



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

DEPARTMENT OF COMPUTER SYSTEMS

**ODHAD PRAVDEPODOBNOTI VÝSKYTU OSÔB
V OBLASTI**

ESTIMATION OF THE PROBABILITY OF OCCURRENCE OF PERSONS IN AN AREA

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

PETER KOPRDA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. VOJTĚCH MRÁZEK, Ph.D.

BRNO 2022

Zadání bakalářské práce



Student: **Koprda Peter**
Program: Informační technologie
Název: **Odhad pravděpodobnosti výskytu osob v oblasti**
Estimation of the Probability of Occurrence of Persons in an Area
Kategorie: Data mining

Zadání:

1. Seznamte se se zdroji mapových dat a jejich reprezentací.
2. Prozkoumejte možnosti dolování dat z volně dostupných zdrojů (např. katastr nemovitostí, OpenStreetMap).
3. Zpracujte studii na výše uvedená témata.
4. Navrhněte aplikaci, která na základě získaných vytvoří odhady pravděpodobnosti výskytu osob v určité oblasti.
5. Aplikaci navrženou v bodě 4 implementujte.
6. Vyhodnoťte získaná data a diskutujte možnosti dalšího zlepšení získaných dat.

Literatura:

- Dle pokynu vedoucího.

Pro udělení zápočtu za první semestr je požadováno:

- Splnění bodů 1 až 3 zadání.

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

Vedoucí práce: **Mrázek Vojtěch, Ing., Ph.D.**
Vedoucí ústavu: Sekanina Lukáš, prof. Ing., Ph.D.
Datum zadání: 1. listopadu 2021
Datum odevzdání: 11. května 2022
Datum schválení: 29. října 2021

Abstrakt

Cielom tejto bakalárskej práce je vytvoriť aplikáciu, ktorá na základe získaných mapových dát vytvorí odhady pravdepodobnosti výskytu osôb v určitej oblasti. Odhad pravdepodobnosti výskytu osôb je zobrazený pomocou tzv. heat mapy. Heat mapa je vytvorená pomocou definovaných značiek, ktoré určujú, aký počet ľudí sa nachádza v danej oblasti. Pri implementácii boli použité dáta z projektu OpenStreetMap. Vytvorená aplikácia môže slúžiť ako nástroj pri plánovaní výstavby elektroenergetických zariadení, u ktorých môže dochádzať pri poruche k ohrozeniu života, ak sa v ich blízkosti nachádzajú osoby.

Abstract

The main goal of this bachelor thesis is to implement application that creates estimation of the probability of occurrence of persons in an area from given map data. The estimation of the probability is displayed by using a heat map. Heat map is created by using defined tags which determine how many people are in the area. During the implementation map data from OpenStreetMap were used. The created application can be used during the planning of the electricity equipment construction, which can be life-threatening if people are close to the area of this equipment.

Kľúčové slová

heat mapa, odhad pravdepodobnosti, pravdepodobnosť výskytu osôb, geografický objekt, OpenStreetMap, Leaflet, GeoPandas

Keywords

heat map, estimation of probability, probability of occurrence of persons, geographical object, OpenStreetMap, Leaflet, GeoPandas

Citácia

KOPRDA, Peter. *Odhad pravdepodobnosti výskytu osôb v oblasti*. Brno, 2022. Bakalárska práca. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Vojtěch Mrázek, Ph.D.

Odhad pravdepodobnosti výskytu osôb v oblasti

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Ing. Vojtěcha Mrázeka, Ph.D. Další informace mi poskytli pán doc. Ing. David Topolánek, Ph.D., a pán Ing. Václav Vyčítal, Ph.D. Uviedol som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpal.

.....
Peter Koprda
10. mája 2022

Podakovanie

Predovšetkým by som sa rád podakoval pánovi Ing. Vojtěchovi Mrázekovi, Ph.D., za vedenie tejto bakalárskej práce, cenné rady a trpezlivosť. Takisto by som sa rád podakoval pánovi doc. Ing. Davidovi Topolánkovi, Ph.D., a pánovi Ing. Václavovi Vyčítalovi, Ph.D., za poskytnutie cenných informácií.

Obsah

1	Úvod	3
2	Mapové dáta	4
2.1	Dáta	4
2.2	Rastrové dáta	5
2.3	Vektorové dáta	6
2.4	Geografické súradnice	8
2.5	Geografický objekt	8
3	Zdroje mapových dát	10
3.1	GIS	10
3.2	OpenStreetMap	13
3.3	Kataster	16
4	Návrh	18
4.1	Analýza požiadavkov	18
4.2	Architektúra	20
4.3	Dátový model	20
4.4	Štruktúra systému	21
4.5	Funkcionalita	22
4.6	Návrh užívateľského rozhrania	23
5	Implementácia	25
5.1	Architektúra	25
5.2	Použité technológie	26
5.3	Webový server	26
5.4	Klientská časť	27
5.5	Implementácia funkcionality	31
5.6	Implementácia užívateľského rozhrania	32
6	Vyhodnotenie výsledkov	34
6.1	Ovládanie	34
6.2	Testovanie aplikácie	35
6.3	Obmedzenia aplikácie	36
7	Záver	38
	Literatúra	39

Kapitola 1

Úvod

Mapy poznáme všetci zo základnej školy. Využívali sme ich na geografiu a učili sme sa s nimi pracovať. Snažili sme sa z nich získať informácie, ktoré boli pre nás dôležité. Vydolované informácie sme upravili podľa svojho uváženia a následne sme sa ich snažili interpretovať. Pri interpretovaní takýchto dát sme vytvárali rôzne skupiny dát, ktoré spolu súviseli.

Mapa teda zjednodušene zobrazuje priestor a vyjadruje vzťahy medzi objektami v priestore. Na mapách sú zobrazené dáta v priestore, ktoré môžeme získať pomocou vhodného nástroja na zbieranie dát z mapy. Získanie mapových dát a upravením ich do podoby, v ktorých ich môžeme interpretovať na mape, je silným nástrojom reprezentácie dát. Osoba, ktorá chce zistiť informácie o oblasti, ktorú si vybral, zobrazenie dát na mape predstavuje pre túto osobu jednoduchší nástroj. Ak by získané dáta boli zobrazené v tabuľke, danej osobe by trvalo dlhšiu dobu, kým by zistil, čo reprezentujú jednotlivé hodnoty v tabuľke.

V dnešnej dobe je elektrická energia nevyhnutným prostriedkom pre náš každodenný život. Kvôli tomu je potrebná aj výstavba elektroenergetických zariadení, ktoré slúžia na výrobu, pripojenie, prenos, výrobu, distribúciu alebo dodávku elektrickej energie. Niektoré z týchto zariadení vytvárajú potenciálne nebezpečenstvo pre človeka, pretože môže u nich dôjsť k poruche. Takáto porucha môže ohroziť ľudské životy osôb, ktoré sa nachádzajú v blízkosti takýchto zariadení počas poruchy. Vytvorením odhadu pravdepodobnosti výskytu osôb pre oblasť na mape sa predídne takémuto zbytočnému ohrozovaniu ľudských životov. Cieľom tejto bakalárskej práce je vytvoriť odhad pravdepodobnosti výskytu osôb v oblasti, ktorá by slúžila pri plánovaní výstavby elektroenergetických zariadení. Takýto odhad je možné vytvoriť pomocou tzv. *heat mapy*. Pomocou *heat mapy* je možné graficky zobrazovať dáta, v ktorých je každá hodnota reprezentovaná farbou určitého farebného spektra. Vytvorená *heat mapa* tejto bakalárskej práce by mala zobrazovať odhad pravdepodobnosti v závislosti od danej značky na mape.

V kapitole 2 sa čitateľ oboznámi so základnými pojmami spojenými s mapovými dátami. V tejto kapitole zistí, aký je rozdiel medzi rastrovými a vektorovými mapovými dátami a akými rôznymi formátmi je ich možné reprezentovať. V kapitole 3 sa dozvie, z akých zdrojov je možné získať mapové dáta a ako je možné ich získať z týchto zdrojov. Kapitola 4 obsahuje popis návrhu systému, ako by mala vyzeráť štruktúra systému, návrh funkcionality a užívateľského rozhrania. Ako vyzerá architektúra systému a aké technológie sa použili pri implementovaní systému sa nachádza v kapitole 5. Okrem toho je v tejto kapitole popísaná implementácia funkcionality systému a užívateľského rozhrania. Kapitola 6 vyhodnocuje implementovaný systém, popisuje získané výsledky a diskutuje o tom, ako by sa dal vylepšiť vytvorený systém.

Kapitola 2

Mapové dáta

Mapa je používaná ako nástroj na získavanie a reprezentovanie mapových dát kvôli možnosti zobrazeniu komplexných dát. Mapové dáta, ktoré sa síce dajú uložiť do tabuľky, neponúkajú vizuálny náhľad. To je možné pomocou máp, ktoré nám umožňujú zobrazovať, ako sú dáta geograficky rozložené.

V tejto kapitole sa čitateľ stručne oboznámi s pojmom dáta, čo sú to mapové dáta, aké sú rozdiely medzi klasickými dátami a mapovými dátami, ako je možné ukladať mapové dáta a ako môžeme s nimi pracovať.

2.1 Dáta

Na to, aby bolo možné lepšie pochopiť, o čom sú mapové dáta, je potrebné vysvetliť, čo sú to dáta. Dáta sú informácie v digitálnej forme, ktoré môžu byť prenášané a spracovávané. Aby dáta mali určitú informačnú hodnotu pre užívateľa, je potrebné poskytovať dáta užívateľovi v dostatočnom množstve, na obmedzenom priestore, v obmedzenom čase a v zrozumiteľnej forme. Dáta je možné zoskupovať rôznymi spôsobmi ako napr. na vstupné a výstupné. Pre vizualizáciu rozdeľujeme dáta do dvoch základných skupín [14]:

1. **Kvantitatívne** dáta sú merateľné dáta, t.j. pomocou nich popisujeme, aká je presná hodnota meranej veličiny (teplota, tlak, ...). Keďže ich hodnota musí byť presná, užívateľ môže mať problém pochopiť význam takýchto dát.
2. **Kvalitatívne** dáta sú kategorické dáta, t.j. pomocou nich popisujeme kvalitu a vlastnosti nejakých javov (spokojnosť zákazníka, farba, ...). Je možné ich pozorovať alebo odvodiť z kvantitatívnych dát.

Dáta je možné reprezentovať graficky a textovo. Obidve reprezentácie dát majú svoje výhody aj nevýhody v závislosti od toho, čo chceme z dát zistiť. Grafická reprezentácia dát dokáže zdôrazniť vzťahy medzi viacerými hodnotami a je vhodná na zdôraznenie trendov, na druhej strane textová reprezentácia kladie dôraz na samotné hodnoty.

Multidimenzionálne dáta

Podľa [14] je dimenzia množina hodnôt určitého typu popisujúca kvantitatívne alebo kvalitatívne dáta. Danú dimenziu môže predstavovať napr. množina časov, množina miest a pod. Ak dáta majú viacero dimenzií, hovoríme o *multidimenzionálnych dátach*. Pomocou dimenzií je možné dáta triediť, kategorizovať a agregovať (zlučovať niekoľko hodnôt do jediného

čísla). Agregáčn  funkcie umo ňujú d ta agregovať r znymi sp sobmi, medzi agregáčn  funkcie patr  napr. suma, po et, priemer, medi n, minimum a maximum.

Mapov  d ta

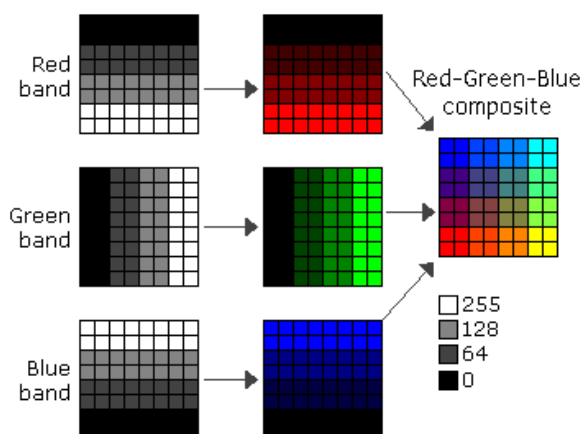
Mapov  d ta s  vlastnou kateg riou d t, pretože pridávajú do klasick ch d t nov  dimenziu s geografick mi d tami. Geografick  d ta m žu buď reprezentovať geografick  s radnice (dvojica hodn t – zemepisn  d ĺzka a širka) alebo identifikovať geografick  objekt (bod,  iara alebo polyg n). V podstate geografick  d ta s  buď v rastrovom form te alebo vo vektorom form te. V čšinou s  tieto dva form ty kombinovan , napr klad keď s  vektorov  d ta prekryt  rastrov mi d tami ziskanych zo satelitn ch sn mkov.

2.2 Rastrov  d ta

Rastrov  d ta s  d ta, ktor  s  ulo en  v mrie ke, kde ka d  bod – *pixel* – tejto mrie ky m  ur en  svoju presn  polohu, farbu a in  parametre (napr. priehľadn st). Pri ur itom pribli en  obr zka je vidno samotn  pixely, tak e obr zok vyzer  rozpixelovane.

Rastrov  p sma

Rastrov  d ta sa používajú nielen pri fote n  zemsk ho povrchu pomocou satelitov. Pixely v rastrov ch d tach nemusia byť iba vyplnen  farbou. Ako s  vyplnen  jednotliv  pixely v rastrovom modeli definuje *rastrov  p sma*. Oby ajn  fotografia m  tri p sma:  erven , zelen  a modr . Na obr zku 2.1 je mo n  vidieť ako vyzerajú tak to tri z kladn  rastrov  p sma a ako spoločne vytv raj  obr zok. Niektor  rastrov  d ta m žu mať menej rastrov ch p siem (napr. iba jedna pou it  na nadmorsk  v šku) alebo aj viac (napr. na vlnov  d ĺzky, ktor  nevid me) [7].



Obr. 2.1: Rastrov  p sma. Prevzat  z [22].

Rastrov  form ty

Pri anal ze a zobrazovan  rastrov ch d t je potrebn  tieto d ta upraviť do kompaktn šej formy. To umo ňuj  r zne rastrov  form ty, ktor  sa staraj  o dve  lohy – zbaľovanie d t

do pixelov, následné ukladanie vzťahov medzi týmito pixelmi a ich umiestňovanie na zemeguľu. Medzi najznámejšie rastrové formáty mapových dát patrí *GeoTIFF*, *JPEG2000* a *IMG* [10].

GeoTIFF¹ je štandardom pre metadáta, ktorý umožňuje vkladanie informácií o georeferenciách do obrázkového súboru. Tieto informácie obsahujú metadáta ako napr. použitá mapová projekcia, použitý súradnicový systém, dátumy, priestorové rozlíšenie a i.

JPEG2000² je systém obrázkového kódovania, ktorý používa kompresné techniky založené na diskretnej waveletovej transformácii (DWT). Ponúka rýchlejšiu a kvalitnejšiu kompresiu oproti klasickej JPEG kompresii. Tento formát je vhodný na spracovanie satelitných obrázkov.

