. Tato metoda se hodí primárně pro zobrazování kvalitativní veličiny, nicméně při vhodném použití třídních intervalů je možné tuto metodu využít i pro veličiny kvantitativní. Kvalitativní veličiny či třídní intervaly bývají typicky odlišeny barvou. Takovéto značky lze shlukovat, nicméně slučování lze provádět pouze mezi značkami které zobrazují stejnou kategorii, nebo stejný třídní interval.

Největší množství datových dimenzí jsou schopny zobrazit metody proporciálních symbolů. Pro vyjádření kvantitativních datových dimenzí se typicky využívá velikost symbolu, která může být buď absolutně škálovaná, nebo odstupňovaná pro třídní intervaly. Pojem *velikost symbolu* se liší dle použitého symbolu. V případě *spike map* je velikost chápána jako výška bodce, v případě *bubble map* je velikost dána plochou kruhu. Díky barevnému odlišení je možné přidat do tohoto druhu diagramů další kategorickou dimenzi. Kruh může být nahrazen koláčovým grafem a tím pádem lze dosáhnout přidání dalších kvalitativních a kvantitativních dimenzí. Oproti tomu *spike map* kvůli povaze použitého symbolu neumožňuje agregovat v jednom místě více záznamů s rozdílnými kvalitativními dimenzemi, a tak je bude vykreslovat přes sebe.

6.3 Uživatelské rozhraní

Protože aplikace Geovisto již disponuje mapovými vrstvami a poskytuje pro ně uživatelské rozhraní ve formě ovládacího panelu, budu toto uživatelské rozhraní pro ovládání a nastavení parametrů jednotlivých vrstev pouze rozšiřovat o nové vrstvy a jejich ovládací prvky. Toto uživatelské rozhraní je minimalistické a i při rozbaleném ovládacím panelu zobrazuje velkou část mapy. Návrh aplikace Geovisto s přidávanými ikonami pro nové vrstvy je na obrázku 6.1.



Obrázek 6.1: V postranním panelu jsou přidány ikony pro jednotlivé mapy.

²Geografickou lokaci budu jako datovou dimenzi v této sekci dále ignorovat, neboť je přítomna u všech mnou rozebíraných druhů zobrazení.

6.4 Mapové vrstvy

V této sekci se budu zabývat podrobným popisem a vizuálním návrhem jednotlivých zvolených vrstev. Všechny vrstvy s výjimkou teplotní mapy umožňují odlišit kvalitativní dimenze pomocí barev v ovládacím panelu pomocí sekce *Category Colors*. Pomocí této funkce je možné data barevně odlišit dle hodnoty obsažené pod klíčem namapovaným v *Category*. Tuto namapovanou hodnotu lze porovnávat, k dispozici jsou operace =, !=, <, >, <= a >=. Pro účel zahrnutí více hodnot je navíc přidána možnost `reg` pro regulární výraz.

Výsledek takového porovnání lze následně obarvit uživatelem zvolenou barvou – tu je možné zadat pomocí HTML kódu pro barvy, případně kliknout na pole s barvou a barvu si zvolit pomocí tzv. *color pickeru*. Takto vytvořené barevné kategorie lze jednoduše přidávat a odebírat. Toto řešení bylo zvoleno proto, že počet rozdílných hodnot v dimenzi *Category* může být v podstatě neomezený, je tím pádem vhodnější data zobrazovat jednotnou barvou a na uživateli ponechat, které hodnoty si chce na výsledném zobrazení zvýraznit. Zároveň, jak bylo zmíněné v Analýze, existují uživatelé s poruchami barvocitu a toto řešení jim umožňuje zvolit si barvy, které budou schopni odlišit. Při použití dat v sekci Vstupní data je tedy například možné namapovat si do pole *Category* klíč `activeCasesTotal` a následně si zvýraznit všechny případy, kde je tato hodnota větší než 150. Lze si povšimnout, že toto řešení taktéž umožňuje pracovat s kvantitativní veličinou jako s veličinou kategorickou pomocí rozdělení hodnot na třídní intervaly odlišené barvou.

Dot map

Tato vrstva představuje metodu teček. Umožňuje mapovat geografickou lokaci jako pár (*lat, long*) a jednu dodatečnou veličinu. Tato veličina by měla být kvalitativní, ale je možné pracovat i s kvantitativními daty. Jak je patrné na obrázku 6.2, uživatel si všechny dimenze mapuje pomocí select boxů. Pro zobrazení teček na mapě je postačující, aby uživatel namapoval korektně dvojici (*lat, long*). Mapování dimenze *Category* nemá samo o sobě žádný vliv, nicméně umožňuje třídění pomocí *Category Colors*.

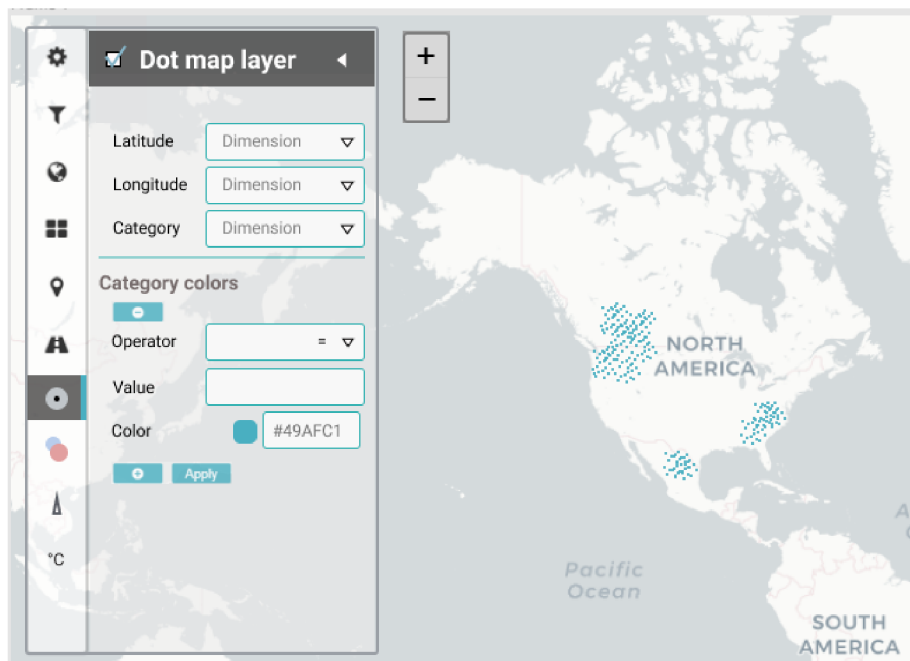
Tato vrstva nepodporuje agregaci dat, takže více záznamů se stejnou lokací bude vykresleno přes sebe. Viditelný tedy bude pouze poslední záznam s danou lokací. Při namapování klíče, který je v jednom záznamu obsažen vícekrát (a tudíž dojde k rozložení na několik záznamů, kde je každý klíč maximálně jednou) se rovněž data vykreslí přes sebe.

Při změně míry přiblížení dochází ke změně měřítka vykreslených teček.

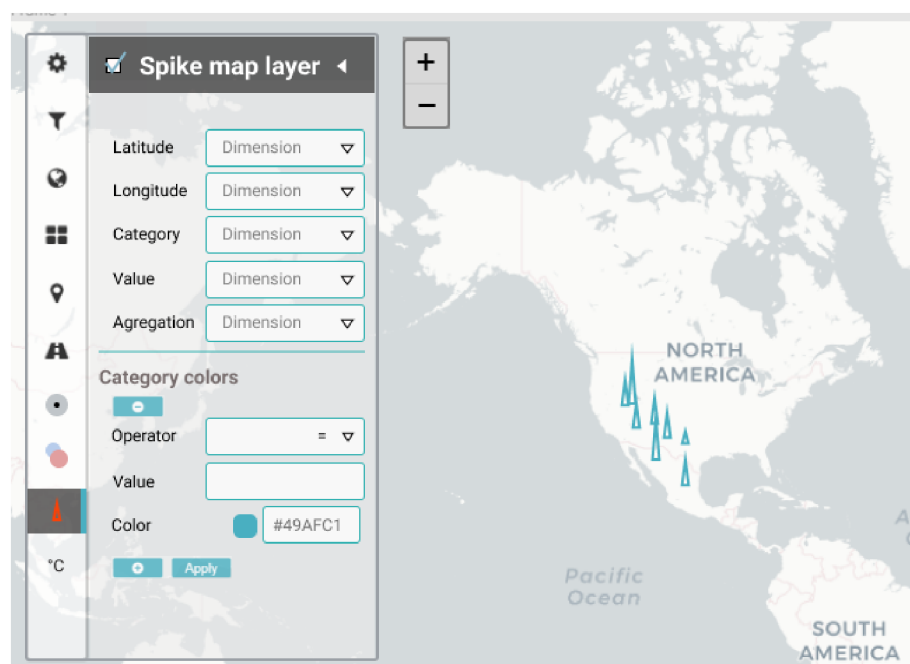
Spike map

Jedná se o vrstvu znázorňující mapu proporciálních symbolů, kde tímto symbolem je bodce. Tato vrstva umožňuje kromě mapování geografické lokace mapovat také kvantitativní dimenzi *Value* a dimenzi *Category*, která se chová stejně jako u metody teček. Pokud nějaký záznam neobsahuje index zadaný do pole *Value*, nebo jeho hodnota není numerická, nebude tento záznam zobrazen. Ve výsledném zobrazení je kvantitativní veličina vyjádřena výškou bodce, kvalitativní veličina barvou, kterou si uživatel může nastavit pro konkrétní hodnoty.

Tato vrstva podporuje agregaci dat, pokud je ve vstupních datech více záznamů se stejnou lokací. K výběru agregační operace slouží select box a nabízí možnost sumy `sum` a počtu `count`. V případě volby počtu je výška bodce dána celkovým počtem záznamů pro danou geografickou lokaci – v tomto případě jsou zahrnuty i ty záznamy, které nemají vyplněnou hodnotu *Value* nebo je tato hodnota neplatná, neboť ta pro tento způsob vizualizace není podstatná. Suma sčítá dohromady hodnoty namapované ve *Value* a opět platí, že záznamy



Obrázek 6.2: Ovládací panel pro vrstvu metody teček s návrhem vizualizace pomocí teček.



Obrázek 6.3: Ovládací panel pro spike map spolu s návrhem vzhledu vizualizace dat.

tuto hodnotu musí obsahovat a ta musí být numerická. Nehledě na volbu agregační metody platí, že agregovány jsou pouze ty záznamy, které mají stejnou geografickou lokaci a zároveň stejnou hodnotu pod klíčem *Category*. Vznikají tedy páry (*Category, agregatedValue*), kde *agregatedValue* je tvořena buď sumou hodnot *Value* (při volbě *sum*) nebo jejich počtem

pro danou kategorii *Category* (při volbě *count*). Data s rozdílnými hodnotami v *Category* a stejnou geografickou lokací jsou vykreslovány přes sebe, je tedy možné, že dojde k překrytí některých výskytů dat.

Reakcí na změnu přiblížení je stejně jako u metody teček změna velikosti vykreslených symbolů.

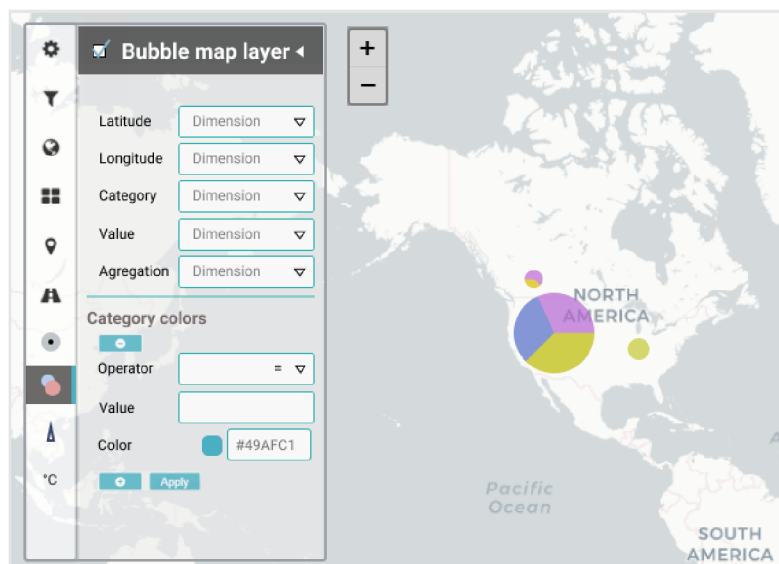
Bubble map

Pro vizualizaci většího množství kategorických a kvantitativních dimenzí jsem zvolil mapu proporciálních symbolů zvanou *bubble map*. Místo obyčejného kruhu bude použit koláčový graf, který umožňuje rozlišení většího počtu dimenzí. Tato mapová vrstva má stejný počet dimenzí k namapování jako spike map, nicméně s těmito hodnotami pracuje jiným způsobem a umožňuje tak vizualizace bohatších datových záznamů.

Jako jediná vrstva umožňuje agregovat data napříč různými hodnotami v *Category* pro jeden zobrazený graf. Při agregaci dochází opět k tvorbě dvojic (*Category, aggregatedValue*). Hodnoty v takto rozdělených dvojicích se následně sečtou a tento výsledný součet se použije pro určení poloměru kruhu. Při barevném třídění dle kategorií je následně možné v kruhu barevně zvýraznit jednotlivé kategorie a tím vytvořit koláčový graf. Velikost jednotlivých výsečí je dána poměrem hodnoty *aggregatedValue* pro danou *Category* vůči součtu *aggregatedValue* napříč všemi kategoriemi pro danou lokaci. Výsledná vizualizace je naznačena na obrázku 6.4.

Díky možnosti agregovat data napříč kategoriemi je zároveň možné vizualizovat záznamy, které mají pro jeden klíč více hodnot v rámci jediného grafu. Stejně jako u ostatních vrstev dojde k rozložení takového záznamu, který je následně agregován a zobrazen pomocí jediného koláčového grafu.

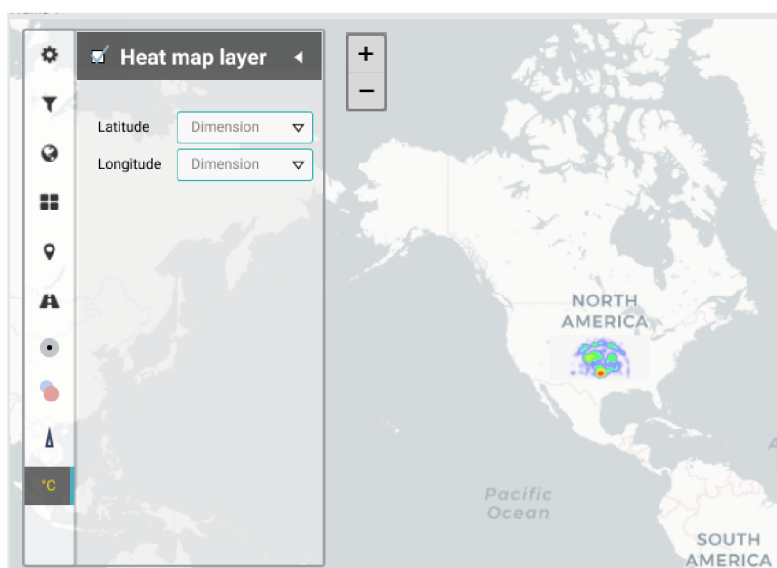
Na oddálení tato vrstva reaguje shlukováním překrývajících se grafů, dochází tedy k další agregaci dat. Při přiblížení se shluky rozpadají na jednotlivé záznamy pro větší míru detailu.



Obrázek 6.4: Pomocí koláčového grafu je možné zobrazit více kategorických a kvantitativních veličin a jejich poměr na jednom místě.

Heat map

Z pohledu zadávání dat je to nejméně komplexní vrstva. Tato mapa pracuje pouze s geografickou lokací a intenzitou. Pro každý umístěný bod je vytvořena kruhová oblast a v místech, kde se překrývá více těchto oblastí je zobrazena vyšší intenzita. Body se stejnou geografickou lokací jsou vykreslovány přes sebe a tím pádem v místě, kde se nachází více bodů na jednom místě vzniká intenzivnější zbarvení. Zbarvení u tohoto druhu mapy je reprezentováno spojitým barevným spektrem od modré po červenou, kde modrá znamená nejnižší intenzitu a červená naopak intenzitu nejvyšší. Způsob vizualizace pro tento druh mapy je zobrazen na obrázku 6.5. Na změnu přiblížení reaguje tato mapa změnou velikosti poloměru pro kruhovou oblast, ze které se následně počítá výsledná intenzita.



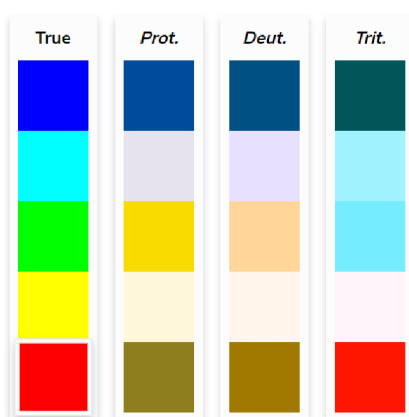
Obrázek 6.5: Intenzita je vyobrazena pomocí spojitého barevného spektra.

Mapa bude kromě mapování datových dimenzí umožňovat nastavení jednotlivých parametrů ovlivňující výsledné zobrazení, jako je poloměr kruhu nebo vztah mezi zobrazenou intenzitou a mírou přiblížení. Tato mapa neumožňuje oproti ostatním vrstvám nastavení barev pro jednotlivé kategorie, protože nedisponuje žádnou kvalitativní dimenzí, na základě které by barevné mapování bylo možné provést. Nicméně, kromě obvyklého barevného spektra bude poskytovat předdefinovaná barevná schémata pro uživatele s poruchou vnímání barev určitého spektra.