IMG súbory používajú hierarchický formát, ktorý môže ukladať informácie napr. o súbore, o pozemných kontrolných bodoch a o type senzora. Každá vrstva rastrového obrázka je súčasťou súboru formátu IMG, ktorý obsahuje informácie o projekcii, štatistikách, atribútoch a pod.

2.3 Vektorové dáta

Vektorové dáta na rozdiel od rastrových dát obsahujú základné geometrické útvary. Výhodu oproti rastrovým dátam majú takú, že pri hocikakom priblížení nie je možné vidieť pixely. Je to tým, že vektorové dáta sú zložené z geometrických bodov a čiar, ktoré môžu byť v prípade potreby prevedené na obrázok.

Vektorové dáta môžu obsahovať viac informácií ako rastrové dáta. Zatiaľ čo rastrové dáta mohli obsahovať iba atribúty o farbe, nepriehľadnosti alebo výške, vektorové dáta môžu obsahovať informácie o tvare. Tvary vektorových dát obsahujú informácie o vlastnostiach alebo atribútoch, ako napr. počet ľudí, ktorí žijú v danej oblasti, názov mesta reprezentovaného polygónom a pod.

Vektorové formáty

Medzi najznámejšie vektorové formáty dát patrí formát *shapefile* vytvorený spoločnosťou ESRI. Tento formát sa ukladá do zariadenia ako skupina aspoň troch súborov, ktoré síce majú rovnaký názov, ale majú rôznu príponu:

- hlavný súbor *.shp – obsahuje popis geometrie každého záznamu
- indexový súbor *.shx – prepája prvok v hlavnom súbore so záznamom v atribútovej tabuľke
- databázový súbor *.dbf – databázový súbor, obsahuje dáta atribútov pre každý záznam

Okrem hore zmienených súboroch môže mať *shapefile* aj iné súbory, ktoré obsahujú ďalšie informácie o dátach:

- projekčný súbor *.prj – ukladá informácie o súradnicovom systéme
- priestorové indexy *.gix, *.sbn, *.sbx – umožňujú rýchlejšie vyhľadávanie prvkov
- atribútové indexy *.atx – urýchľujú vyhľadávanie v atribútovej tabuľke

¹<https://trac.osgeo.org/geotiff>

²<https://jpeg.org/jpeg2000>

- metadátoý súbor *.shp.xml – metadáta o zvolenom prvku
- kódovací súbor *.cpg – súbor pre správnu identifikáciu znakov

Existuje veľa iných vektorových formátov dát, ktoré sa používajú v GIS³ ako napr. *KML*, *KMZ*, *GPX*, *OSM XML*, *PBF* a *GeoJSON* [10].

KML (Keyhole Markup Language) je dátový formát vyvinutý pre aplikáciu Google Earth. Okrem ukladania informácií o geometrii obsahuje možnosti konfigurácie pre Google Earth mapy.

KMZ (Keyhole Markup Zipped) je dátový formát, ktorý sa takisto používa v aplikáciách Google. Tento formát je rozšírením KML formátu, pretože obsahuje okrem textového popisu aj obrázky tvoriace 3D vizualizácie prvku.

GPX (GPS Exchange Format) je formát údajov GPS pre ukladanie bodov, trás a ich atribútov. Ukladá informácie pomocou textu, podobne ako súbory typu KML.

OSM XML (OpenStreetMap XML) je dátový formát založený na jazyku XML⁴. Súbory tohto formátu sú zbierkou vektorových funkcií získaných od open-source komunity.

PBF (Protocolbuffer Binary Format) je efektívnejší formát OSM XML formátu. Jedná sa o binárny dátový formát, pretože veľkosť súboru vo formáte PBF je mnohonásobne menšia ako veľkosť súboru vo formáte OSM XML.

GeoJSON je formát založený na formáte JSON⁵. Tento formát je navrhnutý pre reprezentáciu jednoduchých priestorových geografických dát a ich atribútov. Pomocou tohto formátu je možné reprezentovať nasledujúce geometrické typy: *Point*, *LineString*, *Polygon*, *MultiPoint*, *MultiLineString* a *MultiPolygon* [8]. Vo výpise 2.1 je možné vidieť, ako je reprezentovaný bod (Point) vo formáte GeoJSON.

```

{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "properties": {
        "kraj": "Moravsko-sliezsky",
      },
      "geometry": {
        "type": "Point",
        "coordinates": [
          17.60100156068802,
          49.196293610455584
        ]
      }
    }
  ]
}

```

Výpis 2.1: `FeatureCollection` obsahuje vo vlastnosti `features` objekt typu `Feature`, ktorý v `properties` obsahuje informácie viazané na objekt `geometry`.

³Geografický informačný systém

⁴eXtensible Markup Language

⁵Javascript Object Notation

2.4 Geografické súradnice


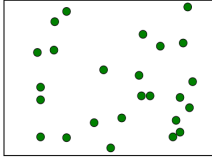

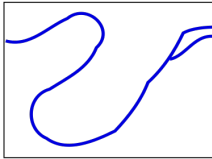

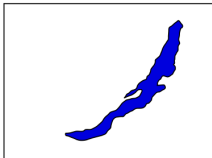
Najbežnejším spôsobom ako je možné ukladať miesta na Zemi je pomocou geografických súradníc. Geografické súradnice súhrnne označujú zemepisnú šírku a zemepisnú dĺžku. Ide o dve uhlové súradnice, ktoré sú vzťahované k Zemi alebo k jej náhradnému telesu, pomocou ktorých možno určiť ľubovoľnú polohu tohto bodu na Zemi [1]. Môžu byť reprezentované v šesťdesiatkovej sústave ako napr. $31^{\circ} 42'$, ale nový štandard umožňuje reprezentovať tieto čísla ako reálne čísla t.j. 31,7.

Zemepisná šírka je uhol medzi zvislicou miesta na zemský povrch a rovinou rovníka. Vyjadruje sa na mapách rovnobežkami. Je to hodnota v intervale $\langle -180; 180 \rangle$.

Zemepisná dĺžka je uhol medzi rovinou základného poludníka a rovinou poludníka daného bodu. Vyjadruje sa na mapách poludníkmi. Je to hodnota v intervale $\langle -90; 90 \rangle$.

2.5 Geografický objekt

Väčšinu geografických dát je možné reprezentovať tromi typmi geografických objektov: body, čiary a polygóny [9]. Na obrázku 2.2 je vidieť popis geometrických objektov a ako je možné ich využiť.

Primitive	Feature	Representation	Attributes																		
Points			<table border="1"> <thead> <tr> <th>ID</th> <th>HEIGHT</th> <th>DIAMETER</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>17.5</td> <td>35</td> </tr> <tr> <td>2</td> <td>22</td> <td>45.6</td> </tr> <tr> <td>3</td> <td>15</td> <td>27.2</td> </tr> <tr> <td>4</td> <td>19.7</td> <td>36.1</td> </tr> <tr> <td>...</td> <td></td> <td></td> </tr> </tbody> </table>	ID	HEIGHT	DIAMETER	1	17.5	35	2	22	45.6	3	15	27.2	4	19.7	36.1	...		
	ID	HEIGHT	DIAMETER																		
1	17.5	35																			
2	22	45.6																			
3	15	27.2																			
4	19.7	36.1																			
...																					
Lines			<table border="1"> <thead> <tr> <th>WIDTH</th> <th>DEPTH</th> <th>LENGTH</th> </tr> </thead> <tbody> <tr> <td>15</td> <td>4.3</td> <td>35</td> </tr> <tr> <td>6.3</td> <td>3.9</td> <td>5.2</td> </tr> </tbody> </table>	WIDTH	DEPTH	LENGTH	15	4.3	35	6.3	3.9	5.2									
	WIDTH	DEPTH	LENGTH																		
15	4.3	35																			
6.3	3.9	5.2																			
Polygons			<table border="1"> <thead> <tr> <th>AREA</th> <th>DEPTH</th> </tr> </thead> <tbody> <tr> <td>31494</td> <td>1637</td> </tr> </tbody> </table>	AREA	DEPTH	31494	1637														
AREA	DEPTH																				
31494	1637																				

Obr. 2.2: Geografické objekty, ich reprezentácia, príklady a možné atribúty. Prevzaté z [19].

Bod (point) reprezentuje jednotlivý pár súradníc X, Y. Body sú bežne používané na vizualizáciu udalostí a javov, ktoré môžu byť identifikované špeciálnymi polohami na mape (napr. supermarket, kostol).

Čiara (line) reprezentuje množinu usporiadaných bodov – uzlov, ktoré sú pospájané do čiary. Tieto geometrické útvary majú jednoznačne zadefinovaný začiatkový a koncový bod (napr. cesta, rieka).

Polygón (polygon) reprezentuje množinu usporiadaných bodov – uzlov, ktoré sú navzájom pospájané líniami do uzatvorenej plochy. Majú regionálny alebo zónový rozsah ohraničený súborom mapovateľných hraníc (napr. regióny).

Použitím bodov, čiar a polygónov môže byť geografický priestor modelovaný priradením hodnôt k týmto objektom. Každá časť mapy môže obsahovať viacero objektov. Napríklad pri vytváraní mapovej vrstvy s krajinami sveta by bolo potrebné, aby pre krajinu USA bolo vytvorených viacero polygónov (kontinentálna časť USA, Aljaška, Havajské ostrovy, a pod.). Všetky tieto polygóny vytvárajú jeden prvok, pretože sú súčasťou jednej krajiny a budú zdieľať rovnaké priradené hodnoty [19].

Pri reprezentovaní špecifického prvku alebo javu na mape nie je vždy jednoznačné, ktorý typ geografického objektu by sa mal použiť. Rovnaký geografický jav môže reprezentovať rôzne typy geografických objektov pri rôznych úrovniach priblíženia. Napríklad mesto môže byť reprezentované ako bod na mape, ale takisto môže byť reprezentované polygónom [9].

Kapitola 3

Zdroje mapových dát

Mapové dáta sa dajú získať z viacerých zdrojov. Každý takýto zdroj má svoje výhody aj nevýhody. Táto kapitola oboznámi čitateľa z akých zdrojov je možné získať mapové dáta, aké sú možné výhody a aké sú nevýhody daného zdroja, ale aj aké formáty súborov mapových dát tieto zdroje ponúkajú.

3.1 GIS

Geografický informačný systém (GIS) je informačný systém, ktorý sa využíva na získavanie, analyzovanie, vizualizáciu a manažment dát s priestorovým alebo mapovým vyjadrením. GIS spracováva geografické údaje v digitálnej podobe. Časť takýchto údajov vzniká napr. pomocou satelitných údajov, meraním pomocou polohového systému GPS alebo inými meracími prístrojmi. Údaje v papierovej podobe je nutné digitalizovať. Súčasťou GIS je hardvér (počítače, servery, zariadenia na zber dát,...), softvér (špecializované programy pre prácu s priestorovými dátami), dáta (priestorové údaje) a používatelia (spracovatelia dát, administrátori GIS a prijímatelia priestorových informácií) [19]. Používatelia systému môžu využívať rôzne metódy spracovania geografických údajov, ktoré umožňujú údaje prehľadávať, triediť, reklasifikovať, transformovať a modelovať [12].

Geografické dáta v GIS môžu byť organizované dvomi základnými modelmi – vektorovým a rastrovým modelom [13].

Vektorový model

Vektorový model je nazývaný podľa spôsobu vyjadrenia jeho jednotlivých častí – úseky kriviek s definovanou veľkosťou a smerom – *vektorom*. Vektorový model GIS pracuje s tromi variantami geografických objektov – bod, čiara a polygón.

Existujú rôzne modely dát pomocou ktorých je možné reprezentovať geografické objekty s využitím vektorovej grafiky:

- **Špagetový model** – vychádza z postupov využívaných pri digitalizácii máp. Každý objekt na mape je reprezentovaný jedným záznamom a je uložený ako reťazec X, Y súradníc. Z tohto modelu nie je možné získať žiadne informácie o vzťahoch medzi jednotlivými subjektami aj keď sa jedná o priestorové dáta.
- **Topologický model** – každá línia tohto modelu začína a končí v uzle. Všetky informácie tohto modelu sú ukladané do tzv. *topologických tabuliek* – tabuľka spojov,

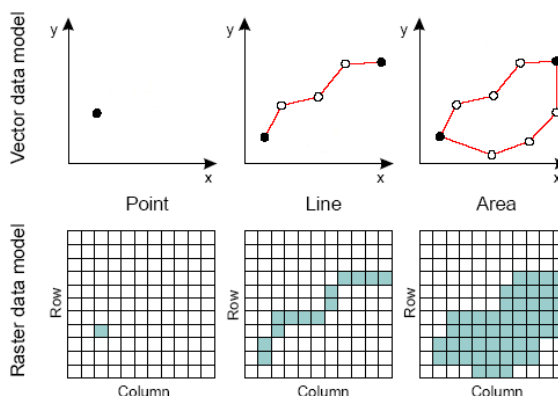
súradníc a polygónov. Model vďaka tomu uchováva priestorové vzťahy medzi objektami.

- **Hierarchický model** – ukladá zvlášť informáciu o bodoch, líniiach a plochách v hierarchickej štruktúre pre jednoduchšie vyhľadávanie v dátach. V modeli sú takisto zahrnuté aj odkazy medzi jednotlivými druhmi objektov a obsahuje topologickú informáciu. Tento model je pre manipuláciu a vyhľadávanie v dátach najvhodnejší.

Rastrový model

Rastrová reprezentácia mapových dát sa na rozdiel od vektorovej zameriava na zemský povrch. Používa sa skôr na javy ako je napr. úhrn zrážok či nadmorská výška.

Základným princípom tohto formátu je pokrytie zemského povrchu pravidelnou alebo nepravidelnou sieťou, pričom jednotka tejto siete je *bunka* (pixel, cell). Nepravidelná sieť má výhodu oproti pravidelnej sieťke takú, že pomocou nepravidelnej siete je možné jednoduchšie reprezentovať rôzne prechody z roviny na terénnu hranu. Na druhú stranu použitie nepravidelnej siete je výpočtovo a aj algoritmicky náročnejšie. Keďže v tomto modeli neexistujú objekty známe z vektorového modelu GIS, bunky definujú vlastnú hodnotu sledovaného javu v konkrétnej časti priestoru. Hodnota v jednej bunke odpovedá bodu, rada spojených buniek s rovnakou hodnotou odpovedá línii a skupina navzájom susediacich buniek odpovedá ploche.



Obr. 3.1: Vektorový a rastrový dátový model. Prevzaté z [16].

ArcGIS

V súčasnosti existujú rôznorodé softvéry pre GIS. Medzi najznámejšie patrí ArcGIS od spoločnosti ESRI, ktorý slúži na mapovanie a priestorovú analýzu navrhnutý tak, aby podporoval poslanie a obchodné ciele organizácií. Tento systém poskytuje tri úrovne licencií. Typ zvolenej licencie rozhoduje o tom, ako sú uložené dáta a ako je možné ich editovať.