Barvy pro jednotlivá barevná schémata byly vybrány pomocí simulačního nástroje³ takovým způsobem, aby simulované barvy pro konkrétní druhy ztráty barvocitu co nejlépe odpovídaly barvám, které vidí člověk se zdravým zrakem. Zároveň byly jednotlivé barvy poskládány tak, aby se udržel typický formát přechodu ze studených barev pro nízkou intenzitu po teplé barvy, které reprezentují vysokou intenzitu. Tento způsob zaručuje, že uživatel se ztrátou barvocitu si – při zvolení správného gradientu – bude moct být jistý, že to, co vidí, je produktem datového souboru a jeho vizualizace, nikoliv zkreslením způsobeným nesprávným vnímáním barev. Na obrázku 6.6 je zobrazeno výchozí barevné schéma.

³<https://davidmathlogic.com/colorblind/>

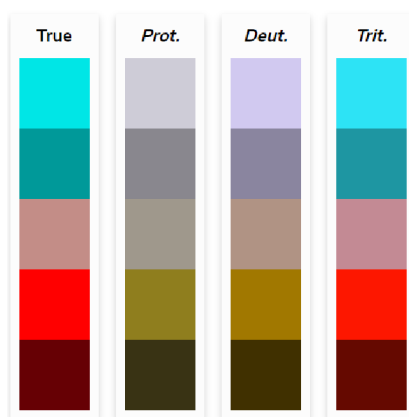
U simulací protanopie a deuteranopie je problematické, že barva, která je ve skutečnosti zelená je zkreslena na relativně sytou barvu ve žlutých, respektive oranžových odstínech. Oproti tomu žlutá barva je vnímána uživateli s těmito poruchami jako velice nevýrazná béžová. Toto může způsobit záměnu vnímané intenzity u těchto dvou barev, tedy že barva reprezentující nižší intenzitu (zelená) bude vnímána jako intenzivnější oproti barvě, která ve skutečnosti má představovat vyšší intenzitu (žlutá). Řešením je volba barevné palety, která je zobrazená na obrázku 6.7 a je stejná pro běžné uživatele i uživatele se ztrátou barvocitu. Tato paleta zároveň musí mít jednotnou posloupnost změn barev či sytostí odstínů stejné barvy tak, aby nemohlo dojít ke zkreslení pozorované intenzity. Problém s nejednotnou změnou odstínů je viditelný i u simulace pro tritanopii. Z prvních tří barev není zřejmé, jestli tmavší odstín stejné barvy znamená vyšší či nižší intenzitu. Řešením (zobrazené na obrázku 6.8) je opět změna barev a sjednocení přechodů mezi nimi tak, aby nedocházelo ke zmatení uživatele.



Obrázek 6.6: Výchozí barevné schéma a jeho simulace pro protanopii, deuteranopii a tritanopii. Barva nejvíce nahoře symbolizuje nejnižší intenzitu, barva vespodu symbolizuje intenzitu nejvyšší.



Obrázek 6.7: Pro ztrátu barvocitu charakterizovanou neschopností vnímat červenou a zelenou barvu byl zvolen gradient od modré přes žlutou po hnědou.



Obrázek 6.8: Pro tritanopii bylo vybráno schéma od tyrkysové po temně červenou. Simulované barvy jsou totožné s barvami, které vnímá člověk bez ztráty barvocitu.

Kapitola 7

Implementace

V této kapitole popíšu použité technologie a způsob implementace jednotlivých vrstev navržených v minulé kapitole.

7.1 Použité technologie

Protože je rozšiřovaná aplikace, popsaná v kapitole 4, implementována v jazyce JavaScript, byl stejný jazyk použit pro implementaci navržených mapových vrstev. Samotné mapové vrstvy jsou implementovány za pomoci knihovny Leaflet a pro škálování velikosti proporciónálních symbolů byla použita knihovna D3.js. Pro instalaci a správu balíčků třetích stran se stará NPM¹.

Jedním z těchto balíčků je Leaflet plugin `Leaflet.markercluster`², který je využit pro shlukování symbolů u *bubble map*. Mimo jiné umožňuje nastavit, jak vzdálené musí být jednotlivé symboly, aby došlo ke shlukování, či vzhled zobrazeného shluku. Dalším použitým pluginem je `Leaflet.heat`³ vykreslující teplotní mapu na základě vstupních dat. Detailní konfigurace jednotlivých pluginů budou popsány u konkrétních mapových vrstev.

7.2 Rozšíření stávající architektury

V kontextu aplikace Geovisto je vše, kromě obalující komponenty knihovny ReactJS a mapového jádra, oddělitelným modulem. Tyto moduly reprezentují jednotlivé nástroje. Pro přidání mapových vrstev bylo nutné rozšířit aplikaci o nové nástroje a část stávající architektury modifikovat.

Mapové jádro, reprezentované třídou `GeovistoMap`, vykresluje Leaflet mapu a jednotlivé nástroje na základě konfiguračního souboru a `props`, které obdrží od obalující komponenty implementované v ReactJS. Pomocí `props` se jádru předávají instance tříd, které reprezentují jednotlivé nástroje. Do konstruktoru nástrojů představujících mapové vrstvy se předává objekt jazyka JavaScript ve formě `{ id: "identifikátor-nástroje"}`, který musí být shodný s identifikátorem v konfiguračním souboru. Konfigurační soubor na základě identifikátorů uchovává serializovaný stav nástrojů. Stavem nástrojů rozumíme například to, jaké datové dimenze jsou namapované na konkrétní grafové dimenze nebo zda vykreslit ovládací panel pro určité vrstvy. Pro možnost využívat nové mapové vrstvy bylo nutné

¹Node Package Manager <https://www.npmjs.com/>

²<https://github.com/Leaflet/Leaflet.markercluster>

³<https://github.com/Leaflet/Leaflet.heat>

jejich instance zaregistrovat v React komponentě `ReactGeovistoMap`, která tyto instance následně předává do jádra. Zároveň je nutné pro správnou funkčnost doplnit informace o nových vrstvách do konfiguračního souboru.

Protože všechny mnou navržené vrstvy reagují na míru přiblížení, bylo nutné umožnit reagovat na tuto událost. Jádro již disponovalo metodou `dispatchEvent`, nicméně tato metoda se používá pro obsluhu událostí, které jsou vyvolány programově, navíc události deleguje na všechny nástroje. Protože změna úrovně přiblížení je událost, která je vyvolána uživatelem, rozhodl jsem se použít generičtější řešení. Protože jádro tvoří API pro samotný objekt mapy knihovny Leaflet, je reakce na změnu přiblížení řešena jako *event listener*, který je při vytvoření vrstev konkrétních nástrojů navázán na tento objekt mapy knihovny Leaflet. Jako parametr je předána *callback* metoda nástroje, která má sloužit jako reakce na specifikovanou událost. Dále tedy nástroj reaguje na události vyvolané uživatelem na mapě bez nutnosti delegace událostí z jádra.

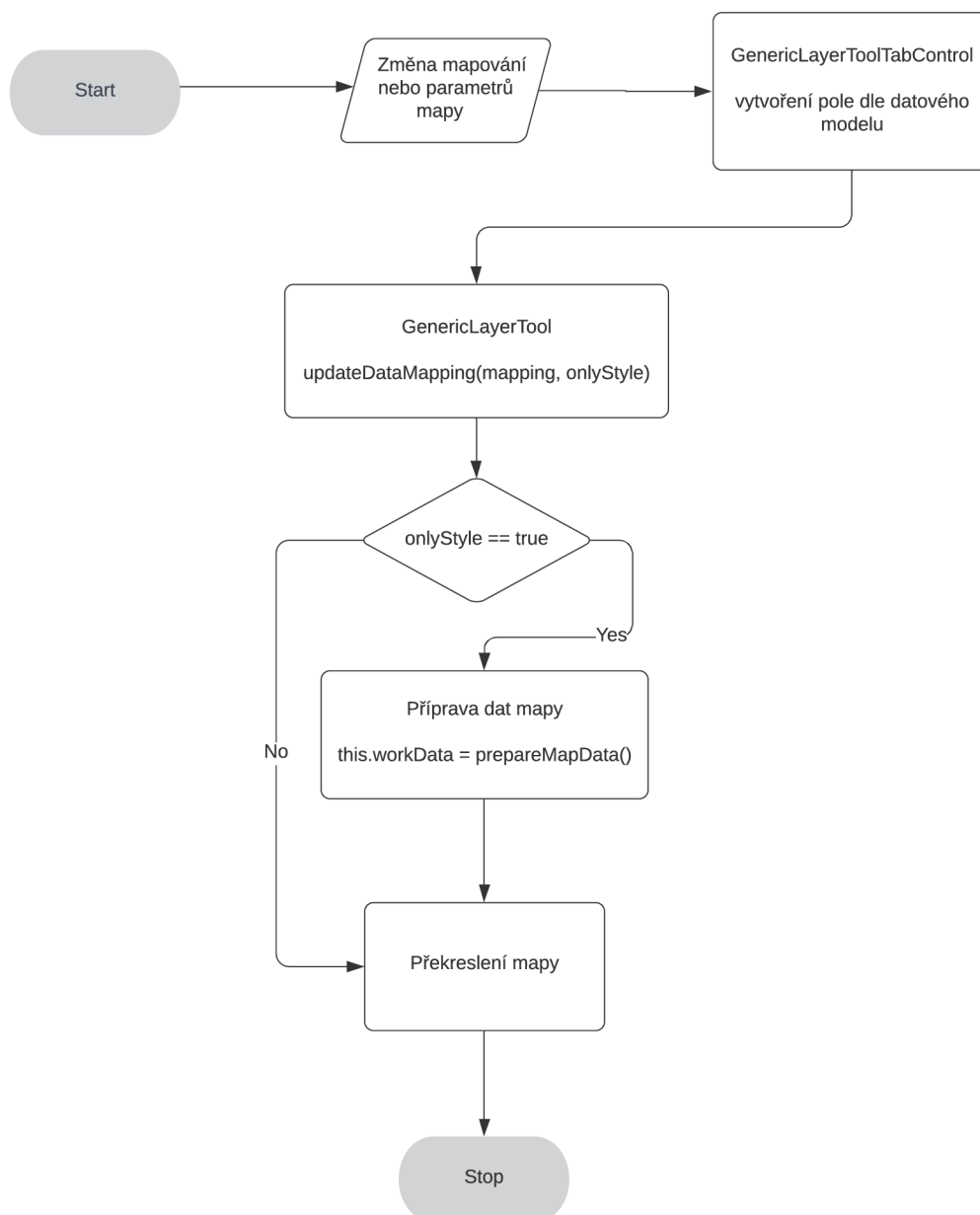
7.3 Implementace mapových vrstev

V této sekci popíši způsob implementace jednotlivých mapových vrstev. V první řadě je vhodné zaměřit se na ty části, které mají jednotlivé vrstvy společné (nebo se v nich liší pouze minimálně). Každá vrstva se skládá ze tříd určených pro obsluhu postranního panelu a dále ze tříd, které zpracovávají vstupní data a na jejich základě tvoří grafickou vizualizaci dat na podkladové mapě.

Obě skupiny tříd, tedy ty obsluhující postranní panel, i ty zajišťující vizualizace obsahují tři druhy objektů. V obecně rovině lze tyto objekty pojmenovat jako `Tool`, `ToolDefaults` a `ToolState`. Třída typu `Tool` řeší samotnou funkcionalitu, tedy vizualizaci dat nebo vykreslování a obsluhu postranního panelu. Pro svou činnost využívá tříd `ToolDefaults` a `ToolState`, které jí poskytují implicitní nastavení a obsahují datový model vrstvy, respektive udržují a nastavují stav konkrétního nástroje. Tento způsob řešení je vhodný z toho důvodu, že umožňuje serializovat a exportovat stav nástroje v podobě konfiguračního souboru. Pokud některé aspekty nejsou nastaveny, použijí se pro serializaci hodnoty poskytované třídami `ToolDefaults`. Takto serializovaný stav lze následně importovat, v třídách typu `ToolState` dojde k deserializaci a nastavení nástroje do stavu, který je popsán v importovaném souboru. Opět platí, že pokud v importovaném souboru chybí popis stavu některých dílčích částí systému, je využita služba tříd `ToolDefaults`.

Z hlediska implementace jsou nejzajímavější třídy typu `Tool`, které lze opět v obecně rovině pro účel jejich odlišení v textu pojmenovat `GenericLayerTool` pro třídu zajišťující vizualizaci a `GenericLayerToolTabControl` pro třídu obsluhující ovládací panel. V implementaci konkrétních vrstev je slovo *Generic* nahrazeno za název příslušné vrstvy, například tedy `HeatLayerTool` a `HeatLayerTabControl`.

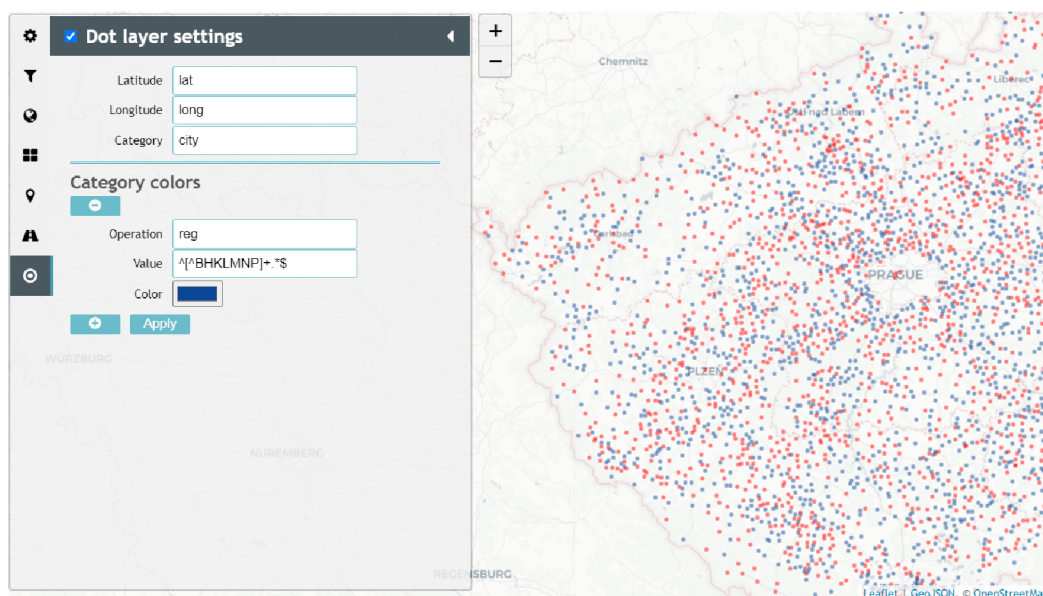
Třída obsluhující postranní panel vykresluje jednotlivé ovládací prvky jako formuláře a tlačítka a na ty navazuje funkce obsluhující tyto prvky. Příkladem může být změna mapování dimenzí dat, kdy reakcí je v `GenericLayerTabControl` propsání namapovaných dat do pole vytvořeného dle datového modelu. Následně se takto vytvořené pole předá jako argument metodě `updateDataMapping` v `GenericLayerTool`, která vyvolá překreslení mapové vrstvy. Tato metoda přijímá ještě druhý volitelný parametr `onlyStyle`, který určuje, zda se jedná o překreslení způsobené změnou dat a nebo jsou data totožná a jedná se pouze o změnu vzhledu. Tato logika je znázorněna na obrázku 7.1.



Obrázek 7.1: Třída `GenericLayerTool` na základě informace o změně dat vyvolá překreslení. Pokud není nastaven příznak, že se jedná pouze o změnu stylu, dojde nejprve k přípravě dat. Datový soubor třída obdrží od jádra a na základě předaného pole obsahujícího klíče k datovým záznamům dojde k prohledání datového souboru a získání odpovídajících hodnot. Takto získané hodnoty se opět uloží do pole a toto pole se nastaví jako atribut `workData` dané vrstvy. Následně je toto pole procházeno a na základě jeho prvků jsou vykreslovány jednotlivé symboly do mapy. V případě nastavení parametru `onlyStyle` je přeskočen krok získávání dat a pouze se použije atribut `workData` pro překreslení symbolů.

7.3.1 Přiřazování barev dle kategorií

Jak bylo popsáno v analýze a návrhu řešení, z důvodu existence uživatelů s poruchami barvocitu a zároveň neznámé povahy vizualizovaných dat je nejvhodnějším řešením poskytnout uživatelům nástroje k tomu, aby si sami mohli rozdílné kategorie dat barevně odlišit. Tento způsob třídění je implementován jako trojice (*pravidlo, hodnota, barva*). Po namapování kategorické dimenze a aplikaci třídění je vytvořeno pole těchto trojic a při přípravě dat pro vykreslení je namapovaná kategorická dimenze porovnána vzhledem k *hodnotě* za použití *pravidla* a pokud splňuje podmínku, je tomuto konkrétnímu záznamu přidělena *barva*. Tento způsob barevného třídění lze stejně jako ostatní části systému serializovat a uchovat pro pozdější použití v konfiguračním souboru. Výsledný vzhled tohoto barevného třídění je zobrazen na obrázku 7.2. Pokud uživatel nevyužívá tohoto třídění, je k vybarvení symbolů využita implicitní barva, kterou lze změnit v kódu vrstvy. Tento způsob třídění není využit u teplotní mapy, která místo toho využívá předdefinované gradienty. Tyto gradienty budou dále popisovat v sekci týkající se teplotní mapy.