Medzi najznámejšie aplikácie systému ArcGIS patrí ArcMap (použiteľná na priestorové analýzy, editáciu dát a tvorbu kartografických výstupov), ArcCatalog (pomáha organizovať a spravovať všetky dáta), ArcGIS Explorer (voľne dostupný prehliadač priestorových dát), ArcGIS for Server (serverové riešenie pre GIS, umožňuje jednoduchú konfiguráciu webových aplikácií a poskytuje kompletné vývojárske prostredie pre *.NET* a *Java*).

ArcČR 500

ArcČR 500 [3] je digitálna vektorová geografická databáza Českej republiky, spracovaná na úrovni podrobnosti 1 : 500 000. Obsahom databázy sú prehľadné geografické informácie o ČR. Zdrojom dát pre geografické dáta ArcČR 500 v 3.3 je databáza Data200, čo je národná vektorová geografická databáza Zeměměřického úřadu (ZÚ) odpovedajúca presnosťou a stupňom generalizácie 1 : 200 000. Vstupné dáta z Data200 majú deklarovanú absolútnu presnosť do 100 m. Absolútna polohová odchýlka ArcČR 500 v 3.3 je odhadovaná do 200 m.

Dáta sú uchovávané iba v GIS formátoch firmy ESRI a to vo formáte súborovej databázy. ArcČR 500 je zložená z dvoch geodatabází – geografické prvky a administratívne členenie.

Geografické prvky ArcČR 500 boli odvodené zo 17-tich vrstiev databázy Data200. Vrstvy súborovej databázy ArcCR500_v33.gdb ako aj ich popis a typ prvkov je možné vidieť v tabuľke 3.1.

Tabuľka 3.1: Súborová databáza ArcCR500_v33.gdb. Prevzaté z [3].

vrstva	popis	typ prvku
Letiste	Letisko	bod
SidlaBody	Sídla nad 500 obyvateľov	bod
VyskoveKoty	Výškové kóty (vrcholy kopcov)	bod
ZeleznicniStanice	Železničná stanica	bod
Hranice	Štátna, krajská a okresná hranica	lína
Silnice	Cesta	lína
VodniToky	Vodné toky	lína
Vrstevnice	Vrstevnice po 25 m	lína
Zeleznice	Železnice	lína
BazinyARaseliniste	Močiar a rašelinisko väčšie ako 30 ha	polygón
Lesy	Lesné plochy väčšie ako 30 ha	polygón
SidlaPlochy	Sídla nad 5 000 obyvateľov	polygón
VodniPlochy	Vodné plochy väčšie ako 15 ha	polygón
ChranenaUzemi	Národné parky a chránené krajinné oblasti	polygón
KladyZakladnichMap	Klady základných máp ČR	polygón
KladyTopografickýchMap	Klady vojenských topografických máp	polygón
SouradnicovaSitJTSK	Súradnicová sieť systému JTSK v intervale 1 km	lína
ZemepisnaSitETRS89	Zemepisná sieť v systéme ETR89	lína
ZemepisnaSitWGS84	Zemepisná sieť v systéme WGS84	lína
DigitalniModelReliefu	Raster digitálneho modelu reliéfu	raster
StinovanyRelief	Raster tieňovaného modelu reliéfu	raster

Každá vrstva tejto databázy môže nadobúdať rôzne hodnoty atribútov, ktoré sú pre danú vrstvu zadané. V tejto práci sú uvedené pre ilustráciu iba vrstvy Letisko (tabuľka 3.2) a Hranica (tabuľka 3.3). V tabuľke 3.2 je možné vidieť, že atribút TYP môže nadobúdať tri hodnoty – t.j. letisko je buď civilné alebo vojenské, alebo civilné a vojenské. Táto vrstva je zobrazená na mape ako bod. Na druhú stranu vrstva *Hranica*, ktorá sa zobrazuje na mape ako línia, nemá takú variabilitu atribútov ako vrstva *Letisko*. V tabuľke 3.3 je

možné vidieť, že daná vrstva má iba jeden zadaný atribút, ktorý ale môže nadobúdať 3 hodnoty – t.j. hranica môže byť štátna, krajská alebo okresná.

Tabuľka 3.2: Vrstva Letisko (Letiste). Prevzaté z [3].

meno atribútu	popis	nadobúdané hodnoty
TYP	Typ letiska	1 - civilné 2 - vojenské 3 - civilné a vojenské
NAZEV	Meno	<i>konkrétne meno</i>
NAZEV_ASCII	Meno (ASCII formát)	<i>konkrétne meno</i>
ICAO	Kód ICAO	<i>konkrétny kód</i>
STATUT	Statut letiska	1 - medzinárodné 2 - vnútroštátne

Tabuľka 3.3: Vrstva Hranica (Hranice). Prevzaté z [3].

meno atribútu	popis	nadobúdané hodnoty
TYP	Typ hranice	1 - štátna 2 - krajská 3 - okresná

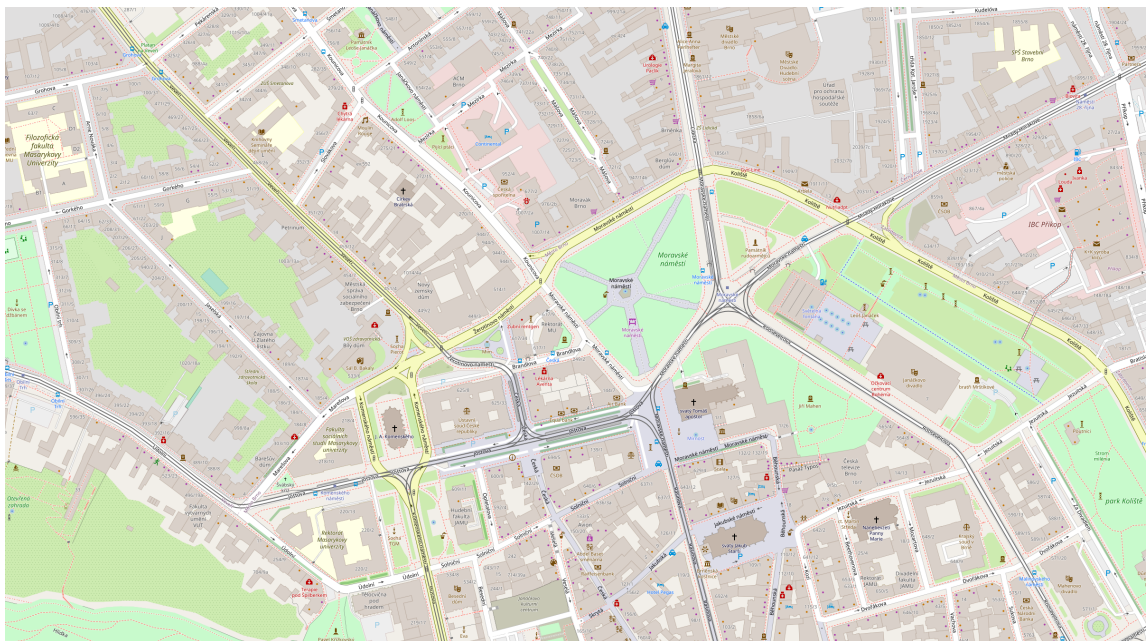
Dáta z Českého statistického úradu (ČSÚ) boli použité pre tvorbu dát **administratívneho členenia**. Vrstvy súborovej geodatabázy `AdministrativniCleneni_v13.gdb` ako aj ich popis sa nachádza v tabuľke 3.4.

Tabuľka 3.4: Súborová databáza `AdministrativniCleneni_v13.gdb`. Prevzaté z [3].

názov	popis	typ prvku
ZSJ	Základné sídelné jednotky	bod/polygón
UTJ	Územné technické jednotky	bod/polygón
KU	Katastrálne územie	bod/polygón
MOaMC	Mestské obvody a mestské časti	bod/polygón
COB	Časti obce	bod/polygón
OBCE	Obce a vojenské újazdy	bod/polygón
POU	Obce s povereným úradom	bod/polygón
ORP	Obce s rozšírenou pôsobnosťou	bod/polygón
OKRESY	Okresy	bod/polygón
KRAJE	Kraje	bod/polygón
STAT	Štát	bod/polygón

3.2 OpenStreetMap

OpenStreetMap [21] je projekt, ktorý vznikol za účelom vytvárania geografickej databázy celého sveta. Cieľom tohto projektu je mať časom záznam o každom geografickom prvku na planéte. Zatiaľ čo to začalo mapovaním ulíc, postupom času tento projekt zahŕňa chodníky, budovy, vodné cesty, potrubia, lesy, pláže, poštové schránky a dokonca aj jednotlivé stromy.



Obr. 3.2: OpenStreetMap. Výsek z mapy časti Brna.

História projektu

Projekt OpenStreetMap má svoj začiatok v auguste v roku 2004, kedy britský programátor Steve Coast experimentoval s USB GPS prijímačom. Použil softvér nazývaný GPSTDrive, ktorý bral mapy z Microsoft MapPoint, ale porušoval licenčné podmienky. Coast vedome nechcel porušovať autorské práva týchto máp, preto hľadal alternatívu, ktorá by neporušovala licenčné podmienky, tú ale nenašiel. Zistil, že neexistujú zdroje mapových dát, ktoré by mohol používať v otvorenom softvéri bez toho, aby porušoval licenčné podmienky alebo platil obrovské sumy peňazí. Po odprezentovaní jeho nápadu o vytvorení vlastnej mapy na konferencii otvorených softvérov v Londýne zistil, že viacerí ľudia mali podobný nápad alebo ich Coastov nápad zaujal, a tak vznikla skupina OpenStreetMap.

V začiatkoch bol dátový model príliš simplistický, pretože obsahoval iba jednoduché čiary nakreslené cez informácie Landsat od NASA. V marci v roku 2006 bola vytvorená prvá editovacia aplikácia pre OpenStreetMap – JOSM¹. Po chvíli bola v tomto roku vytvorená prvá plnofarebná mapa mesta Weybridge. V máji toho roku sa usporiadala prvá spoločná akcia, na ktorej bolo úlohou zmapovať ostrov Wight. Bolo to prvýkrát, kedy sa stretlo viacero mapovačov a znamenalo to pre nich prelomový bod projektu, pretože bola vytvorená detailná mapa. Takéto akcie OpenStreetMap komunity sa začali konať častejšie a boli usporadúvané po celom svete.

V auguste v roku 2006 bola vytvorená nadácia *OpenStreetMap Foundation*, ktorou úlohu je podporovať, ale nie kontrolovať OpenStreetMap projekt. Venuje sa podpore rastu, rozvoja a distribúcii voľne dostupných geografických dát a poskytovaniu geografických údajov komukoľvek na používanie a zdieľanie.

Serverový softvér bol pôvodne napísaný v programovacom jazyku Java, ale v máji v roku 2007 bola implementácia softvéru prepísaná do platformy Ruby on Rails². Časom ako začal

¹<https://josm.openstreetmap.de>

²<https://rubyonrails.org>

projekt postupne narastať, začali svojimi dátami prispievať súkromné spoločnosti, mestá, ale aj štáty [6]. Vo februári v roku 2008 bolo zaregistrovaných 25 000 užívateľov na stránke OpenStreetMap, v marci v roku 2009 to už bolo 100 000 užívateľov. Počet užívateľov stále rastie, v čase písania tohto textu je to už viac ako 8,3 milióna zaregistrovaných užívateľov.

Dátový model OpenStreetMap

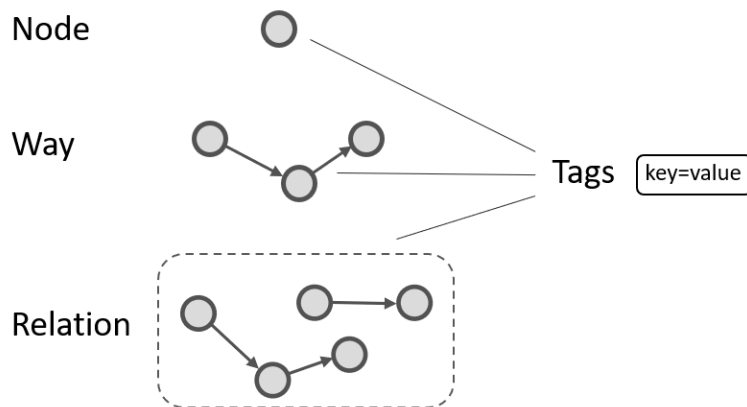
Prvky (anglicky elements) sú základnou stavebnou súčasťou dátového modelu OpenStreetMap slúžiace k popisu reálneho sveta. Medzi základné prvky OpenStreetMap projektu patria uzol, cesta a relácia. Tieto prvky je možné popísať značkami.

Uzol³ (anglicky node) označuje konkrétny bod na povrchu Zeme, je určený svojou zemepisnou šírkou a dĺžkou. Skladá sa minimálne z dvojice súradníc a svojho jednoznačného identifikačného čísla (id).

Cesta⁴ (anglicky way) je usporiadaný zoznam 2 až 2 000 uzlov, ktoré definujú lomenú čiaru. Cesta má aspoň jednu značku alebo je vložená do relácie.

Relácia⁵ (anglicky relation) sa skladá z jednej alebo viacerých značiek a usporiadaného zoznamu jedného alebo viacerých uzlov alebo ciest. Každý prvok relácie je tzv. člen (anglicky member). Používa sa k popisu závislosti medzi rôznymi prvkami. Každý člen relácie môže voliteľne mať nejakú rolu, ktorá popisuje jeho význam v rámci relácie.

Značka⁶ (anglicky tag) sa skladá z kľúča a hodnoty. Každá značka popisuje určitú vlastnosť dátových prvkov (uzlov, ciest a relácií) alebo sadu zmien. Kľúč popisuje tému, kategóriu alebo typ mapového prvku (napr. cesta – *highway* alebo názvy – *name*). Hodnota konkretizuje vlastnosť, ktorú všeobecne popisuje kľúč. Napríklad značka `highway=residential` predstavuje cestu, ktorá vedie obytnou oblasťou.



Obr. 3.3: Dátový model OpenStreetMap. Prevzaté z [15].

Sťahovanie dát

Sťahovať mapové dáta z databázy OpenStreetMap je možné viacerými spôsobmi. Vhodná voľba sťahovania dát závisí od toho, aké veľké množstvo dát sa rozhodneme sťahovať.

³<https://wiki.openstreetmap.org/wiki/Node>

⁴<https://wiki.openstreetmap.org/wiki/Way>

⁵<https://wiki.openstreetmap.org/wiki/Relation>

⁶<https://wiki.openstreetmap.org/wiki/Tags>

Pre veľké oblasti je vhodnou voľbou si stiahnuť dáta lokálne. Stiahnuť databázu celého sveta je možné na stránke Planet.osm⁷. Z tejto stránky je možné stiahnuť databázu celého sveta vo formáte OSM XML alebo PBF. Táto databáza je aktualizovaná každý týždeň a aktuálne (1. 5. 2022) veľkosť súboru vo formáte OSM XML má 115 GB a vo formáte PBF má veľkosť 63 GB.