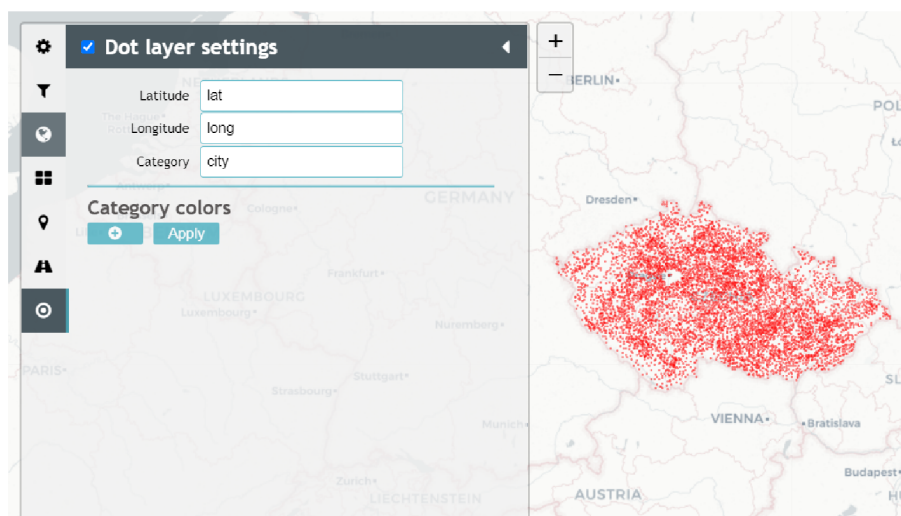
Obrázek 7.2: Demonstrace způsobu fungování barevného třídění: Na kategorickou dimenzi *city* je aplikován regulární výraz, který všechny záznamy, jejichž hodnota pod klíčem *city* nezačíná na písmena *B*, *H*, *K*, *L*, *M*, *N* nebo *P* obarví na modro. Pro zbytek záznamů je aplikována implicitní červená barva.

7.4 Dot map

V uživatelském rozhraní, které je spolu s vizualizací dat zobrazené na obrázku 7.3 tato vrstva nabízí možnost navolit si zeměpisnou šířku, zeměpisnou délku a kategorii. Ve chvíli, kdy dojde ke změně mapování se zpracují data a dojde k vykreslení značek na mapě. Tato vrstva pracuje s přiblížením a využívá služeb jádra pro registraci *event handleru* pro zoom. V reakci na přiblížení si z Leaflet mapy získá aktuální hodnotu přiblížení a znovu vykreslí všechny zobrazené symboly. V tomto případě je předán příznak informující o tom, že jde pouze o změnu stylu, použijí se tedy data, která má vrstva uložené v atributu *workData*.

Před samotným překreslením je aktuální hodnota přiblížení použita pro výpočet velikosti zobrazených teček tak, aby body byly čitelné i při vyšší úrovni přiblížení.

Pro tvorbu teček je použita vrstva `LayerGroup` knihovny Leaflet. Do této vrstvy se přidávají tečky vytvořené pomocí funkce knihovny Leaflet `circleMarker`, která vytváří vektorové kruhy. `CircleMarker` přebírá jako argumenty zeměpisnou délku a šířku, barvu, průhlednost a poloměr. Kvůli optimalizaci výkonu navíc používám parametr `renderer`, kterému předávám Leaflet `canvas`. To vede k tomu, že výsledné kruhy nejsou vykreslené jako SVG, což je při velkých datových souborech výkonostně výhodnější. Pokud si uživatel pomocí barevného klasifikátoru nenastaví jiné barvy, je výchozí barva pro tečky červená a využívají průhlednost nastavenou na hodnotu 0.4. Díky průhlednosti jsou místa s překrývajícími se značkami vykresleny barevně sytěji a lépe tak komunikují zvýšenou intenzitu v daném místě.



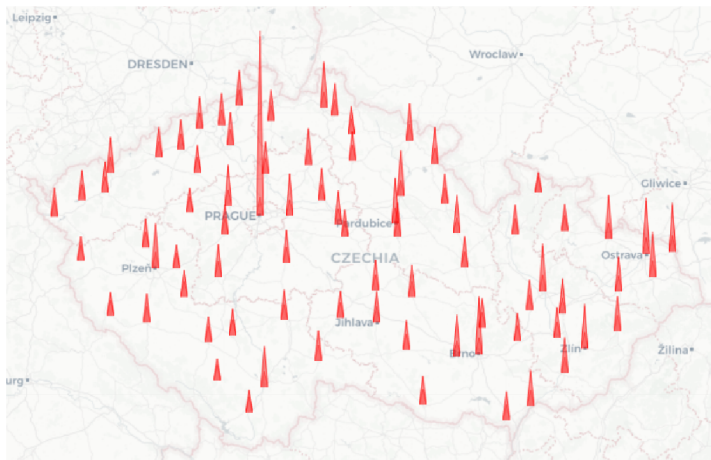
Obrázek 7.3: Výsledný vzhled uživatelského rozhraní je věrný předloze z minulé kapitoly. Průhlednost teček umožňuje lépe komunikovat vyšší výskyt sledované veličiny v místech, kde dochází k překrývání.

7.5 Spike map

Stejně jako vrstva pro *dot map* využívá tato vrstva jádro pro registraci *event handleru* pro reakci na změnu přiblížení. Uživatelské rozhraní je téměř totožné s vrstvou pro metodu teček, pouze umožňuje mapovat navíc intenzitu a způsob, jakým se budou data agregovat. Na výběr je mezi sumou hodnot a počtem hodnot. Výška bodců je škálována lineárně a je získána za pomoci funkce knihovny D3.js `scaleLinear`, která mapuje hodnoty z domény (*min*, *max*) na hodnoty v rozsahu *minHeight*, *maxHeight*. Hodnoty v doméně (*min*, *max*) jsou získány během přípravy dat z namapované intenzity, respektive uchovává se nejvyšší a nejnižší hodnota v datovém souboru.

Bodce jsou následně vykresleny za použití Leaflet funkce `marker`, které je předána ikona `SpikeIcon`. Ta je vytvořena přetížením Leaflet ikony `DivIcon`. `SpikeIcon` jako parametry obdrží vypočítanou výšku, konstantní šířku a barvu. Tyto parametry jsou použity ve funkci `iconCreateFunction`, která je součástí přetížené `DivIcon` a slouží k vytvoření ikony.

Tato funkce výsledný symbol vykreslí jako SVG element `path`. V závěru je na takto vytvořenou ikonu navázán *pop-up*, který se zobrazí po kliknutí na ikonu a obsahuje číselnou hodnotu získanou z namapované dimenze udávající intenzitu.



Obrázek 7.4: Šířka bodce je konstantní, kdežto výška je škálována lineárně na základě intenzity.

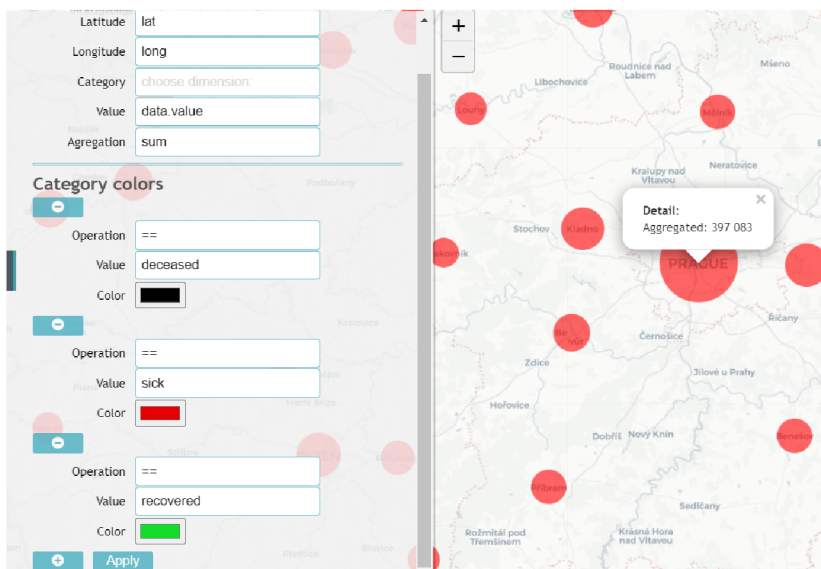
7.6 Bubble map

Tato vrstva má počet dimenzí a uživatelské rozhraní shodné s návrhem. Pro škálování velikosti symbolů je opět použita knihovna `D3.js`, ovšem oproti *spike map* je zde využita funkce `scaleSqrt`, která je určena pro škálování velikosti kruhů dle jejich plochy.

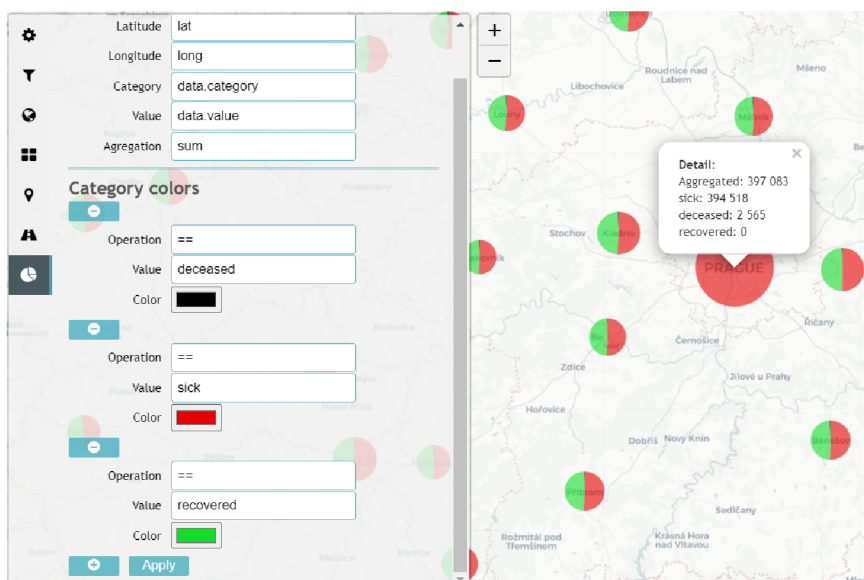
Pro shlukování symbolů v reakci na změnu úrovně přiblížení je využita volně dostupná knihovna `Leaflet.markercluster`. Ta umožňuje vytvořit vrstvu `markerClusterGroup`, u které je v případě implementace *bubble map* nastavena hodnota `maxClusterRadius`, která určuje maximální poloměr pro oblast, ve které bude provedeno shlukování. Dalším parametrem této vrstvy je funkce `iconCreateFunction`, která je automaticky zavolána ve chvíli, kdy dojde ke shlukování a nebo naopak rozpadání shluků na menší celky. Tato funkce po zpracování dat vrací ikonu `BubbleIcon`, která opět vzniká přetížením ikony `DivIcon` poskytované knihovnou `Leaflet`. Tato ikona vytváří kruh či koláčový diagram jako SVG element `path` s pomocí funkce `arc` knihovny `D3.js`.

Výsledná velikost shluku nebo samotného symbolu je dána veličinou reprezentující intenzitu. Při shlukování se tyto veličiny pro jednotlivé záznamy sčítají a velikost je tedy dána jejich součtem. Jednotlivé kategorické veličiny lze odlišit pomocí klasifikátoru barev a poměrná velikost takto zabarvených výsečí je dána zastoupením intenzity pro danou kategorii vůči celku. Implementačně je to řešeno tím způsobem, že při volání `iconCreateFunction` při změně shlukování je vytvořena celková hodnota `value`, která je dána sčítáním jednotlivých shlukovaných záznamů, přičemž každý z těchto záznamů je zároveň vložen do pole `subvalues`, kde je uložena hodnota představující jeho intenzitu a případně ještě barva, která je pro tento záznam přidělena pomocí klasifikátoru. Pro korektní fungování klasifikátoru je nutné mít namapované kategorie, nicméně vrstva je schopná pracovat i bez nich, pouze nelze zobrazené kruhy transformovat na koláčový diagram, jak je patrné na obrázku

7.5. Na ikonu je navázán *pop-up*, který zobrazuje agregovanou hodnotu a pokud jsou namapovány kategorie, tak i jejich dílčí hodnoty, což je zobrazeno na obrázku 7.6.



Obrázek 7.5: Rozpadlé shluky po kliknutí zobrazí *pop-up* s informacemi o vizualizovaném záznamu. Na tomto obrázku nejsou namapované kategorie, a proto chybí informace o jejich dílčích hodnotách. Rovněž barevné třídění nehraje v tomto případě žádnou roli.

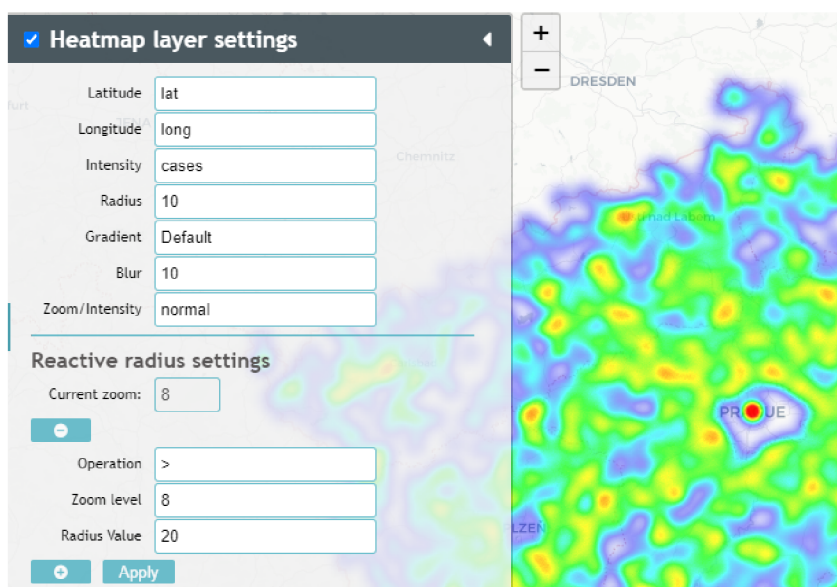


Obrázek 7.6: Při namapování kategorií lze kruh pomocí barevné klasifikace přeměnit na koláčový diagram.

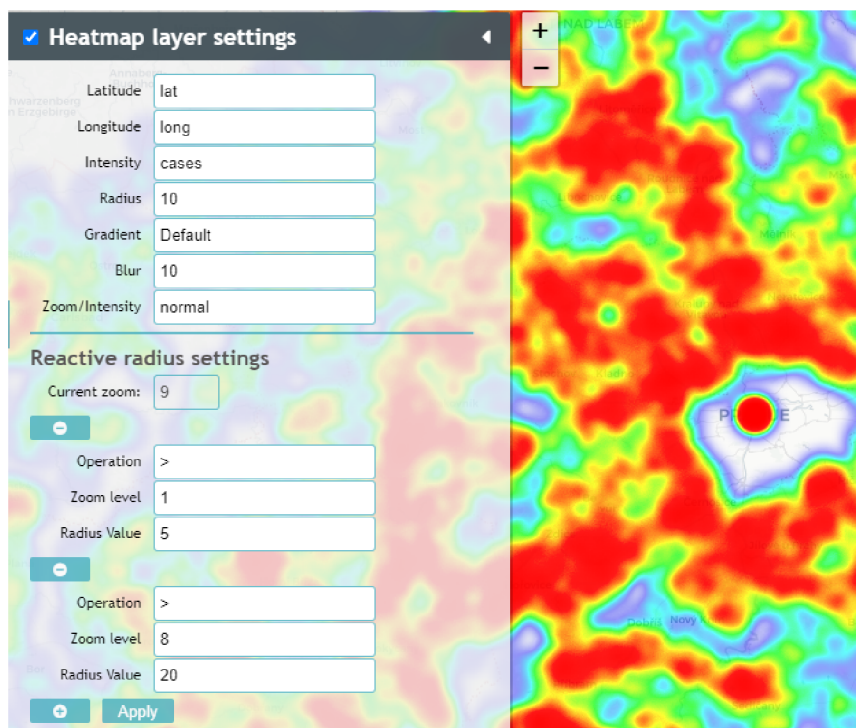
7.7 Heat map

Teplotní mapa je vytvořena pomocí volně dostupné knihovny `Leaflet.heat`. Tato knihovna umožňuje vytvořit vrstvu `heatLayer`, která jako první parametr vyžaduje dvourozměrné pole tvořené trojicemi (*latitude, longitude, intensity*). Druhým parametrem je objekt jazyka JavaScript obsahující nastavení vlastností mapy. Tento objekt obsahuje parametry, které slouží k nastavení poloměru kruhu jednotlivých záznamů, rychlost klesání intenzity od středu kruhu k okrajům, škálování zobrazené intenzity vzhledem k míře přiblížení, maximální hodnotu intenzity a použitý barevný gradient. Gradient určuje, jaké barvy jsou použity pro různé úrovně intenzity. Jedná se o objekt, který má jako klíče desetinné hodnoty od 0 (nejnižší intenzita) po 1 (nejvyšší intenzita). Hodnota identifikovaná těmito klíči je konkrétní barva, která je pro takovou intenzitu využita. Pro korektní fungování tohoto nastavení barev je nutné nastavit vrstvě `heatLayer` parametr představující maximální hodnotu intenzity na maximální hodnotu vyskytující se v datovém souboru. Mapování hodnot představujících intenzitu z datového souboru na jejich vyjádření v desetinném čísle, které určuje barvu, si zajišťuje knihovna automaticky.

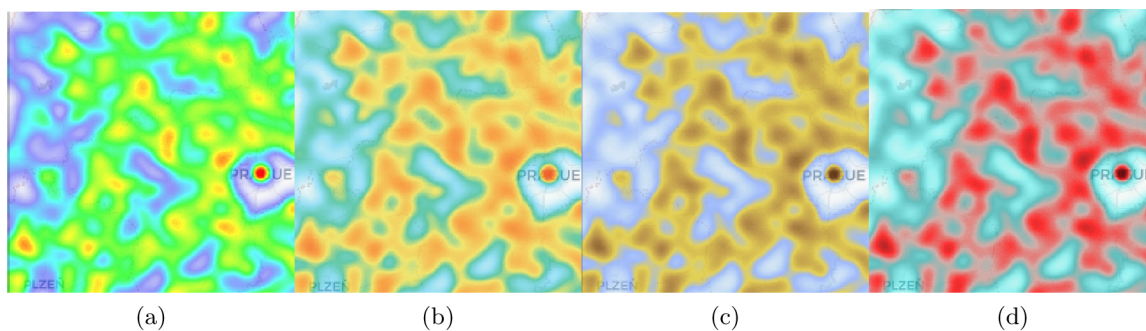
Tato vrstva nevyužívá předem definované změny velikosti zobrazených symbolů v reakci na změnu přiblížení, jako to dělají ostatní vrstvy. Místo toho umožňuje uživatelům, aby si sami nastavili, jak se bude poloměr chovat pro různé úrovně přiblížení. Využívá pro tento účel podobný způsob nastavení, který je u barevného klasifikátoru ostatních vrstev. Uživatel si zvolí operaci *Operation*, zadá hodnotu přiblížení *Zoom Level* a hodnotu poloměru *Radius Value*, která bude použita při kladném vyhodnocení tohoto pravidla. Následně je vytvořen *event listener*, který při změně přiblížení vyhodnotí, zda je na základě uživatelem vytvořených pravidel nutné změnit poloměr a překreslit mapu. Uživatelská tvorba pravidel a jejich aplikace je zobrazena na obrázcích 7.7 a 7.8. Pro snazší práci je po celou dobu v postranním panelu teplotní mapy zobrazena aktuální hodnota přiblížení.