Pre menšie oblasti je tiež možné použiť hore zmienené nástroje, avšak okrem toho je možné vybrať oblasť pomocou ohraničeného boxu, ktorý je zložený z minimálnej a maximálnej zemepisnej dĺžky a zemepisnej šírky. Pomocou týchto hodnôt je možné vytvoriť URL s HTTP API požiadavkou. Ohraničený box v URL je vyjadrený ako štyri hodnoty za sebou, oddelené čiarkami. Vo výpise 3.1 je možné vidieť príklad URL s HTTP API požiadavkou. Hodnoty v parametri `bbox` sú zoradené v poradí: minimálna zemepisná šírka, minimálna zemepisná dĺžka, maximálna zemepisná šírka, maximálna zemepisná dĺžka.

```
https://api.openstreetmap.org/api/0.6/map?bbox=16.61,49.22,16.63,49.23
```

Výpis 3.1: Príklad HTTP API požiadavku.

Ďalšou možnosťou je použiť stránku Geofabrik⁸, ktorá umožňuje sťahovať väčšie aj menšie oblasti podľa kontinentov, štátov, provincií alebo miest. Stiahnuté súbory z tejto stránky môžu byť vo formáte `osm.pbf`, `shp.zip` alebo `osm.bz2`.

3.3 Kataster

Kataster predstavuje register resp. zoznam a používa sa vo viacerých významoch:

- súpis nehnuteľného majetku, ktorý bol vytvorený pre daňové a úradné účely
- kataster nehnuteľností, katastrálna kniha
- katastrálne územie

Kataster nehnuteľností

Súčasný kataster nehnuteľností [5] na území Českej republiky mal ekvivalent medzi svojimi právnymi predchodcami – tzv. pozemkovú knihu, ktorá slúžila na majetkoprávne účely. Okrem pozemkovej knihy sa používal aj pozemkový kataster, ktorý slúžil na daňové účely. Pozemkový kataster sa používal približne do roku 1957 a zapisovanie do pozemkovej knihy bol ukončený v roku 1964, pretože nevyhovovala o skutočných a aktuálnych právnych vzťahoch k nehnuteľnostiam. Preto bola vytvorená nová pozemková evidencia nehnuteľností, ktorej hlavným účelom bolo zaistiť podklady pre plánovanie národného hospodárstva. Táto evidencia bola vedená až do roku 1992, kedy si spoločenské zmeny vyžiadali založenie dnešného katastru nehnuteľností.

Dnešný kataster nehnuteľností plní úlohu pozemkovej knihy aj pozemkového katastru a slúži aj ako podklad pre geografické informačné systémy. Kataster nehnuteľností je definovaný ako súbor údajov o nehnuteľnosti v Českej republike. Okrem súpisu a popisu nehnuteľností zahŕňa ich geometrické a polohové určenie pre jednu katastrálnu obec alebo katastrálne územie. Aj keď v českom katastri sú zapísané všetky pozemky, zo stavieb sú zapísané v katastri len tie budovy, ktoré stanovuje katastrálny zákon. Ide o budovy s popisným

⁷<https://planet.openstreetmap.org>

⁸<https://download.geofabrik.de>

či evidenčným číslom alebo hlavnú budovu v rámci areálu nehnuteľností. Na obrázku 3.4 je možné vidieť ako vyzerá katastrálna mapa mestskej časti Brno-Královo Pole. Obrázok je exportovaný zo stránky Českého katastru nemovitostí⁹, ktorý okrem faktických a právnych informácií o nehnuteľnostiach obsahuje katastrálne mapy a informácie o vlastníkoch nehnuteľností.



Obr. 3.4: Katastrálna mapa s ortofotomapou mestskej časti Brno-Královo Pole. Ružová farba predstavuje budovy, budovy so zelenou farbou sú nehnuteľnosti s cenovými údajmi k jednotke (byty alebo nebytové priestory), vyšrafované budovy sú nehnuteľnosti s cenovými údajmi k parcele.

Územný plán

Vlastníci katastrov nehnuteľností si musia byť istý, že na danom území existuje poriadok, ktorý by mal zaručovať, že ich práva nebudú ohrozené náhodnými a meniacimi sa rozhodnutiami. Územným plánovaním sa predchádza nekonštruktívnemu a zložitému rozvoju obce. Zaoberá sa všetkými aspektmi nášho prostredia. Ide prevažne o stavbu sídiel, dopravnú a technickú infraštruktúru, ale aj o prvky, ktoré vytvárajú prírodné zložky životného prostredia [2].

⁹<https://nahliznidokn.cuzk.cz/VyberKatastrMapa.aspx>

Kapitola 4

Návrh

V tejto kapitole je popísaný návrh systému, na ktorom je založená celá aplikácia. Na začiatku tejto kapitoly sa analyzujú požiadavky na aplikáciu a popisujú kľúčové vlastnosti, ktoré musia byť splnené. Táto kapitola zahŕňa aj bližšie popísanie použitých technológií a dôvody, prečo boli zvolené dané technológie pri návrhu aplikácie.

4.1 Analýza požiadavkov

Požiadavky sú potrebné pri akomkoľvek návrhu aplikácie. S požiadavkami sa treba zamyslieť, ako sa dajú navrhnúť a implementovať. Primárnym cieľom tejto bakalárskej práce je vytvoriť aplikáciu, ktorá na základe získaných dát vytvorí odhady pravdepodobnosti výskytu osôb. Túto aplikáciu je možné vytvoriť tak, aby ju bolo možné v budúcnosti rozšíriť o viaceré funkcionality, ale aj aby bola užívateľsky prívetivá.

Prvou požiadavkou pri vytvorení aplikácie je spracovanie vstupných dát získaných od užívateľa. Vstupné dáta je možné rozdeliť na dva typy: súradnice a hodnoty pravdepodobností. Druhou požiadavkou je vytváranie a zobrazovanie heat vrstvy na mape.

Vstupné súradnice

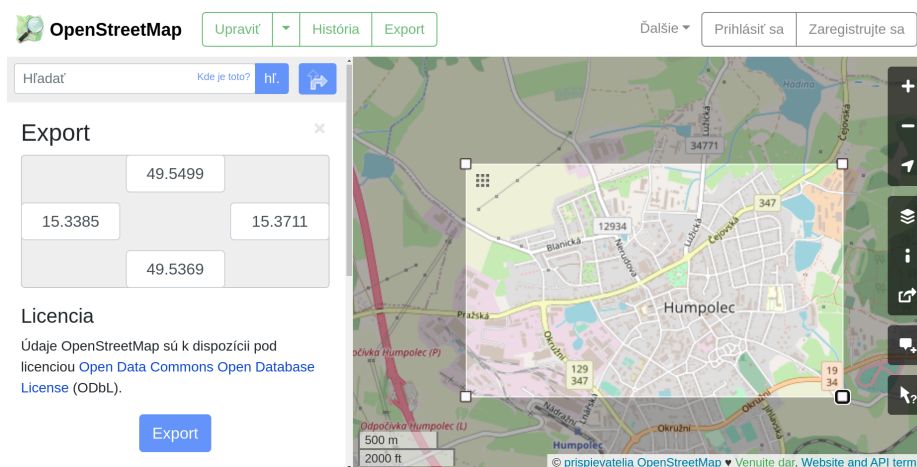
Prvým typom vstupných dát sú súradnice, ktoré užívateľ bude môcť zadať rôznymi spôsobmi. Spôsoby získavania vstupných súradníc budú inšpirované získavaním hraničných súradníc na stránke projektu OpenStreetMap.

Jednou z možností je využitie knižnice `Leaflet.draw`. Užívateľ v navrhovanej aplikácii by mal byť schopný vykresliť na mapu plochu, z ktorej by chcel zistiť, aká je pravdepodobnosť výskytu osôb v danej oblasti. Z tejto nakreslenej plochy sa dá zistiť, aké sú hranice danej plochy, t.j. aká je minimálna a maximálna hodnota zemepisnej dĺžky a aká je minimálna a maximálna hodnota zemepisnej šírky nakreslenej plochy. Tento spôsob získavania vstupných dát je zobrazený na obrázku 4.1.

Druhá možnosť získavania vstupných súradníc, takisto používaná na stránke projektu OpenStreetMap, by mohla fungovať na základe získavania hraníc mapy, ktorá je na stránke zobrazená. To znamená, že horná a dolná hranica mapy predstavujú zemepisné dĺžky, ľavá a pravá hranica mapy predstavujú zemepisné šírky.

Tretiu možnosť získavania vstupných súradníc je možné použiť v kombinácii s už zmienenými možnosťami. Užívateľ by mal byť schopný zadať všetky štyri hraničné súradnice manuálne. Pri návrhu takejto možnosti je potrebné myslieť na to, že súradnica hornej (se-

vernej) hranice musí byť väčšia ako súradnica dolnej (južnej) hranice a súradnica pravej (východnej) hranice musí byť väčšia ako súradnica ľavej (západnej) hranice.



Obr. 4.1: Export dát na stránke projektu OpenStreetMap.

Hodnoty pravdepodobností

Druhým typom vstupných dát od užívateľa sú hodnoty pravdepodobností resp. intenzít pre značky. Vo väčšine prípadoch počet týchto vstupných hodnôt pravdepodobností môže byť aj viac ako 100, preto vhodným návrhom pre zadávanie hodnôt je použitie externého súboru. V tomto súbore by mal užívateľ mať možnosť zadať, ktoré značky by sa mali použiť pri vykresľovaní heat vrstvy a aké hodnoty pravdepodobností by mali mať tieto značky.

Zobrazovanie heat vrstvy

Získané hodnoty pravdepodobností značiek by mali byť pre užívateľa informatívne a užívateľ by mal byť schopný vidieť z danej oblasti, aká je pravdepodobnosť výskytu osôb v danej oblasti. Vhodnou voľbou pre zobrazovanie pravdepodobností by bolo vytvorenie heat vrstvy na mape, ktorá bude interpretovať hodnoty pravdepodobností v danej oblasti.

Hodnoty pravdepodobností je možné interpretovať základnými farebnými schémami. Celé farebné spektrum sa skladá z kombinácií troch farieb (červená, zelená a modrá). Teplé farby sa väčšinou spájajú s ohňom a slnkom, preto sem patria farby ako červená, oranžová a žltá. Keď si ale predstavíme zasnežené hory, farebná škála sa ihneď zmení na studenú, kde prevládajú farby ako modrá, zelená a fialová. V navrhovanej aplikácii by sa dali využiť tieto farebné odtiene na zobrazovanie pravdepodobností. V oblastiach s väčšou pravdepodobnosťou výskytu osôb by aplikácia mala zobrazovať teplejšie farby a na druhej strane v oblastiach s menšou pravdepodobnosťou by aplikácia mala zobrazovať studenšie farby. V oblastiach, kde sú hodnoty pravdepodobností pre dané značky rovné 0 by nemala byť na mapu vykreslená v danej oblasti heat vrstva.

4.2 Architektúra

Cieľom bakalárskej práce je vytvorenie aplikácie, ktorá odhadne pravdepodobnosť výskytu osôb v danej oblasti. Pomocou získaných požiadavkov na aplikáciu je možné vytvoriť architektúru danej aplikácie. Aplikácia bude musieť vedieť získavať hraničné súradnice z oblasti, ktorú vyberie užívateľ. Horná a spodná hranica danej oblasti budú reprezentovať zemepisnú dĺžku, ľavá a pravá hranica budú reprezentovať zemepisnú šírku. V rámci modifikácie objektov by mal byť užívateľ schopný upravovať tvar nakreslených obdĺžnikov. Pri zmenení veľkosti obdĺžnika alebo presunutia obdĺžnika na inú časť mapy by mala aplikácia zarea-govať a zobrazíť tieto upravené hodnoty súradníc v grafickom užívateľskom rozhraní. Tieto štyri súradnice by mala aplikácia použiť na získanie dát o geografických objektoch (bodoch, čiarach a polygónoch). Získané geografické objekty by malo byť možné použiť pri vytváraní heat bodov na vytvorenie heat vrstvy.

4.3 Dátový model

Dátový model reprezentuje dáta, s ktorými bude aplikácia pracovať. Dáta, ktoré je možné získať pri spracovávaní mapových dát, sú geografické objekty. Informácie o získaných geografických objektoch a ich priradenie ku značkám na mape bude aplikácia uchovávať vo svojej pamäti vo forme dátových rámcov.

Dátový rámec

Dátový rámec je dvoj-dimenzionálna dátová štruktúra, podobná tabuľke s riadkami a stĺpcami. Každý riadok dátového rámca reprezentuje jeden geografický objekt. Definícia geometrického objektu sa bude nachádzať v stĺpci *geometry*. Okrem definície geografického objektu sa v tomto riadku bude nachádzať aj to, akú značku na mape predstavuje daný geografický objekt.

Geografický objekt

Geografický objekt, ktorý sa bude nachádzať v stĺpci *geometry* dátového rámca, môže byť jedným z hodnôt: *Point*, *LineString* alebo *Polygon*. Tieto typy predstavujú základné geometrické objekty. Typy, ktoré rozširujú tieto základné typy, sú rozšírenými geometrickými objektami. Sú od nich rozlíšené predponou **Multi**, t.j. *MultiPoint*, *MultiLineString* a *MultiPolygon*. Formát súradníc závisí od typu geografického objektu.

- **Point** predstavuje pole dvoch číselných hodnôt, ktoré reprezentujú x-ovú a y-ovú súradnicu (viď. 4.1).

`coordinates: [40.3, 29.8]`

Výpis 4.1: Príklad súradníc typu *Point*.

- **LineString** predstavuje pole bodov, kde prvý bod je začiatočným bodom a posledný bod je koncovým bodom čiary (viď. 4.2).

`coordinates: [[20.5, 30.1], [21.8, 31.0]]`

Výpis 4.2: Príklad súradníc typu *LineString*.