Obrázek 7.7: Mapa používá hodnotu zadanou v poli *Radius* (v tomto případě jde o hodnotu 10), dokud hodnota přiblížení nezpůsobí kladné vyhodnocení uživatelem vytvořeného pravidla.



Obrázek 7.8: Jakmile je zadané pravidlo vyhodnoceno jako pravdivé, začne se využívat hodnota *Radius Value* zadaná v tomto pravidle. Pokud je více pravidel, které jsou vyhodnoceny jako pravdivé, používá se hodnota definovaná v posledním zadaném pravidle, které splňuje podmínku.



Obrázek 7.9: Srovnání vzhledu jednotlivých gradientů pro stejnou oblast a stejný soubor dat se stejným nastavením.

Místo barevného klasifikátoru, který používají ostatní mapové vrstvy nabízí teplotní mapa několik předdefinovaných barevných schémat. Jedná se o výchozí schéma využívající nejširší barevné spektrum v rozsahu od modré po červenou (obrázek 7.9a), dále dvě schémata pro uživatele trpící protanopií nebo deuteranopií (obrázky 7.9b a 7.9c) a jedno schéma zaměřené na uživatele trpící tritanopií (obrázek 7.9d).

Kapitola 8

Testování

V této kapitole popíši testování implementovaného rozšíření knihovny Geovisto a konzultace s uživateli. Zároveň v této kapitole popíši chyby, které byly objeveny v průběhu implementace a jejich řešení.

8.1 Testování vstupních dat

Pro testování vrstev byly použity dvě datové sady. V obou případech se jedná o data zveřejňované Ministerstvem zdravotnictví České republiky informující o šíření viru Covid-19¹. První datová sada² obsahuje přehled epidemiologické situace dle hlášení krajských hygienických stanic podle ORP³ od 1. března 2020 do 23. března 2021 po jednotlivých dnech. Pro účel testování bylo nutné data nejprve agregovat napříč časem a poté získat geografické souřadnice pro jednotlivá města a vesnice. Toho bylo dosaženo za využití *geocoding* API Nominatim⁴. Agregace dat a dotazování na API Nominatim bylo prováděno pomocí dvou separátních skriptů napsaných v jazyce JavaScript, běžících na platformě Node.js.

Druhá datová sada⁵ obsahuje kumulativní přehled nakažených, vyléčených a zemřelých pro jednotlivé okresy ke dni 13. dubna 2021. Tato data tedy nebylo nutné agregovat napříč časem, ale stačilo pouze zjistit přesné geografické souřadnice pro kódy daných okresů. Kód okresu byl nejdříve převeden na jeho název a následně byly opět pomocí skriptu a dotazování na API Nominatim zjištěny přesné koordináty.

Během vývoje jsem vrstvy testoval průběžně na výše zmíněných datových sadách. Hlavní objevenou chybou týkající se všech vrstev bylo, že původní řešení kategorizace dle barev nebo zvolený gradient nebylo možné serializovat a deserializovat, výsledné nastavení mapy tedy nebylo kompletně přenositelné. Tento problém byl vyřešen reimplementací těchto částí systému tak, aby byly uchovávány ve stavu vrstvy.

Dalším problémem, který se projevil při použití první datové sady, byla rychlost. Tato datová sada obsahuje celkem 6 251 záznamů a proto vrstvy, které jsou vykreslovány pomocí SVG vytvoří velice komplexní DOM. Řešením pro metodu teček bylo použít *renderer* využívající *canvas*. Pro mapy zobrazující proporciální symboly toto řešení bohužel není možné, protože přetěžovaná *DivIcon* specifikací proměnné *renderer* neumožňuje.

¹<https://onemocneni-aktualne.mzcr.cz/api/v2/covid-19>

²<https://onemocneni-aktualne.mzcr.cz/api/v2/covid-19/orp.min.json>

³Obec s rozšířenou působností

⁴<https://nominatim.openstreetmap.org/ui/search.html>

⁵<https://onemocneni-aktualne.mzcr.cz/api/v2/covid-19/kraj-okres-nakazeni-vyleceni-umrti.min.json>

8.2 Uživatelské testování

Testování implementované aplikace bylo rozděleno do dvou kategorií: testování základní funkcionality a testování barevných gradientů, které byly vytvořeny pro uživatele se ztrátou barvocitu. Testování základní funkcionality sestávalo z pozorování uživatelů, kteří s aplikací pracovali bez předchozího vysvětlení jednotlivých funkcionalit a následně po vysvětlení. Následně s těmito uživateli byly konzultovány jejich dojmy a připomínky. Testování barevných gradientů bylo prováděnou formou předvádění a konzultací s bravoslepými uživateli.

8.2.1 Testování základní funkcionality

Testovací vzorek sestával ze čtyř běžných uživatelů a jednoho IT profesionála. Všem byla vysvětlena práce s jednotlivými vrstvami a způsob mapování dat na grafové dimenze. Uživatelům zpočátku dělalo problém zorientovat se ve způsobu práce barevné klasifikace, nicméně po názorném předvedení ho byli schopni používat bez problému.

Na metodu teček a *spike map* reagovali uživatelé kladně a neměli k ní žádné výhrady. U teplotní mapy by ocenili možnost nastavit si, jak bude reagovat poloměr jednotlivých hodnot na přiblížení, protože teplotní mapa původně tuto funkcionality nenabízela. U *bubble map* kritizovali fakt, že při kliknutí na shluk se na tento shluk přiblíží a rozpadne se na dílčí značky nebo menší shluky, ale není možné se nijak dozvědět detailní informace o agregovaných hodnotách, které tento shluk reprezentuje.

V reakci na zpětnou vazbu uživatelů byla v postranním panelu přidána možnost reagovat na změnu přiblížení pomocí uživatelem definovaných pravidel tak, jak je to popsáno v kapitole 7. U *bubble map* byl přidán *pop-up*, který se zobrazí po najetí myši nad shluk. Kliknutí stále umožňuje rozpadnout shluk na menší celky. V rámci konzistence se i u samostatných značek *pop-up* okno zobrazí nikoliv po kliknutí, ale také po najetí myši nad značku.

S IT profesionálem byla kromě uživatelského testování konzultována architektura a filozofie knihovny Geovisto, a také způsob jejího rozšíření novými vrstvami. Kladně hodnocena byla převážně jednotná struktura, která umožňuje knihovnu, ale i jednotlivé vrstvy, snadno rozšířit či přetížít.

8.2.2 Testování gradientů barvoslepými uživateli

Pro otestování vnímání barevných gradientů pro uživatele se ztrátou barvocitu byli vybráni dva lidé s touto diagnózou. Ani jeden z nich neznal přesný název své diagnózy, nicméně pomocí nástrojů určených k simulaci barvosleposti⁶ bylo vyhodnoceno, že oba mají problém vnímat některé barvy červeného a zeleného spektra.

Testování probíhalo tím způsobem, že byly uživatelům prezentovány výřezy z teplotní mapy s použitím výchozího gradientu a na nich ukazovány dvě místa s rozdílnou intenzitou. Následně byli tázáni, zda je na obou místech stejná nebo rozdílná intenzita. Pokud bylo vyhodnoceno, že se zobrazená intenzita liší, byli následně tázáni, na kterém z těchto dvou míst je pozorovaná intenzita vyšší. Následně byly stejné experimenty prováděny pro jiné oblasti s dalšími barevnými gradienty.

Pro prvního uživatele, který (dle vlastního vyjádření) trpěl pouze slabší formou ztráty barvocitu se ukázal jako nejvhodnější výchozí gradient (zobrazený na obrázku 7.9a). Ačkoliv u něho docházelo ke zkreslení pozorovaných barev, byl schopen s perfektní precizností

⁶<https://davidmathlogic.com/colorblind/> a <https://pilestone.com/pages/color-blindness-simulator-1>

popsat změny v intenzitě ve všech místech. U gradientu zaměřeného na uživatele trpící deuteranopíí či protanopíí (obrázek 7.9c) si stěžoval na ztrátu detailu způsobenou ostrými přechody mezi modrou a hnědou barvou a také zúženým barevným spektrem.