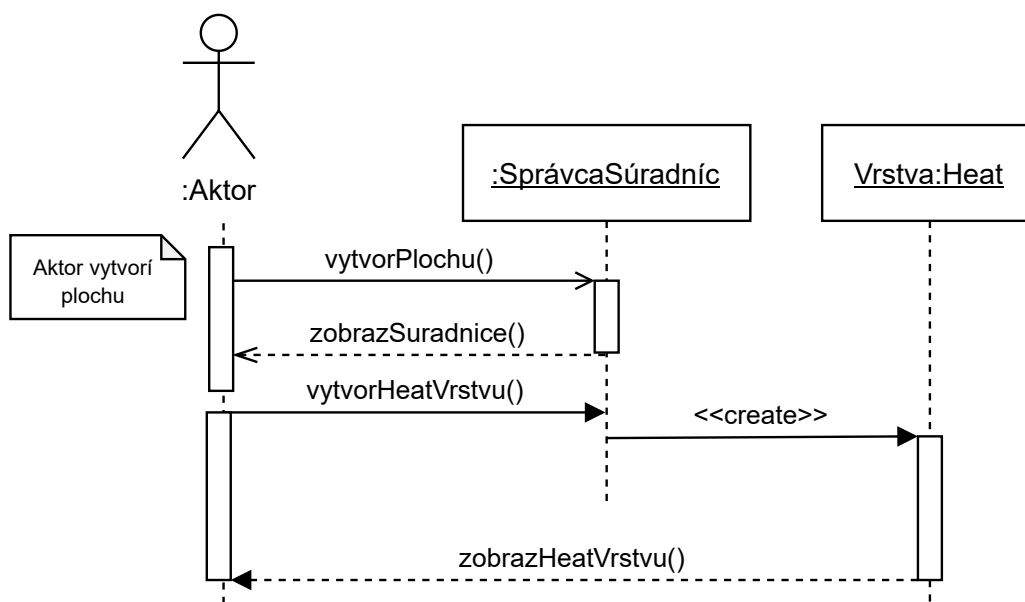
- **Polygon** predstavuje pole bodov, ako u typu *LineString* s tým rozdielom, že body sú pospájané do mnohouholníka. Prvý bod tohto objektu musí byť aj koncovým bodom. Vo výpise 4.3 je možné vidieť, ako je potrebné zdefinovať trojuholník.

```
coordinates: [[20.0, 20.0], [21.9, 21.9], [20.0, 22.8], [20.0, 20.0]]
```

Výpis 4.3: Príklad súradníc typu *Polygon*.

4.4 Štruktúra systému

Pri tvorbe návrhu sa vychádzalo z analýzy požiadaviek v sekcii 4.1. Pri zadávaní vstupných súradníc sa bude môcť užívateľ rozhodnúť, či chce na mapu nakresliť obdĺžnik, z ktorého bude vytvorená heat vrstva na mape alebo či zadá hraničné súradnice textového poľa. Ak užívateľ nakreslí na mapu obdĺžnik (resp. zobrazí si na mape určitú časť), systém spracuje tento nakreslený objekt na mape a použitím asynchrónnej správy zobrazí tieto súradnice vo formulári, kde ich bude môcť užívateľ prípadne upraviť. Takúto úpravu by mal užívateľ byť schopný vykonať veľakrát. Takisto by mohol užívateľ upravovať veľkosť aj pozíciu nakresleného obdĺžnika. Okrem toho by mohol užívateľ odstraňovať nakreslenú plochu. Ak budú zadané súradnice od užívateľa, tak potom by bolo možné vytvoriť heat vrstvu. Užívateľ jednoducho bude môcť poslať požiadavku na server, kde sa spracuje táto požiadavka a server pošle užívateľovi po určitej dobe trvania mapu s vytvorenou heat vrstvou. Všeobecnejší návrh, ako by mohla aplikácia fungovať, je zobrazený na obrázku 4.2.



Obr. 4.2: Sekvenčný diagram, ktorý zobrazuje, v akom poradí budú vykonávané jednotlivé operácie.

Aby systém fungoval správne, bude potrebné brať vstupné údaje o pravdepodobnostiach. Tieto pravdepodobnosti ale užívateľ nebude môcť zadať v užívateľskom grafickom rozhraní. Keďže každá značka má definované, aké môže mať kľúče s hodnotami (*key=value*), bude potrebné pre každú takúto dvojicu hodnôt priradiť hodnoty, ktoré budú vytvárať heat vrstvu.

4.5 Funkcionalita

Analýzou požiadavkov sa podarilo navrhnuť systém, ktorý bude spĺňať tieto požiadavky. Primárnym cieľom je vytvoriť systém, ktorý bude zobrazovať mapu s vytvorenou heat vrstvou zo získaných dát.

Vytváranie oblasti

Oblasť, z ktorej bude chcieť zistiť odhad pravdepodobnosti osôb v danej oblasti, bude možné vybrať pomocou obdĺžnika, ktorý užívateľ nakreslí na mapu. Z vytvorenej oblasti budú následne použité hraničné súradnice daného nakresleného obdĺžnika na vytvorenie dátového rámca s geografickými objektami.

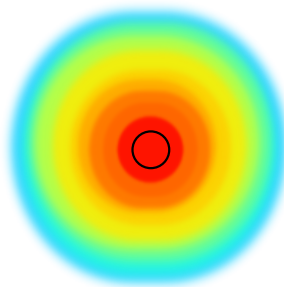
Sťahovanie dát

Získané hraničné súradnice sa použijú pri vytváraní heat vrstvy. Predtým by si systém mal stiahnuť dáta do svojej pamäte. V týchto dátach by mali byť uložené informácie o geometrických objektoch, ktoré sa nachádzajú v danej oblasti. Tieto informácie by mal systém získať pomocou požiadavku na server alebo ich by mal mať uložené lokálne. Podľa [20] existuje vyše 86 tisíc rôznych kľúčov a vyše 132 miliónov rôznych značiek na celej mape sveta. Keďže na vytvorenie heat vrstvy potrebujeme väčšinu značiek, pri takom veľkom množstve rôznych kľúčov by nebolo príliš efektívne sťahovať dáta z webového rozhrania API. Preto je vhodnejšou alternatívou použitie lokálnych súborov, ktoré si bude užívateľ môcť stiahnuť a vytvárať požiadavky na tieto lokálne súbory.

Vytváranie heat vrstvy

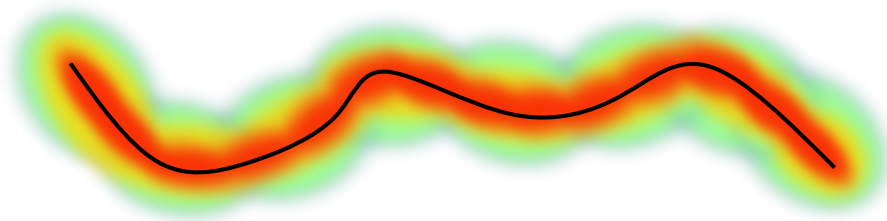
Vytvorená heat vrstva by mala byť súvislá u všetkých geometrických objektoch. Nemala by nastať situácia, kedy časť geometrického objektu je pokrytá iba z časti vrstvou heat mapy a zvyšná časť je bez tejto vrstvy. Keďže geometrické objekty môžu byť jedným z troch typov (bod, čiara alebo polygón), bude potrebné vyriešiť, ako sa bude zobrazovať heat vrstva pre jednotlivé geometrické objekty.

Bod môže na mape predstavovať napríklad autobusovú zastávku. U tohto typu je vykreslenie heat vrstvy najjednoduchšie zo všetkých geometrických objektov, pretože budú stačiť dve hodnoty na vykreslenie – zemepisná dĺžka a zemepisná šírka daného bodu. Na obrázku 4.3 je možné vidieť, ako by sa mala vykresliť heat vrstva pre určitý bod.



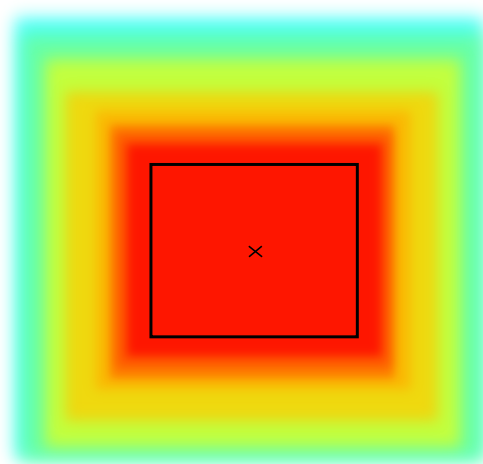
Obr. 4.3: Heat vrstva na geometrickom objekte typu bod.

Čiara môže na mape predstavovať napríklad cestu. Tento typ bude potrebné upraviť tak, aby heat vrstva na danom geometrickom objekte bola súvislá. Na obrázku 4.4 je možné vidieť, ako by mala vyzeráť súvislá vykreslená heat vrstva pre čiaru.



Obr. 4.4: Heat vrstva na geometrickom objekte typu čiara.

Polygón môže na mape predstavovať rôzne budovy, napríklad školu. Vnútro polygónu a najbližšia oblasť okolo polygónu by mala byť v červenej farbe. V tejto oblasti by mala byť pravdepodobnosť výskytu osôb najvyššia. Postupne sa farba mení na studenšie farby a vzniká tým súvislý prechod farieb ako u bodu a čiary. Na obrázku 4.5 je možné vidieť, ako by mohla vyzeráť heat vrstva pre polygón.



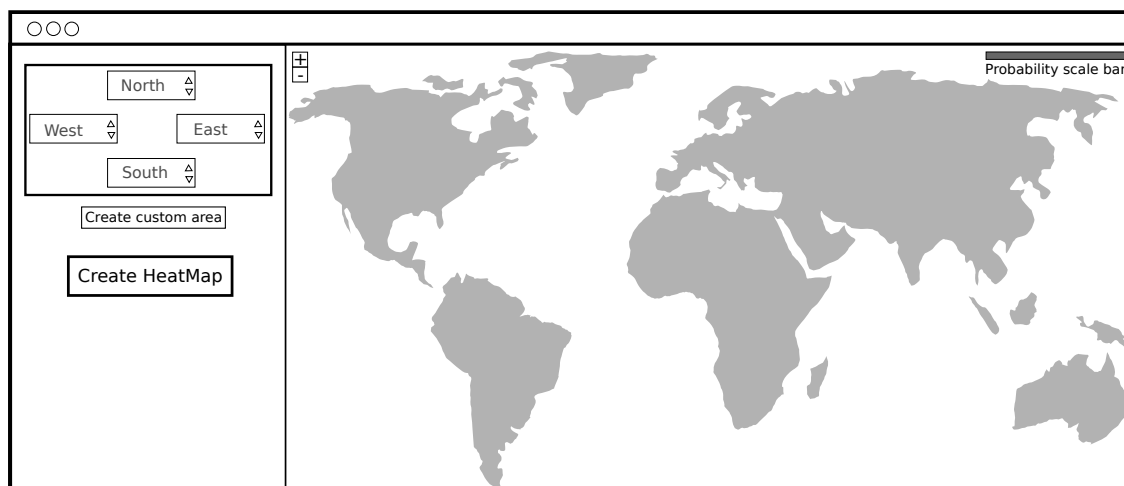
Obr. 4.5: Heat vrstva na geometrickom objekte typu polygón. Prechod vrstiev sa vyvíja od stredového bodu polygónu, ktorý je vyznačený na obrázku značkou ×.

4.6 Návrh užívateľského rozhrania

Hlavným cieľom užívateľského rozhrania bude umožnenie užívateľovi vytvoriť plochu, na ktorej sa má vykresliť heat vrstva. Časť užívateľského rozhrania bude inšpirovaná užívateľským rozhraním stránky OpenStreetMap¹, a to získavanie vstupných súradníc. Na obrázku 4.6 je zobrazený návrh užívateľského rozhrania. Na ľavom paneli bude užívateľ môcť zadávať

¹<https://www.openstreetmap.org>

vstupné súradnice, vytvárať plochu, na ktorej chce vykresliť heat vrstvu, upravovať veľkosť plochy pomocou vstupného formulára. Na ľavom paneli sa bude nachádzať aj tlačítko na vytvorenie mapy s heat vrstvou. Napravo od ľavého panela sa bude nachádzať mapa, na ktorej sa bude zobrazovať heat vrstva. Aké hodnoty pravdepodobností dané farby predstavujú, bude zobrazené na škále v pravom hornom rohu mapy.



Obr. 4.6: Návrh užívateľského rozhrania.

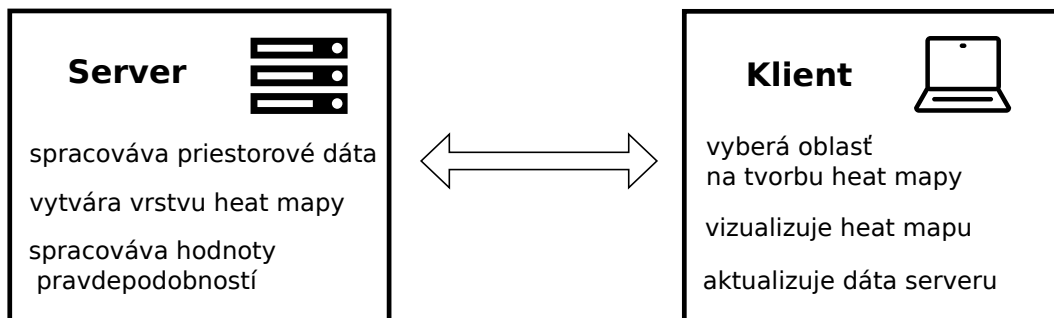
Kapitola 5

Implementácia

V tejto kapitole je popísaná architektúra navrhutej aplikácie, taktiež sú v tejto kapitole popísané použité technológie a ako boli použité. Okrem toho je v tejto kapitole popis implementácie funkcionality a popis implementácie grafického užívateľského rozhrania.

5.1 Architektúra

Navrhnutá aplikácia rozširuje funkcionality projektu OpenStreetMap. Aplikácia pridáva do projektu OpenStreetMap možnosť pridávania teplotnej vrstvy do zobrazenej mapy. Užívateľ si bude môcť vybrať oblasť mapy, pre ktorú by chcel zobraziť mapu s teplotnou vrstvou. Aplikácia teda nezobrazí vygenerovanú mapu s teplotnou vrstvou pre celý svet naraz, ale iba pre časť mapy. Architektúra systému je založená na architektúre klient-server. Takýto typ architektúry umožňuje vytvoriť moduly, ktoré síce vykonávajú odlišné akcie, ale spoločne vytvárajú funkčný systém. Na obrázku 5.1 je možné vidieť, aké akcie vykonáva webový server a aké akcie vykonáva klientská časť aplikácie.



Obr. 5.1: Architektúra navrhovanej aplikácie.

Implementácia webového serveru je napísaná v jazyku Python. Klientská časť aplikácie sa ukladá do prehliadača užívateľa a zobrazuje sa užívateľovi v prehliadači, takže je zložená z jazykov, ktoré slúžia pri vývoji webových aplikácií (Javascript, HTML a CSS). Obidve časti aplikácie si vzájomne vymieňajú informácie pomocou zvoleného webového frameworku. Okrem toho server má uložené informácie o značkách a ich pravdepodobnostiach, ktoré sa nachádzajú na mape.

5.2 Použité technológie

Keďže aplikácia je navrhnutá ako klient-server aplikácia, je potrebné využiť vhodné technológie na webovom serveri aj na klientskej strane aplikácie. Funkcionalita danej aplikácie rozširuje pridávanie heat vrstvy do mapy z OpenStreetMap. Na strane servera je nutné spracovať priestorové dáta a na klientskej strane aplikácie zvoliť technológiu, ktorá by umožňovala zobrazovanie heat vrstvy na mape. Vytváranie aj takejto vrstvy na mapu umožňuje technológia *Leaflet* a jej rozšírenia vytvorené open-source komunitou. Aby si vedel užívateľ zobrazovať danú vrstvu na mape, je potrebné zvoliť vhodnú technológiu, ktorá dokáže komunikovať s klientskou stranou aplikácie.

5.3 Webový server

Na webovom serveri prebieha spracovávanie priestorových dát a ich spracovávanie na výstup. Okrem knižnice *GeoPandas* v programovacom jazyku Python neexistuje veľa technológií, ktoré by zjednodušovali spracovávanie priestorových dát. Keďže na zobrazovanie dát na klientskej strane aplikácie je použitá knižnica *Leaflet* z jazyka Javascript, ktorý je používaný najmä pri tvorbe webových stránok, použitie webového frameworku je vhodnou voľbou. Okrem toho webové frameworky uľahčujú vytváranie webových aplikácií a zobrazovanie spracovaných dát na výstup.

Flask

Flask¹ je mikro webový framework napísaný v programovacom jazyku Python. Flask nevyžaduje konkrétne nástroje ani ďalšie vnútorné knižnice. Aj keď sa tento framework považuje za mikro framework, neznamená to, že nie je možné ho rozširovať o rôzne služby [11].

Zobrazovať dáta vo webových aplikáciách je možné pomocou rôznych webových frameworkoch. Aplikácia je navrhnutá ako jednostránková aplikácia a nevyžaduje vytváranie databázovej schémy. Na základe týchto vlastností som sa rozhodol nepoužiť webové frameworky, ktoré by boli príliš robustné na použitie v tejto aplikácii. Preto som sa rozhodol použiť webový framework Flask.

GeoPandas

GeoPandas² je open-source projekt, ktorý umožňuje jednoduchú manipuláciu geografických dát v jazyku Python. Tento projekt je pokročilejší nástroj umožňujúci načítavanie GIS dát (.geojson, .gdb, .shp, ...), manipuláciu s GIS dátami, manipuláciu s geometrickými útvarmi, prevádzanie súradnicových systémov a vykresľovanie máp. Základnými dátovými štruktúrami projektu GeoPandas sú *GeoDataFrame* a *GeoSeries*. Tieto dve dátové štruktúry sú nadstavbou knižnice *Pandas*³, pretože rozširujú možnosti dátových štruktúr *Series* a *DataFrame* z tejto knižnice.

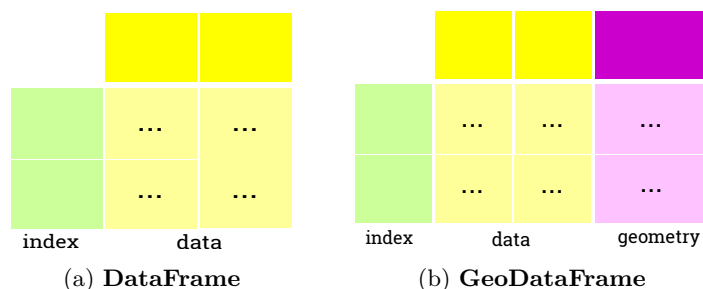
GeoDataFrame je dvojrozmerná dátová štruktúra, podobná tabuľke s riadkami a stĺpcami a je podtriedou *pandas.DataFrame* s pridaným stĺpcom *geometry* (obrázok 5.2).

GeoSeries je jednorozmerná dátová štruktúra, ktorá je podtriedou *pandas.Series* a umožňuje ukladať geometrické objekty – body, čiary, polygóny, prípadne ich násobné varianty. Predstavuje stĺpec *geometry* v dátovej štruktúre *GeoDataFrame*.

¹<https://flask.palletsprojects.com/en/2.1.x>

²<https://geopandas.org/en/stable>

³<https://pandas.pydata.org>



Obr. 5.2: Rozdiel medzi dátovými štruktúrami DataFrame a GeoDataFrame.

Priestorové dáta bolo potrebné spracovať efektívnym spôsobom. V programovacích jazykoch neexistuje veľa overených technológií, ktoré by podporovali prácu s priestorovými dátami. Možnou alternatívou je použitie GIS softvérov, ktoré ale nie všetky sú zadarmo. Preto som sa na základe tohto rozhodol použiť programovací jazyk Python s importovanou knižnicou GeoPandas, ktorá umožňuje spracovávanie priestorových dát.

Ostatné nástroje

Okrem ukladania dát do dátových rámcov je potrebné ich aj upraviť do formy, v ktorej by sa dali použiť pri zobrazovaní heat vrstvy na strane klienta. Medzi vhodné nástroje, ktoré umožňujú takúto úpravu dát v jazyku Python, patria balíky nástrojov *PyProj* a *Shapely*.

PyProj⁴ je rozhranie v jazyku Python na PROJ⁵. PROJ je softvér, ktorý sa používa na transformovanie geografických súradníc z jedného súradnicového systému do iného súradnicového systému.

Shapely⁶ je balík nástrojov v jazyku Python na manipuláciu a analýzu rovinných geometrických objektov. Tento balík je vhodné použiť s už zmieneným nástrojom PyProj, pretože Shapely sa nezaobrá tým, v akom súradnicovom systéme je upravovaný geometrický objekt.

Možnosť transformácie súradnicového systému na iný súradnicový systém bol využitý aj pri implementácii aplikácie. U niektorých geometrických útvaroch (konkrétne u čiar) bolo potrebné zistiť, aké sú hodnoty daného geografického objektu v inom súradnicovom systéme. Implicitne sú dátové rámce s dátami o geografických objektoch v geografickom súradnicovom systéme (EPSG:4326), ktorý využíva uhlové jednotky. Takáto projekcia neumožňuje rôzne úpravy takýchto geometrických objektov. Transformovaním na projektovaný súradnicový systém (EPSG:3857) je možné dostať vzdialenosti bodov na mape v metroch.

5.4 Klientská časť

Na klientskej časti aplikácie vyberá užívateľ oblasť, na ktorej sa má vytvoriť heat vrstva. Keď užívateľ vyberie oblasť, pošle sa požiadavka na server, na ktorom sa spracujú zadané vstupné hodnoty a odošlú sa dáta na tvorbu heat vrstvy späť klientovi.

⁴<https://pyproj4.github.io/pyproj/stable>

⁵<https://proj.org>

⁶<https://shapely.readthedocs.io/en/stable>

Existuje veľa knižníc, ktoré umožňujú vytvárať heat grafy resp. heat vrstvy. Vytváranie heat vrstiev je možné aj v jazyku Python pomocou knižnice *Seaborn*⁷, ale táto knižnica neumožňuje zobrazovanie heat vrstvy spoločne už so zobrazenou mapou. Takisto prechod medzi jednotlivými farbami nie je plynulý. Keďže heat vrstvy sa majú vytvárať na klientskej časti aplikácie, je potrebné použiť nástroje jazykov, ktoré sa používajú na strane klienta. Preto som sa rozhodol použiť knižnicu *Leaflet* implementovanú v jazyku Javascript.

5.4.1 Leaflet

Leaflet⁸ je open-source knižnica v programovacom jazyku JavaScript, používaná na vytváranie mapových webových aplikácií. Táto knižnica je podporovaná na väčšine mobilných a desktopových zariadeniach. Leaflet umožňuje vývojárom jednoducho zobrazovať webové mapy s dlaždicami, získavať geopriestorové dáta zo súborov typu GeoJSON, vytvárať interaktívne vrstvy a pod.

Okrem základnej funkcionality existujú rôzne rozšírenia, ktoré rozširujú základnú funkcionality knižnice Leaflet. V mojej aplikácii sa používajú dve rozšírenia k tejto knižnici: `Leaflet.heat` a `Leaflet.draw`.

Rozšírenie Leaflet.heat

Základom tejto knižnice je knižnica `simpleheat` z jazyka JavaScript, ktorá sa používa na kreslenie heat máp na plátno. Na vytvorenie vrstvy heat vrstvy na mapu sa používa funkcia `L.heatLayer(latlngs, options)`, kde:

- `latlngs` – predstavuje dvojrozmerné pole, kde každé pole v tomto poli je definované dvojicou (zemepisná dĺžka, zemepisná šírka) resp. trojicou hodnôt (zemepisná dĺžka, zemepisná šírka, intenzita),
- `options` – má viacero možností, ktoré sa dajú zdefinovať pre danú vrstvu, najdôležitejšie z nich pre moju aplikáciu sú:
 - `max` – maximálna intenzita bodu, implicitná hodnota je 1.0,
 - `radius` – rádius každého bodu na heat mape, implicitná hodnota je 25,
 - `blur` – množstvo rozptýlenia, implicitná hodnota je 15,
 - `gradient` – konfigurácia prechodu farieb.

Na obrázku 5.3 je možné vidieť ako sa dá pridať heat vrstva do mapy a ako daná vrstva vyzerá na mape. Hodnoty intenzít, ktoré sú tretou hodnotou každého vykresleného bodu, udávajú hodnotu pravdepodobnosti pre danú značku na mape. To znamená, že ak ide napríklad o školu, na mape sa vykreslí bod pre danú značku s intenzitou 1.0. Na druhú stranu, ak by išlo o značku, ktorá by značila lesnú cestu, intenzita farby pre danú značku by bola nižšia.

Rozšírenie Leaflet.draw

Na to, aby užívateľ mohol vybrať určitú oblasť z mapy, na ktorej sa má vykresliť heat vrstva, bolo potrebné použiť rozšírenie `Leaflet.draw`. Ako je možné vidieť na obrázku 5.4, na bočnom

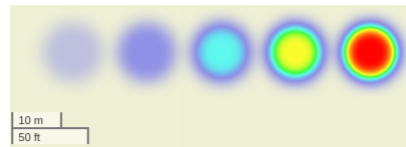
⁷<https://seaborn.pydata.org/generated/seaborn.heatmap.html>

⁸<https://leafletjs.com>


```

var heat = L.heatLayer([
  // lat, lng, intensity
  [49.1784, 16.7772, 0.2],
  [49.1784, 16.7774, 0.4],
  [49.1784, 16.7776, 0.6],
  [49.1784, 16.7778, 0.8],
  [49.1784, 16.7780, 1.0]
], {radius: 30}).addTo(map);

```



Obr. 5.3: **Rozšírenie Leaflet.heat.** Vľavo sa nachádza implementácia vrstvy, zoradená vzostupne podľa hodnôt intenzít, vpravo sa nachádza zobrazenie týchto bodov na mape.

paneli mapy v hornom ľavom rohu sa nachádza panel nástrojov, ktorý umožňuje kresliť geometrické útvary na mapu a následne ich upravovať alebo aj vymazávať. Pre jednoduchosť aplikácie na vybratie určitej oblasti pomocou geometrických tvarov bol použitý iba jeden geometrický tvar – obdĺžnik.



Obr. 5.4: **Rozšírenie Leaflet.draw** pridáva možnosť kreslenia geometrických útvarov na mapu - krivky, polygóny, obdĺžniky, kruhy, značky a kruhové značky. Okrem toho je možné pomocou tohto rozšírenia upravovať, presúvať a vymazávať geometrické útvary.

5.4.2 HTML

Jazyk HTML⁹ je jazyk určený na vytváranie webových stránok. Ako už názov napovedá, slovo *hypertextový* naznačuje možnosť odkazov jednotlivých stránok na ostatné. Slovo *značkovací* v spojitosti s jazykmi predstavuje jazyky, ktoré dopĺňujú text o tzv. *značky* resp. *tagy*. Všetky značky majú svoje atribúty, ktoré upresňujú jej parametre. Jednotlivé značky je možné do seba vnárať a spolu tak vytvárajú hierarchickú štruktúru [17].

⁹HyperText Markup Language, v preklade *Hypertextový značkovací jazyk*.

5.4.3 CSS

Štruktúru webových stránok je možné vytvoriť použitím len jazyka HTML, ale na to, aby stránky vyzerali užívateľsky prívetivo, je nutné použiť aj iné nástroje, ktoré pôsobia na užívateľa prívetivejším dojmom. Preto vznikol mechanizmus *CSS* na vizuálne formátovanie webových stránok. Tento mechanizmus slúži na popis kaskádových štýlov HTML dokumentu.

5.4.4 JQuery

JQuery¹⁰ je rýchla a cross-browser¹¹ knižnica z jazyka JavaScript. Zjednodušuje prechádzanie dokumentov HTML, výber DOM¹² elementov, kontrolu udalostí alebo aj vytváranie animácií. Výhodou tejto knižnice je, že je jednoduchou knižnicou na vývoj webových aplikácií. Oproti iným knižniciam v JavaScripte nevyžaduje lepšiu skúsenosť s jazykom JavaScript. Táto knižnica umožňuje okrem už zmienených možností, jednoduchšiu obsluhu *AJAX* volaní.

5.4.5 AJAX

Technológia *AJAX*¹³ umožňuje meniť obsah stránok bez potreby kompletného načítania zo servera. Keďže protokol HTTP je bezstavový, akékoľvek stavové informácie je nutné poslať pri každej požiadavke servera a opačne. *AJAX* nám umožňuje sa takýmto postupom vyhnúť [18]. Ako je možné poznať z názvu, *AJAX* je kombinácia viacerých prvkov:

- HTML a CSS na prezentovanie informácií,
- DOM na zobrazenie prezentovaných informácií,
- metóda na výmenu dát medzi prehliadačom a serverom (napr. XMLHttpRequest objekt),
- formát dát posielený medzi prehliadačom a serverom (napr. XML)

V mojej aplikácii som použil *AJAX* technológiu na aktualizáciu súradníc pri zmene veľkosti alebo presune oblasti určenej na vytvorenie heat mapy.

5.4.6 Bootstrap

Bootstrap je sada nástrojov, ktorá obsahuje rozšírenie pre jazyky HTML, CSS a JavaScript. Umožňuje vytvárať pomocou veľkého množstva rôznorodých komponentov interaktívne webové stránky. Vyžaduje určitú znalosť HTML a CSS, pretože väčšinu komponentov je možné vkladať iba pomocou HTML a CSS [4].

V mojej aplikácii používam tzv. *Alert* komponentu t.j. výstražnú správu. Ak sa nepodarilo vytvoriť heat mapa z parametrov poslaných na server, server pošle správu užívateľovi, že heat mapa sa pre zadanú oblasť nepodarilo vytvoriť.

⁹Cascading Style Sheets

¹⁰<https://jquery.com>

¹¹Cross-browser kompatibilita je schopnosť webových stránok fungovať v rôznych prehliadačoch.

¹²Document Object Model

¹³Asynchronous JavaScript and XML

5.5 Implementácia funkcionality

Implementácia funkcionality vychádza z návrhu, ktorá bola popísaná v kapitole 4. Ako implementačný programovací jazyk bol vybraný Python spolu s knižnicami, ktoré boli popísané v tejto kapitole. Pri vytváraní heat vrstvy je potrebné urobiť viacero úprav dát, aby sa v aplikácii zobrazovali relevantné dáta na mape.