Druhý uživatel trpěl středně těžkou ztrátou barvocitu. U porovnávání míst s velmi vysokou a velmi nízkou intenzitou neměl problém, nicméně problematické bylo odlišovat místa se středně nízkou a středně vysokou intenzitou. Při testování gradientu pro uživatele s deuteranopíí nebo protanopíí (obrázek 7.9c) byl schopen všechny místa správně určit. Volba barev pro něho nicméně nebyla ideální. Ve spolupráci s tímto uživatelem jsem sestrojil druhé schéma zaměřené na uživatele se ztrátou barvocitu červeného a zeleného spektra. Toto schéma je zobrazené na obrázku 7.9b.

Z tohoto testování vyplývá, že ztráta barvocitu je široké spektrum a bylo by vhodné vypracovat větší množství gradientů, ideálně ve spolupráci s takto omezenými jedinci. Uživatelů s tritanopíí je malé množství a pro konzultace se nepodařilo žádného sehnat a kontaktovat. Při využití simulačních nástrojů se zdá, že gradient pro tyto uživatele je navržen ve vhodných barvách, nicméně je do budoucna vhodné ho dále otestovat s uživateli, pro které je určen.

Kapitola 9

Závěr

Cílem této práce bylo vytvořit aplikaci schopnou vizualizovat generická data v geografických souřadnicích. Pro dosažení tohoto cíle bylo nutné nastudovat způsoby vizualizace dat v souřadnicích, jejich výhody, nevýhody a vhodnost pro různé druhy analýzy dat. Dále bylo nutné se seznámit s existujícími řešeními a technologiemi pro tvorbu vizualizací dat v mapách. Tato práce rozšiřuje existující knihovnu Geovisto, a proto bylo nutné podrobně nastudovat tuto knihovnu, její architekturu a filozofii a způsob, jakým pracuje s uživatelskými daty. Při analýze byly zohledněny rozdílné skupiny uživatelů, včetně těch, kteří trpí ztrátou barvocitu. Omezení těchto uživatelů bylo následně reflektováno při návrhu řešení a samotné implementaci. Výsledkem je knihovna rozšířená o hotové nástroje pro tvorbu vizualizací v geografických souřadnicích určená pro vývojáře i běžné uživatele. Těmito nástroji jsou mapové vrstvy představující teplotní mapu, *dot map*, *bubble map* a *spike map*.

Tato práce mě naučila přemýšlet při návrhu uživatelského rozhraní i nad uživateli, kteří mají jiné dovednosti a zkušenosti než já, a také jiná omezení. Jako hlavní úspěch této práce vidím kladné hodnocení ze strany uživatelů s poruchou barvocitu. Způsob vlastního dělení kategorických dimenzí dat pomocí uživatelem zvolených barev pokládali za vhodné řešení, neboť jim umožňovalo zvolit si pouze ty barvy, které jsou schopni bezpečně odlišit. Rovněž s barevnými gradienty teplotních map pro uživatele se ztrátou barvocitu byli spokojeni a pracovali s nimi efektivně.

Další možnosti pokračování vidím ve tvorbě jednotného rozhraní pro vytváření mapových legend pro již existující mapové vrstvy. Rovněž by bylo možné implementovat další mapové vrstvy, například mapu zobrazující pohyb určitého fenoménu v čase po mapě. Pro rozšíření funkcionality map zobrazujících hodnoty v geografických souřadnicích by bylo vhodné vytvořit serverové řešení, které by v reálném čase převádělo názvy nebo jiné identifikátory lidských sídel a dalších významných objektů na geografické souřadnice. To by uživateli umožnilo jednodušší práci se vstupními daty, kdy by odpadala nutnost datové soubory, které používají názvy místo souřadnic, manuálně převádět. Jako možné řešení vidím instalaci vlastní instance služby Nominatim a zároveň ukládání výsledků převodu těchto datových souborů tak, aby je bylo možné uživatelem stáhnout a příště použít. Tím by odpadla nutnost používat tento server pro stejné datové soubory vícekrát a došlo by ke snížení výpočetní i přenosové zátěže. Z hlediska optimalizace existujícího řešení by bylo zajímavé zaměřit se na vrstvy využívající SVG elementy, které při velkých datových souborech zpomalují prohlížeč. Ve spolupráci s barvoslepými uživateli by bylo možné vytvořit další barevné gradienty pro různé stupně ztráty barvocitu.

Literatura

- [1] BERNARD, H. *Social Research Methods: Qualitative and Quantitative Approaches*. SAGE Publications, 2013. ISBN 9781412978545. Dostupné z: <https://books.google.cz/books?id=7sZHuhyzBNQC>.
- [2] BUTLER, H., DALY, M., DOYLE, A., GILLIES, S., HAGEN, S. et al. *The GeoJSON Format* [online]. RFC 7946. RFC Editor, srpen 2016. Dostupné z: <https://tools.ietf.org/html/rfc7946>.
- [3] CHRISTOPHERSON, G. L. *Mapping Quantities: Proportional or Graduated Symbols* [online]. University of Arizona, říjen 2013 [cit. 2021-01-02]. Dostupné z: https://casa.arizona.edu/data/bsgist/cartography/proportional_graduated_symbols/lecture_13_text.pdf.
- [4] DEBOER, M. Understanding the Heat Map. *Cartographic Perspectives*. Nov. 2015, č. 80, s. 39–43. DOI: 10.14714/CP80.1314. Dostupné z: <https://cartographicperspectives.org/index.php/journal/article/view/cp80-deboer>.
- [5] DENT, B. *Cartography: thematic map design (5th edition)*. WCB/McGraw-Hill, 1999. ISBN 9780679384953.
- [6] ECMA INTERNATIONAL. *ECMAScript® 2022 Language Specification* [online]. Ecma International, červen 2015. Dostupné z: <https://262.ecma-international.org/6.0/>.
- [7] EKENES, K. *How and why to adjust symbol size by scale in web maps* [online]. Leden 2020 [cit. 2021-02-17]. Dostupné z: <https://www.esri.com/arcgis-blog/products/js-api-arcgis/mapping/how-and-why-to-adjust-size-by-scale-in-web-maps/>.
- [8] FLANAGAN, D. *JavaScript: The Definitive Guide: The Definitive Guide*. O'Reilly Media, 2006. ISBN 9780596554477. Dostupné z: <https://books.google.cz/books?id=2weL0iAfrEMC>.
- [9] GAO, T., HULLMAN, J., ADAR, E., HECHT, B. a DIAKOPOULOS, N. NewsViews: An Automated Pipeline for Creating Custom Geovisualizations for News. In: . Duben 2014. DOI: 10.1145/2556288.2557228.
- [10] HARRIS, R. L. *Information Graphics: A Comprehensive Illustrated Reference*. Oxford University Press, Inc., 1999. ISBN 978-0-19-513532-9.
- [11] HEYDT, M. *D3.js By Example*. Packt Publishing, 2015. ISBN 9781785280641. Dostupné z: <https://books.google.cz/books?id=Rj1lCwAAQBAJ>.

- [12] HYNEK., J., KACHLÍK., J. a RUSŇÁK., V. Geovisto: A Toolkit for Generic Geospatial Data Visualization. In: INSTICC. *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 3: IVAPP*,. SciTePress, 2021, s. 101–111. DOI: 10.5220/0010260401010111. ISBN 978-989-758-488-6.
- [13] ISO. *Glossary for ISO 3166* [online]. [cit. 2021-5-4]. Dostupné z: <https://www.iso.org/glossary-for-iso-3166.html>.
- [14] JACKSON, D., SCHEPERS, D., McCORMACK, C., DAHLSTRÖM, E., FUJISAWA, J. et al. *Scalable Vector Graphics (SVG) 1.1 (Second Edition)*. W3C Recommendation. W3C, srpen 2011. <https://www.w3.org/TR/2011/REC-SVG11-20110816/>.
- [15] JAFFE, J. *W3C and WHATWG to work together to advance the open Web platform* [online]. Květen 2019 [cit. 2021-01-19]. Dostupné z: <https://www.w3.org/blog/2019/05/w3c-and-whatwg-to-work-together-to-advance-the-open-web-platform/>.
- [16] JOHNSON, J. *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2010. ISBN 978-0-12-375030-3.
- [17] JR., T. A., ETEMAD, E. a RIVOAL, F. *CSS Snapshot 2020*. W3C Note. W3C, prosinec 2020. <https://www.w3.org/TR/2020/NOTE-css-2020-20201222/>.
- [18] KACHLÍK, J. *Grafická vizualizace geografických dat síťového provozu*. Brno, CZ, 2020. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Dostupné z: <https://www.fit.vut.cz/study/thesis/22756/>.
- [19] SARKAR, D. *The Art of Effective Visualization of Multi-dimensional Data* [online]. Leden 2018 [cit. 2021-01-01]. Dostupné z: <https://towardsdatascience.com/the-art-of-effective-visualization-of-multi-dimensional-data-6c7202990c57>.
- [20] SIMUNOVIC, M. Colour vision deficiency. *Eye*. Nature Publishing Group. 2010, sv. 24, č. 5, s. 747–755.
- [21] THROWER, N. *Maps and Civilization: Cartography in Culture and Society, Third Edition*. University of Chicago Press, 2008. ISBN 9780226799759. Dostupné z: https://books.google.cz/books?id=cGeCNa_vookC.
- [22] TUFTE, E. R. *The Visual Display of Quantitative Information*. 2. vyd. Graphics Press, 2001. ISBN 978-0-9613921-4-7.