Spracovanie hodnôt pravdepodobností

Prvou potrebnou vecou, ktorú systém spracováva, sú dáta pravdepodobností pre jednotlivé značky. Dáta o značkách sú uložené v súbore, ktorý je vo formáte JSON. Tieto dáta sú prednastavené s tým, že tieto hodnoty užívateľ môže pozmeniť v tomto súbore, prípadne pridať alebo odstrániť niektoré hodnoty. Súbor sa v systéme spracuje a uloží sa ako asociatívne pole do pamäte, s ktorým bude systém pracovať pri úprave dátového rámca.

Vytvorenie dátového rámca

Keďže pri vyššom počte hodnôt pravdepodobností pre jednotlivé značky by počet požiadaviek na webové rozhranie API bol vysoký, systém spracováva požiadavky z lokálnych súborov. Tieto súbory sú vo formáte *shapefile* a sú získané zo stránky Geofabrik¹⁴. Súbory sú pomenované podľa názvu kľúča, ktorý sa nachádza v danom súbore. Keďže súbory tohto formátu sú naviazané medzi sebou, pri spracovávaní týchto súborov nám stačí súbor vo formáte **.shp*, z ktorého je možné vytvoriť dátový rámec so stĺpcom o geografických objektoch.

Vytvorený dátový rámec má viacero stĺpcov, ale využiteľné pre tento systém je iba stĺpec s typom značky a stĺpec s geometrickými objektami. Každý riadok v tomto rámci predstavuje jeden geometrický objekt na mape. Ak sa daná značka pre daný geometrický objekt nenachádza vo vytvorenom asociatívnom poli s pravdepodobnosťami pre značky, riadok s touto značkou sa preskočí.

Úprava geometrických objektov

Každý riadok v dátovom rámci sa spracuje podľa toho, aký predstavuje geometrický objekt. Jednotlivé riadky v dátovom rámci je potrebné upraviť do zoznamu bodov, ktorý sa použije pri vytváraní heat vrstvy pomocou knižnice *Leaflet.heat*. Vytvorením zoznamu bodov vznikne iba jedna súvislá heat vrstva, ktorá je vhodnejšia pri zobrazovaní na mape.

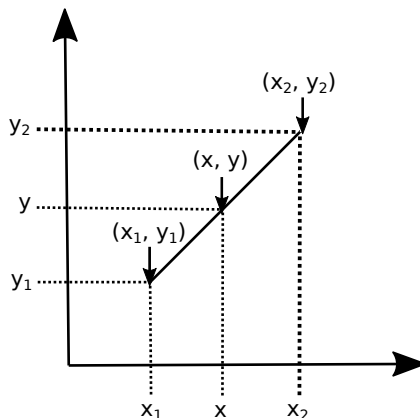
Bod predstavuje jednoduchý geometrický objekt na mape, pretože je reprezentovaný iba dvoma hodnotami – zemepisnou dĺžkou a zemepisnou šírkou. K týmto dvom hodnotám sa pridá tretia hodnota, ktorá pridáva intenzitu k danému bodu.

Čiara je geometrický objekt na mape, ktorý slúži na zobrazovanie ciest. Heat vrstva pre cesty bez zákrut by sa bez dodatočnej úpravy vykreslili iba s dvoma bodmi – počiatočným bodom a koncovým bodom cesty. Takéto riešenie nie je vhodné, preto je lepším riešením takéhoto problému použiť interpoláciu úsečky. Na interpoláciu úsečky je použitá rovnica 5.1 odvodená z obrázka 5.5:

$$y = y_1 + \frac{(x - x_1)(y_2 - y_1)}{x_2 - x_1}, \quad (5.1)$$

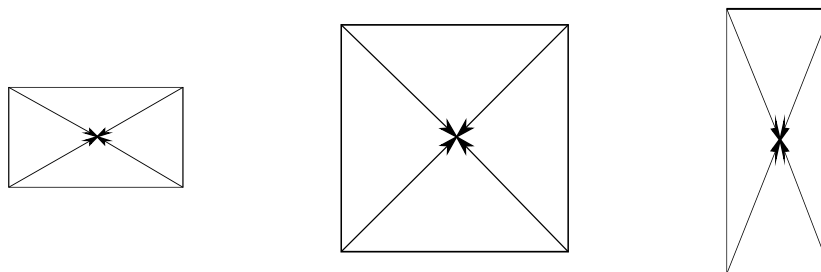
¹⁴<https://download.geofabrik.de>

kde x_1 a y_1 sú prvé súradnice, x_2 a y_2 sú druhé súradnice, x je stredná hodnota hodnôt x_1 a x_2 , a y je hľadaná interpolovaná hodnota. Koľkokrát interpolujeme danú úsečku závisí od dĺžky danej úsečky.



Obr. 5.5: Interpolácia úsečky.

Polygón je geometrický objekt na mape, ktorý slúži na zobrazovanie budov. Z polygónov sa dajú získať iba okrajové body, body vo vnútrajšku polygónov je potrebné zistiť pomocou vhodnej metódy. Najjednoduchším spôsobom, ktorý som použil aj pri implementácii, je získanie stredového bodu polygónu. Na obrázku 5.6 je zobrazený zistený stredový bod pre rôzne typy obdĺžnikov. Na zistenie stredového bodu polygónu bola použitá funkcia `centroid` z knižnice `GeoPandas`.



Obr. 5.6: Zistenie stredového bodu pre rôzne typy polygónov.

Vytvorenie heat vrstvy

Po uprave geometrických objektov na body a následné vytvorenie zoznamu z týchto bodov je možné vytvoriť heat vrstvu na mape. Pri vytváraní heat vrstvy sa zapisuje vytvorený list bodov do súboru HTML. Tento súbor je poslaný klientovi do prehliadača po dokončení požiadavky.

5.6 Implementácia užívateľského rozhrania

Užívateľské rozhranie sa skladá z viacerých komponentov, ktoré vytvárajú jednotný celok. V ľavom paneli sa nachádza panel, v ktorom môže užívateľ zadávať vstupné súradnice

a upravovať ich. V hlavnej časti obrazovky sa nachádza zobrazená mapa, na ktorú sa vykresľuje heat vrstva. Na zobrazenej mape sa nachádza aj škála pre vytvorenú heat vrstvu.

Získavanie vstupných súradníc

Vstupné súradnice, ktoré zadal užívateľ, je potrebné aktualizovať pri každom zmenšení alebo zväčšení mapy resp. oblasti vytvorenej pomocou nástroja `Leaflet.draw`. Pri nakreslení oblasti sa pomocou nástroja AJAX, ktorý bez potreby kompletného načítania stránky zo servera, upraví súradnice vo formulári na ľavom paneli aplikácie. AJAX správa sa po šle po vytvorení oblasti na mape pomocou udalosti `L.Draw.Event.CREATED` a spustí sa udalosť `L.Draw.Event.EDITSTART`. Táto udalosť umožní užívateľovi meniť veľkosť aj polohu nakreslenej oblasti. Odkaz na vytvorenie oblasti (`Create custom area`) sa po tejto udalosti zmení na odkaz na vymazanie vytvorenej oblasti (`Delete custom area`).

Škála pravdepodobností

Zo získaných hodnôt pravdepodobností pre každú značku je možné vytvoriť škálu, ktorá ukazuje, aká je odhad pravdepodobnosti osôb v danej oblasti. Na vytvorenie tejto škály bol použitý nástroj *D3.js*¹⁵. Keďže funkcia `Leaflet.heatLayer()` umožňuje pridávať vlastne vytvorenú konfiguráciu prechodu farieb, táto škála je prepojená s vytvorenou heat vrstvou. Aby sa zjednodušilo vytváranie takejto škály, na vytvorenie tejto farebnej škály bol použitý nástroj *branca*¹⁶, konkrétne modul pre prácu s farebnými mapami. Vo výslednej implementácii tejto aplikácie sa tento nástroj nepoužíva.

¹⁵<https://d3js.org>

¹⁶<https://python-visualization.github.io/branca>

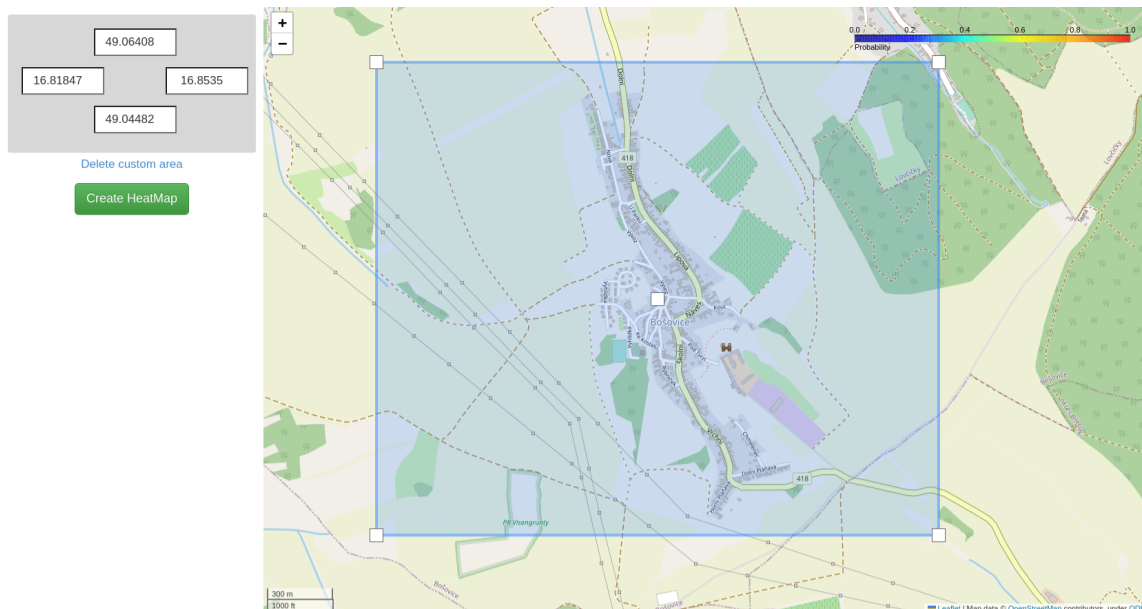
Kapitola 6

Vyhodnotenie výsledkov

Táto kapitola sa zaoberá ovládaním, testovaním a vyhodnotením výsledkov z vytvorenej aplikácie. V tejto kapitole sa nachádza aj popis možných chyby a nevýhody implementovaného systému. Okrem toho, táto kapitola popisuje ako by sa dali vylepšiť získané dáta.

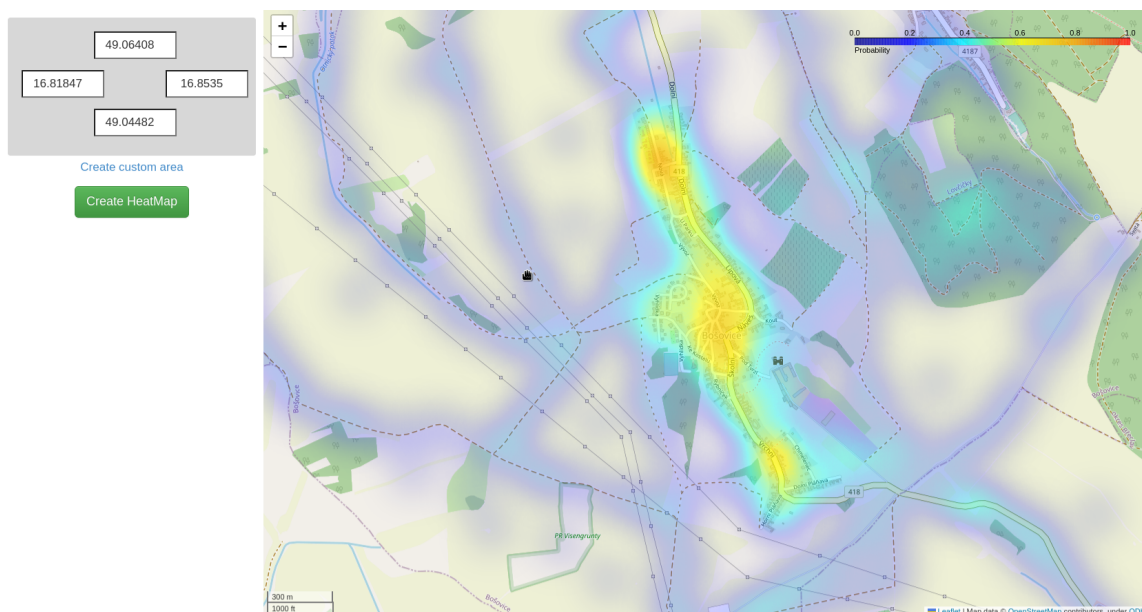
6.1 Ovládanie

Keďže aplikácia má slúžiť primárne pri plánovaní výstavby elektroenergetických zariadení na území Českej republiky, pri ovládaní aplikácie je potrebné vybrať oblasť, ktorá sa nachádza na území Českej republiky. Užívateľ si prípadne môže stiahnuť mapové dáta aj pre iné krajiny na svete. Z užívateľského hľadiska je aplikáciu jednoduché ovládať, pretože neobsahuje veľa možností, ktoré užívateľ môže vykonať. Na obrázku 6.1 je snímok obrazovky časti mapy s nakresleným obdĺžnikom, kde sa nachádza obec Bošovice. Hraničné body tohto obdĺžnika sa použijú pri získavaní geografických dát. Veľkosť aj polohu tohto obdĺžnika môže užívateľ ľubovoľne meniť.



Obr. 6.1: Výber oblasti na mape.

Po stlačení tlačítka **Create HeatMap** sa vytvoria zo vstupných súborov dátové rámce. Následne sa k nim pripoja hodnoty pravdepodobností a vytvorí sa z týchto bodov výstupná heat vrstva, ktorá sa zobrazí na mape. Na obrázku 6.2 je možné vidieť, ako vyzerá vytvorená heat vrstva na mape. Táto vrstva sa zobrazí aj mimo vybranej oblasti. Je to z dôvodu, že geografické objekty typu cesta a polygón sú tvorené viacerými bodmi. Niektoré z týchto geografických objektov môžu sčasti presahovať vybranú oblasť. Aby sa zbytočne nevytvárali náročné výpočty v aplikácii, berie sa celá plocha geometrických objektov, ktoré sa nachádzajú aspoň sčasti vo vybranej oblasti.

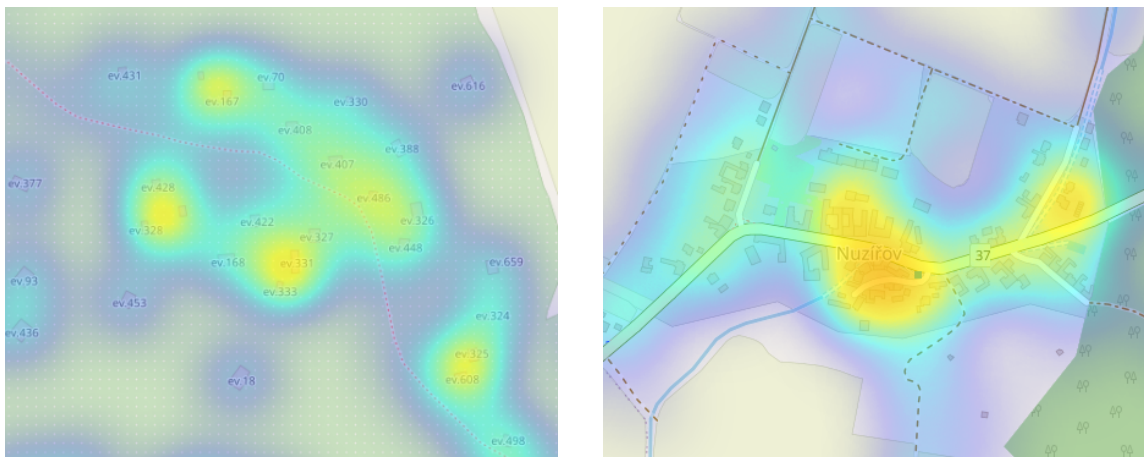


Obr. 6.2: Heat vrstva pre vybranú oblasť.

6.2 Testovanie aplikácie

Pri prvotnej implementácii sa brali mapové dáta pomocou API požiadavkov na server. Takáto implementácia vytvorila veľa požiadavkov na daný server a viedlo to k tomu, že by sa pri väčšom počte značiek v danej oblasti muselo sťahovať veľké množstvo dát. To viedlo k spomaleniu aplikácie. Preto vhodnejšou alternatívou, ktorá bola použitá, bolo stiahnutie dát z webového serveru pred spustením aplikácie. Získanie lokálnych dát viedlo k výraznému zrýchleniu aplikácie a vylepšeniu samotnej implementácie aplikácie.

Hlavným výstupom aplikácie je mapa s vytvorenými heat vrstvami. Na obrázku 6.3 je možné vidieť dva rôzne výstupy aplikácie. Ľavý obrázok predstavuje oblasť v časti Brno-Medlánky, v ktorej sa nachádzajú chaty. V tejto oblasti je heat vrstva zobrazená iba nad chatami, okolité oblasti sú bez takejto vrstvy. Pravý obrázok predstavuje zastavanú oblasť, konkrétne o obec Nuzířov resp. ide o časť obce Malhostovice v okrese Brno-venkov v Juhomoravskom kraji. Tento obrázok ukazuje, že najväčšia pravdepodobnosť odhadu výskytu osôb v tejto obci je v strednej časti obce.

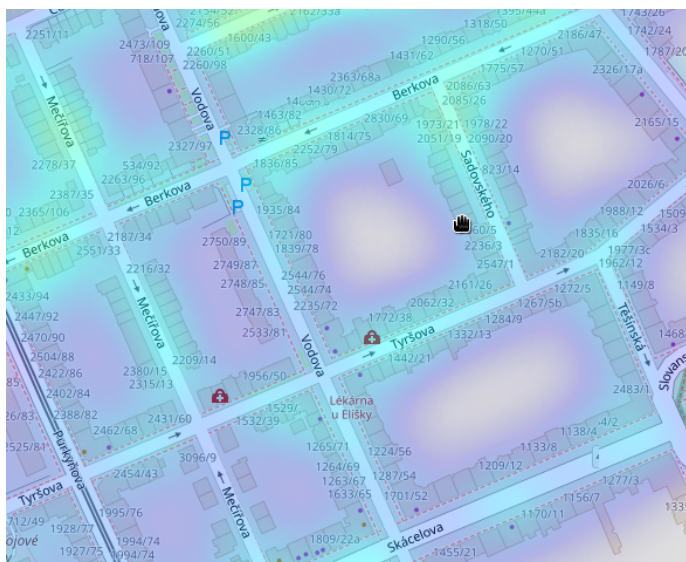


Obr. 6.3: Heat vrstva dvoch rôznych oblastí mapy.

6.3 Obmedzenia aplikácie

Odhad pravdepodobnosti v danej oblasti je iba teoretický, pretože aplikácia zatiaľ neberie do úvahy viacero aspektov. Všetky aspekty nie je možné implementovať triviálne, pretože nie všetky sa dajú vyriešiť nástrojmi, ktoré existujú v programovacích jazykoch.

Zastavané oblasti v obciach by mali byť tiež pokryté heat vrstvou. Na obrázku 6.4 je možné vidieť, že časti zastavaných oblastí sú bez vytvorenej heat vrstvy. Možným riešením daného problému by bolo použitie rastrovej siete v danej oblasti, v ktorej by sa zistili body vo vnútri tejto zastavanej plochy. Týmto by sa vytvorili body v danej oblasti a bolo by možné vytvoriť pomocou toho súvislú heat vrstvu na danej mape.



Obr. 6.4: Heat vrstva zastavanej časti Brna.

Heat vrstva pre polygóny by mala mať väčší rádius. Nestačí, keď sa pre polygón vytvorí vrstva iba vo vnútornej časti budovy, vrstva by mala mať takú istú hodnotu aj v najbližšej oblasti budovy. Na obrázku 6.5 je vidieť, že časť polygónov je bez vytvorenej heat vrstvy a takisto aj najbližšie okolie polygónov je bez tejto vrstvy.



Obr. 6.5: Heat vrstva by sa mala vykresliť aj vo vonkajšej časti polygónov.

Cesta medzi dvoma významnými objektami by mala mať vyššiu hodnotu pravdepodobnosti. Napríklad ak si predstavíme cestu medzi školou a autobusovou stanicou (obidva objekty majú pravdepodobnosť 1.0), tak vieme, že na takejto ceste by mala byť pre danú cestu vyššia pravdepodobnosť výskytu osôb. Možným riešením by bolo použiť pri vytváraní vrstvy kataster nehnuteľností alebo územný plán obce.

Nepresnosti v získaných mapových dátach vytvára na mape body heat vrstvy, ktoré by sa nemali s vytvorenou vrstvou zobrazovať. Aj keď väčšina mapových dát zo stráky Geofabrik býva aktualizovaná každý deň, aj tak sa v týchto mapových dátach vyskytujú chyby. Takéto potenciálne chyby by sa dali odstrániť pridaním možnosti užívateľovi upravovať vytvorenú heat vrstvu. Funkcionalita takejto vlastnosti by sa dala implementovať pomocou vytvárania heat vrstvy myšou alebo pridaním a odstraňovaním bodov heat vrstvy z mapy pomocou vstupného formulára.

Kapitola 7

Záver

Cieľom tejto bakalárskej práce bolo vytvoriť aplikáciu, ktorá na základe získaných dát vytvorí heat mapu pravdepodobnosti výskytu osôb v určitej oblasti. Aplikáciu sa podarilo úspešne vytvoriť s miernymi nedostatkami.

Pred začatím implementácie bolo potrebné sa zoznámiť s potrebnou teóriou. Obsahom teórie boli zdroje mapových dát a spôsoby ich reprezentácie. Zistil som, že mapové dáta môžu byť reprezentované v rastrovom formáte alebo vo vektorovom formáte a tieto formáty majú rôzne formáty súborov, ktorými môžu byť reprezentované. Pri implementácii boli použité mapové dáta vo vektorovom formáte, pretože poskytujú informácie o jednotlivých geografických objektoch na mape. Potom bolo nutné naštudovať, aké sú možnosti dolovania dát z voľne dostupných zdrojov. Zistil som, že najjednoduchším spôsobom je získavanie dát zo stránok projektu OpenStreetMap.

Pri implementácii boli zvolené vhodné technológie, ktoré vyšli z návrhu aplikácie. Samotná aplikácia pozostáva z primárnej funkcionality a z užívateľského rozhrania. Primárnu funkcionality zahŕňa získanie dát a ich spracovanie pomocou knižnice GeoPandas do dvojrozmernej dátovej štruktúry. Vytvorený dátový rámec obsahuje geografické objekty a tie sa využívajú spoločne so získanými pravdepodobnosťami na vytvorenie heat vrstvy na mape pomocou knižnice Leaflet. Takáto vytvorená heat vrstva na mape reprezentuje pravdepodobný odhad výskytu osôb v určitej oblasti.

Vypracovaním tejto bakalárskej práce som spoznal, ako je možné získavať mapové dáta a ako sú reprezentované na mape. Počas testovania aplikácie som narazil na viacero problémov. Preto by som chcel v práci pokračovať a vylepšiť aplikáciu o lepšie zobrazovanie heat vrstvy u rôznych geografických objektoch. Určite by som chcel zlepšiť vytváranie heat vrstvy u polygónov, aby zobrazená vrstva bola aj v okolí zobrazeného polygónu na mape. Ďalším možným vylepšením by bolo pridanie viacerých parametrov, ktoré by mohol užívateľ zadať. Teda okrem hodnoty pravdepodobnosti pre danú značku by mohol zadať napríklad aj veľkosť priemeru kruhu heat bodov pre daný geografický objekt.

Literatúra

- [1] Zeměpisné souřadnice. In: *Technický slovník naučný 8 T-Ž*. 1. vyd. Praha: Encyklopedický dům, 2001, s. 540 [cit. 2022-05-02]. ISBN 80-86044-26-2.
- [2] *Čo je to územný plán?* [online]. Košice: UzemnePlany.sk, 2007. ISSN 1338-2772. Dostupné z: <https://www.uzemneplany.sk/co-je-to-uzemny-plan>. Path: Úvod; Územné plány; Čo je to územný plán;.
- [3] *ArcČR 500* [online]. 3.3. ArcČR, ARCDATA PRAHA, ZÚ, ČSÚ, október 2016 [cit. 2022-02-18]. Dostupné z: https://download.arcdata.cz/data/ArcCR_500-3.3-Popis-dat.pdf.
- [4] *Bootstrap – The most popular HTML, CSS, and JS library in the world*. [online]. 2022 [cit. 2022-04-26]. Dostupné z: <https://getbootstrap.com>.
- [5] BAUDYŠ, P. *Katastr a nemovitosti*. 2. vyd. Praha: C. H. Beck, 2010. 312 s. ISBN 9788074003042.
- [6] BENNETT, J. *OpenStreetMap: Be Your Own Cartographer*. 1. vyd. Packt Publishing, september 2010 [cit. 2022-02-19]. Community Experience Distilled. ISBN 978-1-847197-50-4.
- [7] BOLSTAD, P. *GIS Fundamentals: A First Text on Geographic Information Systems*. 6. vyd. XanEdu Publishing Inc, júl 2019 [cit. 2022-05-04]. 764 s. ISBN 978-1-59399-552-2.
- [8] BUTLER, H., DALY, M., DOYLE, A., GILLIES, S., HAGEN, S. et al. *The GeoJSON Format* [online]. RFC 7946. RFC Editor, august 2016 [cit. 2022-04-18]. 28 s. Dostupné z: <https://tools.ietf.org/html/rfc7946>.
- [9] COCKINGS, S. *Geographical Object Types* [online]. University of Southampton. Dostupné z: <https://www.restore.ac.uk/geo-refer/52611cwors00y00000000.php>.
- [10] GISGEOGRAPHY. The Ultimate List of GIS Formats and Geospatial File Extensions. *GIS Geography* [online]. Revidované 16. 4. 2022 [cit. 2022-05-04]. Dostupné z: <https://gisgeography.com/gis-formats>.
- [11] GRINBERG, M. *Flask Web Development*. 1. vyd. O'Reilly Media, Inc., máj 2014. ISBN 978-1-449-37262-0.
- [12] HOFIERKA, J. *Geografické informačné systémy a diaľkový prieskum Zeme*. 1. vyd. Prešov: Prešovská univerzita v Prešove, Fakulta humanitných a prírodných vied, 2003. ISBN 80-8068-219-4.

- [13] HOLMAN, L. *Prostorová data* [online]. Výukový list, 1.0. Praha: Přírodovědecká fakulta Univerzity Karlovy, 2014 [cit. 2022-04-18]. 13 s. Dostupné z: <https://www.natur.cuni.cz/geografie/geoinformatika-kartografie/ke-stazeni/projekty/moderni-geoinformacni-metody-ve-vyuce-gis-a-kartografie/prostorova-data/>.
- [14] HYNEK, J. *Vizualizace dat: Principy, nástroje, webová vizualizace Canvas, SVG, D3.js, Charting libs, Geo* [online]. Revidováno 4. 9. 2021 [cit. 2022-04-29].
- [15] JAFARI, A., BOTH, A., SINGH, D., GUNN, L. a GILES CORTI, B. Building the road network for city-scale active transport simulation models. *Simulation Modelling Practice and Theory* [online]. 1. vyd. Elsevier. Apríl 2021, zv. 114, č. 102398, [cit. 2022-04-19]. DOI: <https://doi.org/10.1016/j.simpat.2021.102398>. ISSN 1569-190X. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S1569190X21001003>.
- [16] JUKIL, G. A. M. *Highway Vertical Alignment Optimization Using Genetic Algorithm (GA)* [online]. Erbil, 2017. [cit. 2022-04-18]. Dizertačná práce. Erbil Polytechnic University. Vedúci práce AL HADAD, B. M. A. Dostupné z: https://www.researchgate.net/publication/330468019_Highway_Vertical_Alignment_Optimization_Using_Genetic_Algorithm_GA.
- [17] KOSEK, J. *HTML - tvorba dokonalých WWW stránek*. 1. vyd. Praha: Grada, júl 2002 [cit. 2022-04-26]. ISBN 80-7169-608-0.
- [18] LACKO Luboslav. *Ajax – Hotová řešení*. 1. vyd. Brno: Computer Press, 2008 [cit. 2022-04-26]. ISBN 978-80-251-2108-5.
- [19] OLAYA, V. *Introduction to GIS*. CreateSpace Independent Publishing Platform, máj 2018. 138 s. ISBN 1720353956.
- [20] OPENSTREETMAP WIKI CONTRIBUTORS. *Statistika databáze* [online]. Revidováno 6.5.2022 [cit. 2022-07-05]. Dostupné z: https://taginfo.openstreetmap.org/reports/database_statistics.
- [21] OPENSTREETMAP WIKI CONTRIBUTORS. *OpenStreetMap Wiki* [online]. 2004. Revidováno 19.7.2020 [cit. 2022-02-18]. Dostupné z: https://wiki.openstreetmap.org/wiki/Main_Page.
- [22] REDLANDS, C. E. S. R. I. Raster bands. *ArcGIS Desktop* [online]. 2011 [cit. 2022-05-04]. Dostupné z: <https://desktop.arcgis.com/en/arcmap/latest/manage-data/raster-and-images/raster-bands.htm>.

Príloha A

Obsah priloženého pamäťového média

/		
	app/	
	data/	
	static/	- zložka na ukladanie mapových dát
	tags.json	- hodnoty pravdepodobností pre jednotlivé značky
	static/	
	css/	
	js/	- zdrojové súbory klientskej časti programu
	templates/	- HTML súbory
	__init__.py	
	osm_map.py	- zdrojový súbor webového serveru
	view.py	- zdrojový súbor webového serveru (pohľad)
	text/	- zdrojové súbory textu (Latex)
	run.py	- spúšťač aplikácie
	requirements.txt	- súbor s potrebnými závislosťami
	README.md	- návod na inštaláciu